PSL
UNIVERSITÉ PARIS

THÈSE DE DOCTORAT
DE L'UNIVERSITÉ PSL

Préparée au College de France

Physical inference from microscopy images

Soutenue par
Sacha ICHBIAH
Le 9 octobre 2023

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

Spécialité
Physique

Composition du jury :

Charles Kervrann Université de Rennes	*Président*
François GRANER Université Paris Diderot	*Examinateur*
Jean-Baptiste MASSON Institut Pasteur	*Examinateur*
Pierre-François LENNE Aix-Marseille Université	*Rapporteur*
Virginie UHLMANN EMBL European Bioinformatics Institute	*Rapporteur*
Hervé TURLIER Collège de France	*Directeur de thèse*

PSL

Contents

Avant-propos

Dès mon plus jeune âge, j'ai toujours baigné dans un optimisme vague. Dans ma famille, on considérait que la vie était difficile. Dans ma famille, on estimait l'intelligence et la débrouillardise. Et surtout, dans ma famille, on préférait la liberté à la sécurité. Je crois que mes parents voulaient simplement faire de nous des enfants normaux. J'ai connu une enfance heureuse, partagée entre la ville et la campagne, avec un père un peu trop sévère et une mère qui ne l'était pas assez. Ayant fréquenté des établissements passables ou médiocres durant la majeure partie de ma jeunesse, j'eus beaucoup de temps pour cultiver de nombreux loisirs en dehors de l'école, ce qui me permit de me passionner pour tout un tas de choses. J'ai toujours beaucoup lu, de la science-fiction, de la fantaisie, des livres d'histoires. Je me suis perdu sur internet, je me suis perdu dans des mondes virtuels, je me suis perdu dans la nature. J'aimais faire du vélo, j'aimais la neige, j'aimais l'hiver, comme j'aimais l'été. On m'a toujours dit que j'allais devenir ingénieur. J'acceptais mon destin, flou lui aussi, avec obéissance.

Je pense avoir compris ma vocation à l'âge de 15 ans. À cet âge-là, inspiré par la lecture de forums sur internet, je portai beaucoup d'attention à la manière d'enregistrer du son. Faire du piano m'intéressait très peu, en revanche, je développai un intérêt aigu pour la façon dont on pouvait capturer fidèlement un signal sonore. Dans un microphone, une onde sonore fait vibrer une membrane, le diaphragme. Ces vibrations sont ensuite converties en un signal électrique, qui est alors amplifié et numérisé. Un tel signal peut ensuite subir divers traitements, chacun correspondant à des opérations mathématiques bien définies. Je découvris ce qu'était une transformée de Fourier, une convolution, ainsi que les différents algorithmes de compression. Je dus également m'intéresser aux problèmes d'acoustique et de réverbérations. Sans le savoir, je faisais mes premières incursions dans le monde de la physique et du traitement du signal.

J'ai ensuite passé deux années en classes préparatoires au Lycée Jean-Baptiste Say, où j'ai été préparé par d'excellents professeurs, et logeais dans un internat pas très loin de là, rue du docteur Blanche. L'un dans l'autre, j'ai pu faire mes études dans un cadre exceptionnel. Au bout de ces deux années, j'eus l'immense chance d'intégrer l'École Polytechnique. Cette école m'apporta énormément, et me laissait toutes les portes ouvertes, ce qui me permit de choisir ce que je devais faire de ma vie. J'y ai effectué deux stages, le premier dans une start-up développant un télescope connecté, où j'ai découvert à la fois le monde de l'entrepreneuriat et de l'analyse d'images. Le second se déroula au sein du département de chimie de l'Université d'Oxford, où mon travail consistait à analyser statistiquement des images de microparticules dans un fluide. Je suis ressorti de cette école plus mûr, endurci, en un mot: adulte.

Au début de ma thèse, en 2020, le monde était devenu vraiment étrange. Les gouvernements occidentaux, plongés dans une sorte de folie, annonçaient via des bilans quotidiens alarmants une pandémie qui peinait à se concrétiser. En réponse, chaque jour de nouvelles règles venaient restreindre notre liberté de façon toujours plus extrême. Les principales sources d'information,

les réseaux sociaux, subissaient tous une censure féroce. L'état et sa police déployaient toute leur force dans des actes de répression absurdes. Dans ce climat, devenir docteur en physique m'intéressait assez peu. Cette thèse fut donc le résultat d'un concours de circonstances très improbable : Mon stage de fin d'études devait initialement se dérouler en Suisse, mais à cause du premier confinement, je dus trouver un autre stage à la dernière minute, et alla dans le laboratoire d'Hervé Turlier au Collège de France. Mon stage était très intéressant, et j'obtins une bourse pour continuer en thèse. Cependant, j'avais envie de changer d'air et de vivre dans le feu de l'action, donc vivre trois années de plus dans un Paris sordide et paranoïaque ne m'intéressait guère. Assez peu motivé, ce contrat m'offrait tout de même beaucoup de liberté. Je restai donc.. Aujourd'hui, tout cela semble lointain. Maintenant que ma thèse est achevée, je ne regrette pas d'y avoir consacré trois ans de ma vie. La science est quelque chose de superbe, et les avancées techniques forment l'essence de l'optimisme pragmatique. Jeune Français de 24 ans, j'ai trouvé peu de buts dignes de vouer ma vie. Plutôt que de remettre en question le monde que nos ancêtres ont bâti et de souscrire aux utopies les plus absurdes, je préfère me tourner vers le progrès technique, convaincu que les hommes qui ont le pouvoir de changer le monde sont ceux qui parviennent à ne pas sombrer dans la démoralisation et le sentiment d'impuissance ambiants, pour s'élever au-dessus d'un quotidien pesant et imaginer un futur nouveau. La foi en Jesus Christ aide à dépasser ce stade de révolte, à accepter que notre monde est brisé et à trouver la paix auprès de lui. Une fois ce deuil accompli, le futur apparaît plein de possibilités, et nous avons le devoir de bâtir ce futur, de le faire émerger de l'imagination, de l'état de possibilité, pour l'ancrer dans le réel. Silencieusement, le progrès suit son chemin, et les discours millénaristes s'estompent, les nouvelles prouesses de l'intelligence artificielle nous montrant que le progrès technique et scientifique pourrait finalement résoudre les problèmes existentiels causés par notre civilisation.

En parallèle, la croyance autrefois répandue en le monopole de l'homme sur l'intelligence est en train de s'effriter, une évolution qui n'est pas dénuée de conséquences saines et positives. Il devient difficile d'embrasser le positivisme lorsque l'on est surpassé par une machine ! Face à la lucidité impartiale des modèles de langage, les impasses intellectuelles s'estompent. La beauté des images générées artificiellement a conquis le cœur des hommes, lassés des élucubrations stériles des pseudos-artistes modernes. Après avoir transformé l'art, elle s'apprête à conquérir le reste du monde. La vérité, dans tous les domaines, finira par se révéler. Dans les années à venir, ceux qui ont bâti des empires sur des mensonges verront leurs impostures exposées en plein jour par la machine, qui se révèle être plus sage que les hommes.

Je tiens à exprimer ma profonde gratitude à Hervé Turlier et à toute son équipe pour m'avoir accueilli dans leur laboratoire, et encadré durant ma thèse. Un grand merci également aux autres chercheurs avec lesquels j'ai eu l'opportunité d'interagir, notamment Andreas Weber, Henry de Belly, Fabrice Delbary, Rémi Dumollard, Alex McDougall, et Guillaume Charras.
Je remercie aussi tous mes professeurs et mentors, mes professeurs en particulier mes professeurs de physique de l'École Polytechnique: Manuel Joffre, Philippe Grangier, Roland Lehoucq, Kirone Mallick, Rémi Monasson, et David Quéré.

Un grand merci à tous les hommes qui ont voué leur vie à servir le Christ, en particulier le père Antoine, le père Thibaud, le père Maxime, le père Dominique, et tous ceux qui m'ont inspiré et accompagné avec charité au fil des années.

Je tiens à saluer les icônes culturelles, passées et présentes, qui m'ont accompagné durant toutes ces années: Honoré de Balzac, Fiodor Dostoïevski, Louis-Ferdinand Céline, Michel Houellebecq, ainsi qu'a tous les hommes qui ont eu a coeur de rendre hommage à la vérité !

Et enfin, les scientifiques, entrepreneurs, visionnaires et ingénieurs qui motivent mes efforts pour faire de cette terre un monde meilleur : Richard Feynman, Freeman Dyson, John Von Neumann, Nikola Tesla, James Dewey Watson, William Shockley, Elon Musk.

Mes remerciements s'adressent aussi à mes parents, pour m'avoir offert une enfance heureuse et pour avoir toujours cru en moi et en mes projets. Vous m'avez inculqué les valeurs de la famille et du devoir. Je suis également reconnaissant envers ma grand-mère Monique, qui m'a enseigné l'importance de la foi, de la charité, et de la force de la volonté, et ma grand-mère Marianne, pour son courage et son intégrité intellectuelle. Un merci spécial à mes sœurs pour leur accompagnement et leur soutien continus au fil des années. Je remercie également mes oncles et tantes, particulièrement Steve pour son appui dans les moments cruciaux, ainsi que mes cousins Mathias et Andreas, toujours présents dans les moments importants. Mes amis, en particulier Louis, Martin, et Benoit, ont éclairé et guidé ma vie avec la lumière et la joie du Christ. Je remercie aussi tous mes autres camarades d'école qui ont croisé mon chemin au cours de ces longues années.

Maintenant que mon doctorat touche à sa fin, mon optimisme est pragmatique. Nous vivons actuellement le tournant le plus extraordinaire de l'histoire de l'humanité. La recherche académique est un univers fascinant ; cependant, les progrès de notre monde sont si rapides qu'un tel corporatisme me semble trop contraignant pour naviguer de manière agile dans les bouleversements qui s'annoncent. Je ne compte pas rester passif et me lance à corps perdu dans le domaine de l'intelligence artificielle. Bâtissons le monde dont nous rêvons !

Chapter 1

Introduction: The Biophysicist

"Nous déclarons que la splendeur du monde s'est enrichie
d'une beauté nouvelle: la beauté de la vitesse."
Filippo Tommaso Marinetti
Manifeste du futurisme
Published in Le Figaro on February 20, 1909.

Amidst the rapid evolution of biology, catalyzed by big data and AI, we will discuss the role of physicists in biology, the value they could contribute, both in theoretical and experimental approaches. To understand this, we will need to delve into epistemology, the nature of the scientific method and modelling, that we will then revisit through the lens of information theory. We also discuss opportunities AI brings to quantitative biology, and share insights about the future of data gathering and analysis.

Dans le contexte de l'évolution rapide de la biologie, catalysée par les grandes données et l'intelligence artificielle, nous discuterons du rôle des physiciens en biologie, de la valeur qu'ils pourraient apporter, tant dans les approches théoriques qu'expérimentales. Pour comprendre cela, nous devrons nous plonger dans l'épistémologie, la nature de la méthode scientifique et de la modélisation, que nous réexaminerons ensuite à travers le prisme de la théorie de l'information. Nous discuterons également des opportunités que l'intelligence artificielle apporte à la biologie quantitative et partagerons des perspectives sur l'avenir de la collecte et de l'analyse des données.

1.1 Why should a physicist do biology ?

Why should a physicist do biology?

After earning my master's degree in statistical physics, I chose to pursue a PhD in biophysics. Nature and living things have fascinated me for as long as I can remember. Alongside this, I found myself deeply drawn to history, intrigued by the wisdom to be learned from past human experiences, preserved in books, art, and temples. It took me some time to discover the realm of genetics, the most hidden temple of all. In the sequences of nucleotides lie the legacy of our ancestry, which we will pass on to our descendants. These chains contain the entirety of our ancestors' history, a magnificent biological tale that provides insights into our most distant past, before we had the capabilities to write, speak, or even think. They epitomize the tragic dimension of our existence: our genes offer no escape. They are what they are, and we are what they are.

The lab of Hervé Turlier, which welcomed me, specializes in the mechanical modelling of early embryogenesis. The first divisions of the egg cell are crucial, as they pave the way for the rest of the organism's development. Understanding the mechanical aspects of these early divisions is key to understanding how genes execute the developmental program at the very beginning of our lives.

All of this is undoubtedly fascinating. But do biologists really need physicists? What value could I bring to their projects? Should I do experiments or focus on theory? To my surprise, everyone, from biologists to physicists, was eager to discuss with me and seemed happy to welcome me to their team. Biology is rapidly changing, partly due to the emergence of high-throughput and AI approaches. To evaluate their potential to revolutionize the way we make scientific discoveries, it is interesting to revisit the nature of science, biology and physics, and see how these emerging technologies could fit into this framework. Today, biophysics has become a buzzword. Everyone wants to do "modelling" and to be "quantitative". In my opinion, this trend is justified and here to stay, and I will try to provide an explanation for this in this introduction. The topic is infinitely broad, and this will thus be a rather eclectic discussion, where I will offer several complementary views that help understand how we might unravel the mysteries of life in the years to come. In the end, we are all biologists and we are all physicists: our goal is to uncover the truth about the deepest mysteries of our universe.

A PhD is a weird endeavor, when you really ponder it. It seems like a vestige of the old apprenticeship system, where you have a supervisor, who is himself a doctor, guiding you on the questions you should pursue, helping you get started, aiding in writing your articles - essentially mentoring you in every aspect of your research before you become yourself worthy of the title of doctor.

The current state of affairs: PhDs, Globalization and Innovation-driven growth

I come from the French higher education system, where students undergo a rigorous selection process to gain admission into prestigious institutions, called "Grandes Écoles". In this system, entering top-tier schools is way more challenging than obtaining a PhD, making it a more reliable indicator of one's cognitive abilities, motivation, and mental strenght. A PhD is more a testament to one's perseverance and expertise in a specific domain, as candidates spend 3 to 5 years developing a research project and acquiring new skills.

When I started my PhD, there was a trend among leading companies like Google and Facebook to make a PhD mandatory for research roles. This emphasis on the PhD, from a French point of view, does not make sense, but in a globalized world, the PhD remains the internationally recognized standard. An explanation for these trends is the increasing focus on innovation and Research and Development in modern economies. As software and the internet have lowered entry barriers to many industries, many businesses face a greater risk of being disrupted, making innovation a key driver of growth, and even survival in an ever-more competitive economic macro environnement. In this context, having completed a PhD provides valuable insights into the innovation and the research process, which has numerous aspects that **one must experience in their flesh to truly understand their relevance**. Here are some of the key lessons that I learned:

- The difficulty of inventing something new, the amount of time it takes and how little rewarding it is.
- The complete lack of correlation between the impact of your work, and the time invested in a project.
- The importance of knowing how to navigate the infinite search space for new ideas, by consulting experts, attending conferences, and exploring various resources.
- The instructive nature of both failures and successes: PhD programs provides a low-risk environment to learn from these experiences.
- The crucial role of execution.
- The need to minimize distractions that drain creative energy.
- The immense value of creative individuals, who may have unconventional personalities but are difficult to replace.

Innovation is a peculiar world, where individuals explore uncharted territories in search of valuable discoveries. There are two kinds of people: Problem-posers and problem-solvers. Formulating clear problems is not an activity specific to research, and a lot of people entirely devoted to engineering are brilliant problem-posers. However, an engineer that is a very good problem-solver can focus on solving problems, whereas in research you will be confronted every time to problem formulation. In engineering, you have practical goals, that can always be more or less summed up as: "Make the thing work". Money is the ultimate boussole. In research, you have no real boussole: You need to create your own flair, and constantly ask yourself: Is my question interesting ? Is it solvable ? With the advent of AI, the problem-posers are going to be immensely valuable, whereas the problem-solvers will most probably be replaced by algorithms.

I see the PhD as a space of freedom, a small-scale playground where one is allowed to develop this flair, and hone their problem-posing skills, along with their problem-solving ones. This process of venturing into the unknown without any guarantees can generate a lot of stress, but it will be what will make you stand by the crowd, and will probably be one of the only way to remain relevant in an AI world (The other obvious way being doing a manual job). However, my description of the innovative process is a bit too romantic. In reality, research is not a lonely exploration into uncharted territories. You are never alone, because at any time the intersection between interesting and tractable questions is quite small. Numerous other explorers are navigating along with you, using a variety of strategies. One method is go into

the fray, and try solving the problems everybody wants to solve (if everyone is trying to solve a problem, then this problem must be important). Of course, to win, you should better be faster and more efficient than others.

An alternative strategy consists of developing your own "pré-carré", tackling problems that you believe are interesting but that remained under the radar of other researchers. A frequent reason for this is that solving such problems requires a non-trivial set of skills and tools, an expertise that you will develop specifically, and that will make you uniquely positionned in your field. This expertise might be rooted in your prior knowledge or rely on unique resources you have access to. This expertise can manifest in various ways, such as having access to a specialized lab instrument or undisclosed software. It might also be more abstract, like having a robust network of collaborators who can provide access to exclusive data, or showcasing proficiency in specific experimental techniques or mathematical theories. However, if you've achieved this, others can too: If your pré-carré is very prolific, it will soon be invaded—there's no unchallengeable monopoly in expertise.

Research and engineering. What is "the true thing" ?

A good student in STEM[1] has to choose between two paths: concrete problems, that has to be solved to attain a short term goal, and abstract problems, with goals that are more fuzzy, that may be necessary to solve in the long term to advance our understanding of a particular topic of interest. The first end is often called Engineering, while the second is called Research. Research oriented on a shorter timescale is called Research and Development, by opposition with Basic Research. Today, all these things are quite hard to disentangle: in computer science, solving abstract problems without any practical goal can more and more result in very fast and successful solutions.

If a problem is interesting to be solved in the medium term, say 15 years, a company can be interested in investing resources into its solution. However, no company will invest in a 50 years horizon, and this is more and more true in our fast-paced finance-driven world[2]. Thus most of the problems will be exclusively studied in academia: The ones with no clear practical applications, where the only goal is to understand things. Or the ones with practical applications but that are so risky and badly defined that nobody really studies them.

I think choosing between one and the other is a matter of taste, and of personal goals. The culture between the two environnements, engineering and research, is far different, and will not suit the same personnalities. Engineering teams will tend to have a more rigid culture, with less room devoted to personal initiative, whereas research teams will be more flexible, letting everybody draw their own path. Of course, the culture will depend on the country and the particular environnement you are in, but the culture adapts to the kind of tasks a team is doing, so my considerations holds universally, no matter the peculiarities of any place and situations.

I also need to adress a very critized aspect of academic life, which is grant writing. I have to say that I never wrote any grant, except single pages documents here and there. I think people tend to criticize them because it is a kind of bureaucratic endeavour, with no clear

[1] Science, technology, engineering and mathematics.

[2] This can change, however, with the emergence of individuals that are so rich that they can invest their money as they want, on moonshot problems. There are more and more non-for-profit research institutes in the Bay Area.

output. I believe it is going to be better and better with time, because people will automatize the dull part with AI tools, leaving time for researchers to focus on idea development, study of litterature, and question formulation, which is the interesting and fruitful part of a grant. Making a grant will be about making a project, with clear references and objectives as bullet points, and letting AI tools write the grant for you. (On the other end, probably, people will use AI tools to get concise summary of the text of the grant, under the form of bullet points. Of course this is stupid and this is why we call it bureaucracy). So I am quite optimistic on this question, and the same holds to some extent for paper writing.

What does academia has to offer to people and to society?

In academia you have certain advantages that you may not encounter in the corporate world:

- Curiosity: You have more freedom to choose your research topic and follow your interests. You have a bigger variance of personal experiences than if you work for a company.
- Glory: While the recognition may be limited to the academic community, your work is personally attributed to you, your lab may bear your name. If you excel, you might receive awards and media attention, and in exceptional cases leave a lasting legacy.
- Belonging: You are part of a community, and you have the opportunity to be someone in this community, and shine.

However, pursuing a career in academia often requires some sacrifices:

- Work-life balance: Striking the perfect balance between work and personal life can be challenging in the world of research.
- Geographical stability: Opportunities may be scarce and require frequent relocation, limiting your choice of where to live. Mobility grants (such as Marie Curie) explicitly require you to change countries, which is both unnecessary and harming.
- Financial compensation: While money may not be the primary motivation of every individual, the financial rewards in academia are often modest, especially considering the extensive education required.
- Personal life: Raising a family can be more difficult due to demanding work schedules and frequent moves.
- Choice of colleagues: In the corporate world, engineers have the option to work in various environments and with diverse groups of people, providing flexibility and variety in professional relationships. You cannot do so in academia.

While this analysis might seem excessively negative towards academia, the thing is that, glory is cool[3], even in very moderate amounts. The freedom to choose a research topic, in my opinion, is vastly overrated: Curious people can develop a passion about everything. One should also note that many drawbacks (salary, choice of city) can be mitigated if you excel in your field. World renowned experts can work wherever they want, and top research institutes will be ready to pay a lot of money to attract them. This explains why some people still opt for academia despite seemingly better professional opportunities. However, I have to notice that researchers were generally better-off 30 years ago and we should collectively question why the situation has deteriorated. If it continues to do so, research will eventually fail to attract and

[3] Yes glory is cool.

retain top talents.

As I said before, the culture of the workplace you evolve in will depend greatly on the kind of endeavour you are pursuing. It seems obvious to me that not everyone is made for traditional corporate roles or management positions. Some people are more introverted, hard to manage and solitary. Though brilliant and productive, they cannot really fit in an organized team, and would be terribly unhappy working in a company. Contrary to company cultures that requires to fit into a certain mould, academia accept everybody as he is, and the diversity of personalities in research is very high. Research provides a setting that enable these individuals to flourish and contribute meaningfully to society by advancing knowledge in their respective fields. This is a great thing in academia.

Lastly, academic research serves as a vital conduit for preserving and disseminating knowledge and new methodologies, both in academia and in industry. Through research publications, conferences, and collaborations, researchers share their findings and insights with the wider academic community, fostering the continuous growth and refinement of collective knowledge. This process also promotes interdisciplinary collaborations, stimulating the development of innovative solutions to tackle complex problems. Academic research is useful, and plays an irreplaceable role in the ongoing expansion of human knowledge.

Is a PhD worth it ?

After having given all the good aspects of following a doctoral program, an inevitable question emerges: Is it not possible to acquire this knowledge while employed in a company, and potentially enjoying a superior remuneration? I firmly believe that people study for too long in our society, and that it hurts us as a whole, as people are not really able to start building their life before the end of their studies. My two grand-mothers had three children at the age I will earn my PhD, which makes them vastly more adult and accomplished than I am, objectively. Our societies are in a demographic winter, and it is a problem that we will need to solve if we do not want to be obliterated in years to come. The society would be better-off if talented people marry early and create families, instead of privileging doing endless studies just to add "PhD." at the end of their linkedin profile. The academic game is nice, but **having children is a greater achievement than publishing papers, even Nature or Science ones**. Thus, if you are close to 30 years old, I would say that starting a PhD is not the best decision you can make. **Some things are priceless and should not be sacrificed. A PhD is not one of them**. There is a life beyond studies, or at least there should be.

However, we should note that the current work scene appears bleak. The opportunity cost of embarking on something unconventional like a PhD, is practically at an all-time low. If you are young, I would still encourage you to start a PhD, preferably right after a Bachelor. I did mine in 3 years. 4 years seems like a good duration. 5 years looks reasonable, and 6 years is too much.

There is a dichotomy to be made between hard and simple problems, or, in corporate terms, between labour-intensive and high value-added jobs. **If you are doing a labour-intensive job, solving problems many other people could solve, to be more successful you do not need to be the most clever person, but just the most efficient, focus, and hardworking one. This is a bad idea**. The perfect example is consulting. In consulting, you solve problems that all your collegues are able to solve as well. Thus, the only way

to make a difference is by working long office ours at a high pace, sacrificing every other aspects of your life. Engineering and research can allow you to extract yourself from this fray, if you are particularly gifted you can differentiate yourself by your vision rather than by your hardworking capabilities. Given the global competition that characterizes today's world, it becomes increasingly essential to differentiate yourself from the crowd. Among other things, a successful PhD can help you provide you this differentiation, and land into these high value-added jobs.

Why we should stop writing theses

The system of doctoral dissertation is an heritage of the medieval system. It originates from the early middle ages, in the 12th century in the university of Bologna in Italy and the University of Paris in France. The PhD thesis became a standard in the 19th century. Two precursors were the Humboldt University of Berlin, and Yale University in the United States, which introduced the PhD degrees, that required writing a dissertation based on original research. Over time, the nature of the thesis has evolved from a demonstration of knowledge in the field to a requirement to produce original research. It was viewed as a way for the candidate to contribute new ideas and knowledge to their area of study.

However, today every worthy new idea ends up in articles. People want to publish and share results as soon as they are established, not least because of the global competition, that makes people hurry to be the first ones to publish results, for fear from being scooped. Nobody wants to wait till the end of the PhD to share results to the community. Now, we are in a next stage, where we cannot even bear the time needed to peer-review papers. Thus, we put preprints on servers, and do a thread on Twitter, and can start immediately to talk about that with the other persons of the academic community.

How many doctoral dissertations did I read during my PhD? Zero. I downloaded some, but never took the time to delve into them. This is the biggest argument for why I think it is useless. Across history, it seems that the utility of the doctoral dissertations has decreased progressively with time, attaining something close to zero today. When I want an insight, I want it fast. I prefer to do a quick research on google, ask a few questions to chatGPT, watch a talk on YouTube, or spend the morning to read 2/3 articles and reviews in depth if I want to learn something new. We just live in a world that is just too fast-paced for dissertations. I think this is a huge argument that has to be made, that was already valid 10 years ago and that is now more relevant than ever. One thing will just put the final nail on the coffin of the uselessness of writing a dissertation: Large language models. If an algorithm writes better than you, then you it would be stupid to not let him write things itself instead of loosing your time writing something that will be less good. I clearly believe that you are reading one of the last PhD thesis that is not written completely by a language model. World is going to get weird. People in charge need to rethink this system completely, because it is obsolete since a lot of time and it is getting more and more absurd.

1.2 A bit of Epistemology

1.2.1 Early-life crisis

After having sucessfully passed my competitive exams, at 20, I recall thinking that I had studied a lot of math, and that it allowed me to understand many things. However, I felt that I needed to explore other schools of thought to gain a deeper understanding of reality. Of

course, my primary focus remained the sciences - I chose to major in quantum and statistical physics - but I also had a lot of free time to read all sorts of books, and had some courses on humanities included in my cursus.

The first course I took was called "Philosophy of sciences." I remember thinking, "Can we please study something else? I already spend the rest of my day studying science!" However, looking back now, I believe it was one of the most significant courses in my curriculum. It brought my attention to the very nature of what I was studying. While it's not unusual to see accomplished scientists delving into epistemology and the philosophy of sciences, I strongly believe that this should be more of a norm than an exception. Every scientist, I assert, should possess a deep understanding of the scientific method[4]. Pursuing a career in science makes more sense if it is accompanied by curiosity about its meaning, its history, and its scope. The intricacies of the scientific method, what science means, and the degree of certainty of our work are crucial questions for everybody who consider himself a scientist.

Maintaining a balance between specialization and overspecialization is always a challenge. It's necessary to specialize, to gain genuine expertise on a specific topic, to understand its stakes, problems, and controversies, and to formulate potential solutions in the hope of making a significant impact. However, in doing so, it becomes difficult to set aside time to maintain a broad enough perspective, to pay attention to topics that may seem only weakly related but could hold unforeseen relevance to your immediate problems. As Pierre-Gilles de Gennes once said, science should not be about "knowing more and more about less and less." Two things are of importance: having a vision and possessing the technical skills to execute that vision. But, the vision is the most important, especially in an era where computing, artificial intelligence, and robotics will be more and more able to execute clearly expressed visions.

Public trust in scientists is lacking, a sentiment that I understand, unfortunately. I did my PhD during the pathetic "The Lancet" incident, where a top tier journal published a blantly fake study in order to promote pharmaceutical interest. Regrettably, many scientists portray a negative image in two distinct ways: Firstly, they seem overly confident in the value of their work. Much like the media, scientists possess a finite stock of public attention and credibility. Every time they make false assertions, they diminish some of it. Scientists often find themselves in the peculiar position of being the best informed individuals in the world on a particular topic, but still likely to be incorrect, simply because the questions we ask them are incredibly challenging. We should be careful not to be too sure about the truths that our science provides us. After all, as Richard Feynman said, "Science is the organized skepticism in the reliability of expert opinion".

In spite of all that, scientific knowledge progresses at an incredible pace. Save for exceptional individuals, individual humans have probably not a clear idea of where they are going. Most people have a narrow vision, follow only their current interests, are influenced only by the trends of their colleagues and friends, by a few serendipitous encounters, and from this organized chaos, a solid comprehension of many aspects of our world emerges. Even if the current way of doing things is quite "functional" in this sense, it is full of frictions, and I believe we could do better if we strove for a clearer understanding of the fundamental purpose of our discipline, the potential societal applications of our work and the methods by which we establish facts and the truth value of these facts.

[4] What a strong assertion !

With these considerations in mind, I will provide a brief introduction to biophysics and quantitative biology.

1.2.2 Scientific methods and The Scientific Method

I believe there is an objective reality and this is an act of faith

People tend, today, to oppose science and faith. I do not, and part of the reasons for not opposing both is that I am conscious that everybody, when reasoning, do acts of faith all the time. I believe that there is an objective reality, that is there independently of me. I perceive things subjectively, from my senses, however. Ultimately, I cannot prove that I am not trumped by my senses and by my brain. Identifying what I perceive as the objective reality is an act of faith, that is the basis of the scientific inquiry: If the reality is objective, it can be observed, described and explained. How can we differentiate between reasonable and unreasonable acts of faith in life ?

One principle that supports an act of faith is Ockham's razor, a philosophical guideline stating that the simplest explanation for a phenomenon is most likely the correct one. As an example, I believe, as an act of faith, that I am not the principal character of a video game full of realistic NPCs (Non playing characters) only because I have an invariance by permutation symmetry assumption: I assume that as people *kind of* look like me and act like me[5], they are real human with real feelings as me and not some cleverly engineered Trompe-l'oeil. This is obvious, but this is still an act of faith.

The belief in an objective reality can be traced back to the beginnings of human civilization, when ancient philosophers and early scientists sought to understand the natural world and its underlying principles. It is ubiquitous because it is quite natural: Occam's razor tells me that if all our senses and experiences is consistent with the fact that there are perceptions of an underlying reality that is independent of me, it is natural to consider the hypothesis that there is indeed an objective reality. Assuming that there is an objective reality, let's see how people strove to make sense of it.

The history of *do your best* accross the ages

Science is simply explaining the nature and the rules of this objective reality. The process of understanding and describing it has evolved throughout history, with various approaches being developed to systematically investigate and explain natural phenomena. What method should we use? The historical approach has always been the following one: ***do your best!*** In ancient times, greek philosophers, starting with Aristotle and Plato discovered the rules of logical reasoning, they started to establish truth by writing dissertations, arguing to convince others of their points. They did, in a word, philosophy. They naturally applied this method to understand the natural world, giving birth to "natural philosophy", seeking to uncover the principles governing reality through observation, contemplation, and logical reasoning. Natural philosophy laid the groundwork for the development of modern science, as it encouraged the exploration of objective reality and the search for underlying patterns and principles.

Then, little by little, this approach solidified itself around one fundamental principle: **falsifiability**. This is a bit a renunciation, that stems from the fact that it is not possible to establish the truth directly, from the first try. Reality is complex, and its modelling need to be refined iteratively, through a trial-and-error process, during which we realize a never-ending series of

[5] Though a lot of people act like NPCs I have to say.

steps to formulate, test, and refine hypotheses. This is the formalized, pragmatic version of the historical approach to explain reality: *do your best*. The key steps of the scientific method include:

- Observation and description of phenomena
- Formulation of hypotheses
- Experimental design and data collection
- Data analysis and interpretation
- Replication and refinement of knowledge

By following these steps, scientists can develop and refine theories that accurately describe objective reality, while remaining open to the possibility of revising or discarding these theories in light of new evidence. In this framework, mathematical modeling is very interesting because it allows to perform quantitative predictions. We can directly assess the validity of our model, i.e our description of the phenomena, by making quantified predictions, that we can confront to our data, obtained from experiments. However, it is neither sufficient nor necessary to use mathematics. Some disciplines use mathematics to make predictions that are not falsifiable, and thus are not sciences. Falsifiability, and not numerical quantification, is the root of scientific method.

Let's note immediately that in order to refine iteratively our hypotheses, they need to be falsifiable. This can not be applied to everything, and characterizes the disciplines that we can call natural sciences. Scientific method is a very effective ways to establish accurate models of reality. However, it is not the only way to describe it, and there are a lot of other methodologies, adapted to each domains, which corresponds, in their own manner, to the initial motto of *do your best*.

Certain fields diverge from the scientific method either because their subject matter is not amenable to empirical investigation or because they use distinct methodologies to address their inquiries. For instance mathematicians derive and prove theorems through logical deductions from established axioms or principles. Philosophy, does the same, at its humble level, and employs critical thinking and logical analysis rather than empirical observation or experimentation, in a more loose way than mathematics, but to tackle abstract questions that are more interesting for us humans.

Another good example is given by History. While it can incorporate empirical evidence in the form of primary and secondary sources, it follows a methodology distinct from the scientific method. Historians focus on understanding and interpreting past events, societies, and cultures, often engaging in qualitative analysis and narrative construction. The scientific method's emphasis on controlled experimentation and reproducibility is not applicable to historical inquiry, as past events can't be directly observed or replicated. Although these disciplines do not strictly adhere to the scientific method, they have rigorous methodologies and make significant contributions to human knowledge. The variations in methodologies reflect the diverse nature of the questions and subjects these disciplines aim to address. Each field offers a unique approach to comprehend the complexities of reality in the best possible way.

Science gives us a self-consistent network of models

A question arise when employing the scientific method: When we develop a model A to make predictions about a physical quantity, we ultimately compare these predictions to experimental results. However, mathematically speaking, measuring a physical quantity involves fitting the parameters of another model, called model B. In this scenario, we are comparing the outcomes of two different models, not confronting a model with reality itself. Does this process entail circular reasoning, and is it contradictory?

To address this concern, we must first acknowledge that all measurements and observations are influenced by our comprehension of the experimental apparatus and the physical principles governing it. Developing a model to describe the experiment (model B) requires making certain assumptions and approximations based on well-established principles and previous experimental findings. This provides us with confidence in the reliability of model B. Moreover, when comparing the predictions of model A and model B, we are not simply attempting to prove one model right and the other wrong. Rather, we are examining the degree of agreement between the two models. If their predictions align well, it suggests that model A is a good representation of the physical phenomenon under study, given the accuracy and reliability of model B. If a significant discrepancy exists between the predictions, it could indicate that either model A, model B, or both are incorrect or incomplete. In such cases, further investigation is needed to identify the source of the discrepancy and improve the models.

In essence, the comparison of models is not circular reasoning but a way to continuously refine and improve our understanding of the physical world. By comparing model A with model B, we can pinpoint areas where our knowledge may be lacking, which in turn leads to the development of new theories or experimental techniques. This iterative process of testing and refining models lies at the core of the scientific method, helping to construct a robust body of knowledge grounded in empirical evidence and theoretical reasoning. In conclusion, while comparing theoretical predictions with experimental results involves contrasting two models, it is not inherently circular reasoning. The whole field of natural sciences is building a self-consistent and interconnected network of models that describes a plethora of observed natural phenomena. The ultimate goal of science is to develop a comprehensive understanding of the natural world, and this network of consistent interconnected models serve as a foundation for that understanding.

1.2.3 Physics, Modeling and Biology

What is physics ?

Physics is a natural science that seeks to understand and describe the fundamental laws governing the behavior of matter and energy in the universe. It encompasses a wide range of phenomena, from subatomic particles to the largest structures in the cosmos, and includes both theoretical and experimental approaches. In physics, there is a distinction between fundamental (or microscopic) and mesoscopic/macroscopic approaches. Fundamental physics aims to understand the most basic building blocks of matter and their interactions, such as particles described by quantum mechanics and their governing forces. Meso/macro approaches, on the other hand, focus on large-scale phenomena that emerge from the collective behavior of many particles. Examples of meso/macro approaches include fluid mechanics, which studies the motion of fluids, and thermodynamics, which investigates the relationships between heat, work, and energy in systems.

Statistical physics, the branch that I love the most, is an area of physics that attempts to bridge the gap between fundamental and mesoscopic/macroscopic approaches by providing a framework for understanding the behavior of large ensembles of particles. It combines the microscopic laws of motion with statistical methods to derive the macroscopic properties of systems, such as temperature, pressure, and entropy. This approach has been particularly useful in explaining the properties of complex systems, including those found in biophysics, where the collective behavior of numerous molecules gives rise to emergent phenomena.

Biophysics is a bit an ill-defined topic. It is the study of the physics of biological objects. As these objects can range from a few nanometers (proteins, DNA, RNA) to a few meters (organisms), there is a wide array of physics associated with them. Biophysics applies concepts from classical and quantum mechanics, thermodynamics, statistical physics, and other areas of physics to the study of biological molecules, cells, tissues, and organisms. It occupies a unique position within the broader field of physics, as it applies physical principles to understand the behavior of biological systems. Biophysics leverages insights from both fundamental and mesoscopic/macroscopic approaches, often drawing on statistical physics to describe the complex interactions and emergent phenomena that characterize life at various scales.

Quantitative biology and biophysics are closely related fields with some key differences. Quantitative biology primarily focuses on the application of mathematical and computational tools to analyze and model biological data, striving to provide a quantitative description of specific biological phenomena. Biophysics, on the other hand, aims to uncover the fundamental physical principles governing biological systems and processes, often seeking common rules and patterns between various biological phenomena. While both fields share the goal of achieving a deeper, quantitative understanding of biological phenomena, their approaches and primary objectives differ, with quantitative biology emphasizing data analysis and modeling, and biophysics focusing on understanding the underlying physical principles.

What is biology ?

Biology, the study of living things, covers an extensive spectrum of sub-disciplines, ranging from the microscopic molecular interactions to macroscopic systems. In biology, the fundamental units of study are the basic building blocks of life - cells, genes, and proteins - and their interactions. These microscopic components give rise to diverse and sophisticated forms of life, from single-celled organisms to intricate multicellular organisms such as plants and animals. Biologists both investigate these building blocks and strive to comprehend the emergent properties arising from the collective behavior of biological components at various scales.

Because of its richness, it is hard to class biology as a whole into the realm of *soft* or *hard* science. Biochemistry disciplines such as structural biology and genetics lead to reproducible experiments, whereas in cell biology and systems biology, researchers deal with highly variable systems, that makes the experiments less reproducible, and thus less falsifiable. Therefore, they adopt another approach, utilizing various complementary methods to validate their theories, falling into the realm of *soft* science. Yet, all across biology's diverse spectrum, one can observe a transition towards more rigorous scientific methodologies, in order to have a more reliable, robust understanding of life. This is made possible by the use of cutting-edge technologies such as genomics, proteomics, and advanced imaging techniques, as well as methods from other disciplines like physics, chemistry, and computer science. These approaches, as a whole, are often described under the generic terms of quantitative biology of biophysics.

Models and mathematics

A model, within the framework of the scientific method, is a simplified representation of a complex system or phenomenon. This description can be of several natures, symbolic, relational, but it will always, eventually, be represented by mathematical objects. The model offer an unambiguous representation of the phenomena under investigation, allowing to explain which quantities intervene, and the nature of the relation between each quantities. Formalizing these relations mathematically, through the use of equations, allow to make predictions, and thus make these models falsifiable, making them a perfect building blocks of the scientific method.

There are numerous way to describe mathematically physical systems. Such systems can be deterministic or stochastic, static or evolving with time. Popular methods include systems of differential equations, like Partial Differential Equations (PDEs) or Ordinary Differential Equations (ODEs). Classical equations, such as the heat equation or Navier-Stokes equations, form a part of the curriculum for every physics and mathematics student due to their widespread relevance.

A physicist who is familiar with these classical equations can bring valuable insights to biological data. By "knowing its classics", it can make hypotheses about the physical models governing the system early on. This early hypothesis formulation is vital as it guides subsequent investigations and the design of experiments aimed at validating or refuting these hypotheses. However, the creation of a model is only the first step. Physicists have also a valuable expertise in mathematics, computer science, that is useful to analyze the data further. A fundamental methodological difficulty encountered by the physicist is that many equation systems could potentially describe a given system, necessitating extensive testing of the model to confirm its accuracy. As Von Neumann famously stated, "With four parameters I can fit an elephant, and with five I can make him wiggle his trunk," indicating the importance of thorough testing and validation of models, as redundancy in validations reduces the chance of overlooking a key element in the model.

Model creation has thus transformed natural sciences into data-centric disciplines. This data-centric approach is especially evident in biology, where the inherent variability demands extensive data for validation. To affirm the significance of their data and not just consider it an artifact of experimental variability, researchers often rely on statistical tests. Consequently, laboratories have specialized in various data acquisition methods, such as sequencing, omics, mass spectrometry, and imaging (microscopy), to answer fundamental questions.

But is it reasonable to believe our systems will always follow simple models?

Why mesoscopic and macroscopic systems can or cannot be described by simple equations

The reason for which one can hope to describe complex systems using equations is quite straightforward: our universe's smallest building blocks, atoms, electrons, and other elementary particles, follow mathematical equations in a manner that can seem almost magical (cite the unreasonable effectiveness...). These elementary components, in turn, constitute the larger mesoscopic objects that we want to study, be it macromolecules such as proteins, DNA, lipids, or bigger objects. This means that the behavior of these larger systems is described by the integration of the behaviors of their individual components. Therefore, if the building blocks follow mathematical equations, the systems composed of these blocks will also naturally follow mathematical equations. However, these equations can range from simple to incredibly

complex.

In some cases, the macroscopic behavior of a system can be described easily. This is the case in Thermodynamics, where the state of the system can be described with a finite number of observables such as heat, work, temperature, or entropy, related with themselves with three laws of thermodynamics. The simplicity of this coarse-grained behavior can be explained by the kinetic theory of gasses, which explain successfully the macroscopic properties of gases in terms of the microscopic behaviors of their molecules. The theory was substantially developed by James Clerk Maxwell and Ludwig Boltzmann in the latter half of the 19th century. They assumed that a gas consists of a large number of tiny particles in constant motion, colliding with each other and the walls of their container. The movement, tough deterministic, exhibit chaotic dynamics that has the same statistical properties as a random motion, and can thus be considered as such. From this assumption, they were able to derive the ideal gas law and explain macroscopic phenomena such as pressure and temperature.

However, in other cases there is no way to predict the behavior of a complex systems. Sometimes, non-linear dynamics and chaotic systems do not lead to a statistical distribution that can be easily averaged as in the case of perfect gasses. This brings us to the concept of computational irreducibility.

1.2.4 Complex Systems

Reductionnism and Computational Irreducibility

First introduced by Stephen Wolfram in his work on cellular automata, Computational irreducibility, is a concept that describes systems for which there are no shortcuts to predicting future states. To know the system's state at time t, you have to simulate all the steps from the initial state up to time t, regardless of the simplicity or complexity of the initial conditions. The behavior of these systems cannot be 'fast-forwarded' using a mathematical formula or algorithm. Instead, one has to follow the system's evolution step-by-step to understand its behavior.

Computational irreducibility is common in complex systems, and should be encountered in many biological systems, where a multitude of factors interact in nonlinear and often unpredictable ways. Biophysics seeks to find simple and general principles behind the complexity of biological systems, by capturing their essential features into simplified models. When a system is computationally irreducible, it means it cannot be described into a simpler form without losing essential information about its behavior. Therefore, any attempt to create a simplified or 'compressed' model will inevitably fall short in capturing the intricacies of the system's behavior.

On the other hand, quantitative biology aims to capture the full complexity of biological systems in their models, often leading to models that are as complex as the systems they represent. While these models may be more successful in capturing the behavior of computationally irreducible systems, they suffer from a lack of generalizability. The models become too tied to the specific system and conditions they were designed for and may fail to provide insights that can be applied more broadly.

Therefore, the concept of computational irreducibility adds an extra layer of challenge to both fields. It limits the ability of biophysics to find general laws, and at the same time, it reinforces the complexity and specificity that quantitative biology tries to capture. Un-

derstanding computational irreducibility, thus, provides important insight into the nature of biological systems and the limitations we face when trying to model them.

Can we still study these complex systems ? Is the concept of falsifiability doomed? Not entirely. While the overall system may resist simple falsifiable hypotheses due to its irreducible complexity, we can still apply falsifiability to the individual building blocks. Despite the complexity of the overall system, rigorous analysis requires a reductionist approach. By understanding the individual components and the equations governing them, we can gain insight into the integrated behavior of the whole system, no matter how complicated it may appear. We will discuss later about reductionnism. First, we will discuss about the cases where a complex system can be simplified.

Principled ways to simplify a complex system

When confronted to a complex system composed of thousands of elements, one might anticipate an equally complex description of their behavior. However, intriguingly, many such systems' collective behavior can be described by relatively simple macroscopic equations. Biological systems are characterized by their stability and their reproducibility. In a way or another, the extreme reproducibility of our development means that all the potential chaotic dynamics that would lead to sensitivity to noise or to initial conditions are selected negatively through evolution. Each sensitive developmental and biological steps are buffered efficiently, which gives good reason to believe that biological systems can be simplified and coarse grained, and will not be subject to computational irreducibility, and could be simplified. In statistical physics, this simplification is made possible due to several key factors: symmetry, conservation laws, emergence of dominant interactions, separation of scales with coarse graining, and statistical averaging.

Symmetry plays a pivotal role in simplifying our understanding of the system. Often, the intricate microscopic interactions within these systems exhibit certain spatial, temporal, or fundamental property-related symmetries. From symmetries results conservation laws, such as those of energy, momentum, and charge, that constrain the possible outcomes of particle interactions, reducing the complexity of the governing equations.

Moreover, many systems display dominant interactions that dictate their macroscopic behavior. When these interactions significantly outweigh others, they effectively characterize the system, allowing for a streamlined description. For instance, the macroscopic behavior of a gas can often be described by the ideal gas law, largely attributed to dominant particle collision interactions.

The separation of scales and the practice of coarse-graining also serve as critical facilitators of simplification. When a clear divide exists between the microscopic and macroscopic phenomena, the system's behavior can often be streamlined using coarse-graining techniques. By concentrating on macroscopic properties, the intricate details of microscopic interactions can often be overlooked, leading to simpler governing equations.

Statistical averaging is another powerful tool in simplifying system behavior. With an increase in particle numbers, individual behavioral fluctuations tend to cancel each other out. This cancellation results in the emergence of stable, predictable macroscopic properties that can be described by far simpler equations.

Though these principles offer considerable simplification, it's crucial to remember they may not always apply. Complex systems can indeed exhibit complex behavior, resistant to these simplifying techniques. However, in a significant range of scenarios, they allow us to describe systems of thousands of protons and electrons using remarkably simple macroscopic equations. Thus, the science of statistical physics and complex systems, with its focus on emergence, serves as an integrating bridge, amalgamating reductionist insights to simplify and understand complex system behavior.

A fascinating thing is that many model-design principles that we explained in this paragraph are at the basis of data-compression algorithms, showing the great unity in the mathematical nature of our reality. Symmetries directly relates to Fourier Transform, and the closely related Discrete Cosine Transform, at the basis of the JPEG image-compression algorithm [1,2]. Besides, coarse graining shares deep analogies with the wavelet transform [3], that is a fundamental building block of many state of the art compression algorithms. In the next section, I will do a short introduction of information theory, showing how this perspective allows to draw a clear link between Biophysics and Quantitative Biology, and showing how it can shed light over future important advances in biology.

1.3 Insights from Data science and Information Theory

We will finish this discussion about epistemology by delving into information theory, motivated by its central role in Erwin Schrodinger's definition of life [4], where the notion of entropy takes center stage. This exploration will help us clarify what we are looking for when studying biological systems, by giving a definition of a model, and to provide a framework for distinguishing good models from poor ones. By examining these concepts, we will clarify the distinction between biophysics and quantitative biology, underscoring their respective roles in our understanding of life. **Moreover, these reflections bear significant implications for our ongoing efforts to construct intelligent systems capable of replicating human-like scientific reasoning and inquiry, which is the most important practical progress in science that we will witness in our lifetime, unlocking many others**.

1.3.1 Information and Entropy

The field of information theory is deeply intertwined with the concept of entropy, a concept that originates from thermodynamics and statistical physics. Information theory was born in the mid-20th century, when Claude Shannon, a researcher at Bell Labs, sought to define a quantitative measure of information. In his seminal 1948 paper, "A Mathematical Theory of Communication" [5], he introduced the concept of entropy as a measure of the amount of uncertainty or randomness in a set of data. This dataset can be described by an ensemble $\mathcal{X}$, called the dictionnary, and a distribution of probability $p : \mathcal{X} \mapsto [0, 1]$. Then Shannon's entropy is defined as:

$$S = \sum_{x \in \mathcal{X}} p(x) \log(p(x)) \tag{1.1}$$

Borrowing the term "entropy" from thermodynamics, Shannon's entropy was a way of measuring the average information content of a message, that is, the degree of surprise or the amount of new information a particular message provides. A highly predictable message carries little new information and thus has low entropy, while a less predictable message carries more new information and thus has higher entropy. Shannon's entropy shares many formal properties with the concept of entropy in statistical physics, especially as it was developed by Ludwig Boltzmann [6]. In Boltzmann's interpretation, the entropy of a physical system is a measure of the number of possible microscopic states that could give rise to the observed macroscopic state of the system. A more disordered system has more possible microscopic configurations, and thus higher entropy, while a more ordered system has fewer possible configurations, and lower entropy.

The parallel between the concepts translates into a parallel between the mathematical form of Shannon's entropy and Boltzmann's entropy formula. Both take the form of a sum (or integral) over states, where each term is the product of the probability of a state and the logarithm of that probability. This logarithmic dependence signifies the multiplicative accumulation of possibilities in both systems - combinations of messages in information theory, and arrangements of particles in statistical physics.

Shannon himself noted the analogy between his measure of information and thermodynamic entropy, even consulting with the eminent physicist John von Neumann about the appropriateness of the term "entropy" for his measure. Von Neumann reportedly agreed with the analogy, stating that the two concepts have a similar uncertainty property. The link between

information theory and statistical physics was deepened by Edwin Jaynes in the 1950s [7], who advocated for the principle of maximum entropy as a method of statistical inference. He proposed that, when choosing among probability distributions that represent our state of knowledge about a system, we should select the distribution that maximizes entropy, subject to whatever constraints are imposed by the information we have [8].

Information theory and statistical physics share a rich historical and conceptual connection, centered around the fundamental concept of entropy. We will now explore some insights provided by this conceptual framework.

1.3.2 Information theory, definitions and models

Aristotle, Aquinas and Shannon

A more practical introduction to the theory of information can be explored with the concept of definition. The traditional view of the definition of Aristotle and Aquinas can be revisited very elegantly using information theory.

According to Aristotle, a definition is a phrase that signifies the essence of a thing. It consists of the genus (or category) to which a thing belongs and the differentia, which distinguishes the thing from all other members of the genus. In his works, such as *The Categories* [9] and *Posterior Analytics* [10], he has written extensively on the nature and roles of definitions. For Aristotle, the goal of a definition is to provide insight into a thing's nature and to categorize it according to its natural place in the hierarchy of things. In the Aristotelian approach, a definition is intended to be a reflection of the objective nature of the entity in question. For example, a human being might be defined as a 'rational animal', with 'animal' as the genus and 'rational' as the differentia, distinguishing humans from other animals.

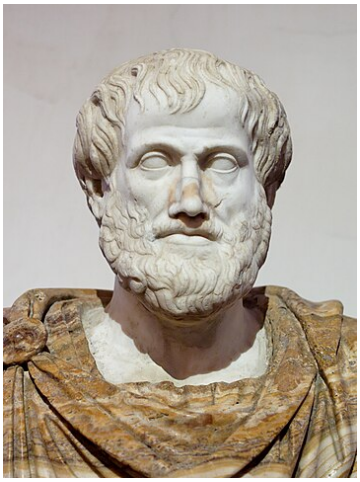


Figure 1.1: Aristotle, among other things, was a pionner in embryology.

In his work *De Ente et Essentia* [11](On Being and Essence), Saint Thomas Aquinas expanded upon Aristotle's understanding of definitions and discussed the concept of a thing's "essence" or "quiddity". For Aquinas, a definition was intended to describe the essence of a thing, which is its fundamental nature or "what it is." The essence of a thing comprises those characteristics that make the thing what it is and without which it would not be that thing. In other words, it is the necessary attributes or properties of a thing, which cannot be removed without

making the thing cease to be what it is.

Figure 1.2: Saint Thomas Aquinas was very cultivated and well advised, but never heard about entropy and information theory. Painting from Jacopo Vignali.

Aquinas made a clear distinction between a thing's essence and what he termed "accidents." In Scholastic philosophy, from which Aquinas drew, accidents are those properties or characteristics of a thing that are not essential to its nature. They are non-essential attributes that do not change the identity of the being even if they change. For example, a human might change in size, age, or color, but still remain human. Those are accidental properties, not essential ones. So, according to Aquinas, a definition seeks to capture the essence of a thing without the accidents. It does not include any accidental or non-essential characteristics that might vary from one individual to another or change over time within the same individual. By focusing on the essence, a definition attempts to describe what is most fundamental and universal about a thing, thereby giving us a deep understanding of its nature.

It's worth noting that Aquinas also distinguished between "essential accidents" which are those closely related to the essence (such as the ability to laugh in humans) and "pure accidents" which are entirely arbitrary and not connected to the essence (such as hair color). Despite this, for a strict definition, only the essence of the thing, without any accidents, is considered. **The notion of accident introduced by Saint Thomas Aquinas is exactly the notion of uncertainty, or randomness, of Shannon**. We have a set of data that takes values into a dictionnary $\mathcal{X}$, and that follows a certain distribution p, and giving a definition is finding a way to summarize this distribution in a relevant manner.

Some examples brought from biology

Last paragraph was a quite abstract discourse, that we will revisit using concrete examples. First, for a precise physical object, the definition can be extremely clear and unambiguous. For example, here is the definition of DNA on the Thesaurus:

- *Deoxyribonucleic acid*: an extremely long macromolecule that is the main component of chromosomes and is the material that transfers genetic characteristics in all life forms, constructed of two nucleotide strands coiled around each other in a ladderlike arrangement with the sidepieces composed of alternating phosphate and deoxyribose units and the rungs composed of the purine and pyrimidine bases adenine, guanine, cytosine, and thymine: the genetic information of DNA is encoded in the sequence of the bases and is

transcribed as the strands unwind and replicate.Compare base pair, gene, genetic code, RNA.

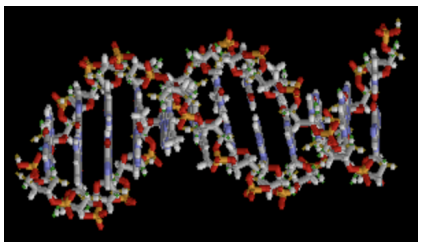


Figure 1.3: Desoxyribonucleic acid can be defined very clearly and without ambiguity.

However this is not the case with every concepts. When you open a dictionnary, you will usually find several definitions for a given word. Let's take the word *Life*. The Thesaurus gives not less than 27 definitions. Let's consider the 5 first: 1-the condition that distinguishes organisms from inorganic objects and dead organisms, being manifested by growth through metabolism, reproduction, and the power of adaptation to environment through changes originating internally. 2- the sum of the distinguishing phenomena of organisms, especially metabolism, growth, reproduction, and adaptation to environment. 3- the animate existence or period of animate existence of an individual. 4- a corresponding state, existence, or principle of existence conceived of as belonging to the soul. 5- the general or universal condition of human existence. etc...

The crucial point is that any of these 5 definitions kind of "do the job", but if you consider all these definitions altogether, you will end up with a way-better notion of what is life than with any of these 5 definitions. Life is a blurry concept, that encompasses many notions. As with Aquinas view, we see that **a definition is a compression of information**. If you compress the information to a greater extent, you will loose in precision or generality: you will have a definition that encompasses a smaller number of realities, or that characterises them less precisely.

A first reason for this is polysemy: a word can have several meanings, that are loosely related which each other. A second reason is more fundamental, and related to the information content. The efficiency of the compression of some data depends on the information contained in this data, of its regularities, of its symmetries, independently of the talent of the person that is in charge of giving a definition. It is easy to give an exact definition of *planet Earth*, *the hydrogen atom*,or of *a chair*, because they describe objective realities of the world. However, the concept that I took, *life*, is intrinsicly difficult to define, because this concept contains much more information. As do *love*, *passion*, or *happiness*, which are all tied to personal experiences of life, that varies for each individuals.

In his book "What is Life?" [4], Erwin Schrödinger attempts to provide a physicist's perspective on the nature of life. Schrödinger proposes that life can be described as a system that maintains order and resists the natural tendency towards disorder, or entropy. He introduces the concept of "negative entropy" or "negentropy" as a characteristic of living systems, which are capable of extracting energy from their surroundings to maintain their order. While Schrödinger's definition of life can encompass many living organisms, it may not include certain life-like forms, including viruses, or self replicating entities such as RNAs, cristalline structures or even computer programs.

Schrödinger's definition of life, while influential, is not universally accepted, and modern definitions of life have evolved to consider additional characteristics, such as reproduction, metabolism, and cellular organization. Biologists argue about the definition of life, but in the end, nobody is wrong, and everybody is right. A definition compresses the information, which means that all the information that we have at the beginning will have to be summed up in a few words, or in a book. You will have to squeeze the richness of animals, vegetals, mushrooms, parasites, archeas, bacteria into a few words. A definition is a model of a concept, and as in every model, you have to ask yourself Aquinas eternal question: **what part of the information is essential, and what part is not ?**

1.3.3 Biophysics versus Quantitative Biology

Previously, I drew a distinction between biophysics and quantitative biology. These terms are used in a very loose manner, interchangeably, and I will try to disolve some imprecisions of language by showing that though they are interrelated disciplines, each present unique goals that often generate a fundamental divide. Biophysics pursues the discovery of unifying principles that govern biological phenomena, striving to unify diverse observations into a single theoretical framework. In contrast, quantitative biology is committed to delivering a precise, faithful, and quantified depiction of biological systems, with an emphasis on accuracy and specificity.

This division becomes can be easily apprehended through the lens of model complexity. Biophysics favors compression, formulating simpler models that encapsulate the most pivotal features of the data. Conversely, quantitative biology gravitates towards elaborate representation, expanding the model until it mirrors the observed data with utmost fidelity. The tradeoff between generalization and precision shows that the notion of an 'optimal' model becomes subjective. *"All models are wrong, but some are useful"* [12].

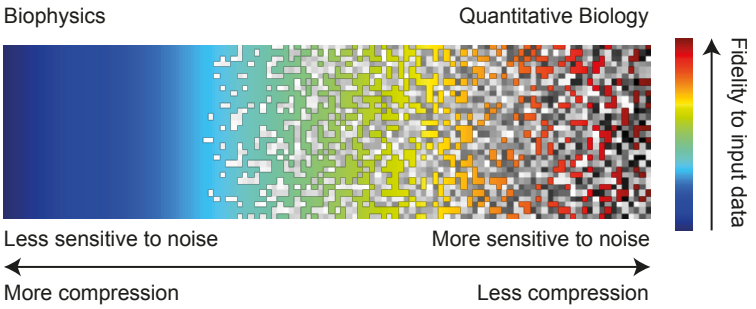


Figure 1.4: Noise limits the horizon of quantitative biology, stressing the need for high-throughput experiments to access to a higher fidelity.

I said that choosing between biophysics and quantitative biology is a matter of choice, but what could draw this choice ? Humans naturally wants to convert the complex realities that they are observing into an understandable simplification, the famous textbook picture, namely Biophysics (A professor once told me *"With a good PhD, a student has the opportunity to change one figure of a textbook of biology. This is amazing !"*). Another cultural aspect is that of mathematically litterate people who study biology are physicists that comes from related fields such as soft-matter or optics, that are naturally enclined to seek for general

principles.

However, there are also technical reasons that drive this choice. When it comes to modeling a biological phenomenon, answering the question, *"What level of compression is too much?"* is challenging. An easier question to tackle is, *"What level of compression is not enough?"* Here, we have a clear metric: "How well does the model reproduce your data?" As you add parameters to your model, its ability to reconstruct your data will improve. However, at some point, this metric will plateau, indicating that there's no additional value in further complexifying the model. In biology, the data tends to be quite noisy, so this plateau occurs rather swiftly.

Another technical reason, that is here to stay, is our reliance on model systems and organisms. Researchers study creatures like Drosophila or Zebrafish not *only* because they are intrinsically interested in their biology. These organisms are favored because they are inexpensive, amenable to genetic manipulation, subject to fewer regulations, breed quickly, and possess structures or genes that we can relate to our own through homology. While the precise quantitative behavior of these biological systems may diverge from ours, the fundamental principles they illuminate could remain valid. Therefore, these principles form the focus of studies involving these model organisms.

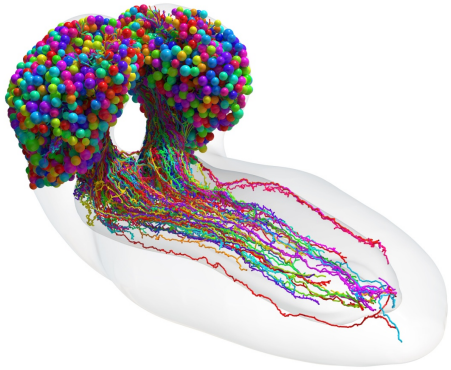


Figure 1.5: Dissecting Drosophila's brain will allow to understand how drosophilas think, but most importantly it will allow people to better understand how intelligence emerges from neuronal interactions. From [13].

For this reason, the focus has been more on biophysics, and less on quantitative biology in past years. This is poised to change, as every factors that were orienting towards biophysics are changing rapidely. First, biology is really hard, and it is not sure that every processes are understandable by humans. A good example of that is given by the marvelous complexity of gene regulation networks, where redundancy and complexity is the norm rather than the exception. It is absolutely possible that biologists will run out of understandable things to study, and that they will thus have to study less tractable problems, without general principles.

Secondly, the industrial players in the big tech industry are becoming increasingly interested in life sciences[6]. They will be driven by the therapeutic possibilities of biology, more than explainability and a desire to understand. A good example of that is given by disease prediction based on scans [14]. With them, a lot of people with a different skillset, from Applied-maths and Machine learning, are comming into the field. Statistical Physicists as me are uniquely positionned in this field, as we borrow techniques from traditionnal physics but have very good

[6] A lot of rich people fear death

understanding about the peculiarities of high-dimensional spaces. Additionally, the development of high-throughput methods such as microfluidics and robotics has enabled researchers to automate experiments and increase the volume of data. This in turn helps reduce the level of observation noise.

Eventually the new progresses in stem cell cultures allow researchers to have easier access to structures ressembling closely to mammalian embryos or organs, such as organoids, blastoids, and embryoids. These methods allow to perform high throughput experiments and gain insights about human biology without the traditional difficulties encountered when dealing with mammals. All these factors put together will probably lead to a golden age of quantitative biology in years to come, but a side effect could be that biophysics could benefit from this.

Then, another key change is the advent of robotics and AI. I firmly believe that we will see intelligent systems emerging very soon, and that these systems could greatly help us doing better biology. To better understand how these systems could work and how they could help us unravel the mysteries of life, I will first try to give a little introduction about how I believe human think, and how silicon-based systems reproduce such tasks.

1.3.4 Intuition, Logic and high-dimensional data

High dimensional embeddings

I will start this paragraph by saying that I am not a specialist in cognitive science, and only based this paragraph on what I read. I will thus be short and humble, but will nevertheless try draw some principles to help design computational systems capable of intelligence.

Humans possess two ways of thinking. We are creatures guided by intuition, and are able to do analogies between loosely defined and abstract concepts, but also able to perform rigorous logical reasoning when needed. Daniel Kahneman and Amos Tversky, cognitive psychologists, introduced the concept of "System 1" and "System 2", and vulgarized in the book "Thinking, Fast and Slow" [15].

System 1 refers to our intuitive system, which is typically fast, automatic, effortless, implicit, and often emotionally charged; it operates with little or no sense of voluntary control. This system excels at making connections and detecting patterns, and it often relies on mental shortcuts or heuristics to make quick judgments. It might correspond with the lower-level, automatic computational processes that handle tasks like initial perception and basic associative learning. System 2, on the other hand, refers to slower, more deliberate and effortful processing. This includes activities that require attention, such as complex computations, logical reasoning, and conscious decision-making.

Cognitive science often postulates a computational theory of mind, a paradigm widely agreed upon to some extent. According to this view, mental processes can be considered computations on mental representations. Associated with prominent mid-20th-century figures like Alan Turing [16] and Noam Chomsky [17] , this theory proposes that thoughts are computations performed on mental representations of our experiences, comparable to software processes in a computer. System 1 and system 2 would just be two different systems that engage with these computations differently. System 1, through neural computations, generates abstract representations of objects, potentially in high-dimensional space, with the highly constraining necessity of rapid processing [18–20]. System 2, on the other hand, allows us to

manipulate these concepts by conducting abstract computations with them.

A key element in this perspective is the concept of abstraction: our minds construct abstract models of the world, and we manipulate these models to understand and predict reality. The level and form of this abstraction can greatly vary, ranging from simple rule-based systems to complex high-dimensional representations, contingent on the cognitive task and context.

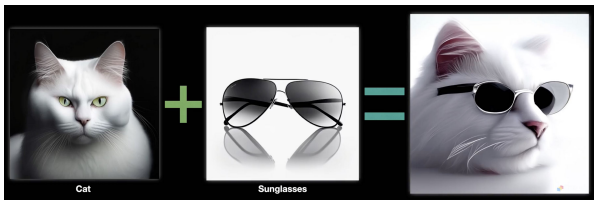


Figure 1.6: In our abstract representation of the world, the operation "Cat + Glasses" make sense. From [21].

Drawing parallels with artificial intelligence, some processes in machine learning models seem to mimic aspects of these two systems. For instance, neural networks conducting tasks like image or speech recognition could be viewed as operating in a "System 1" way, extracting patterns from high-dimensional data swiftly, automatically, and implicitly. In contrast, more symbolic or rule-based AI models, which explicitly manipulate abstract representations to execute logical reasoning, could be considered to operate in a "System 2" manner. Yet, these parallels aren't flawless, and AI models still lack several human cognition facets, including the seamless integration and mutual influence between System 1 and System 2, and the ability to perform high-level abstract reasoning across various domains, among other things.

Interestingly, feedforward deep neural network architectures, inspired by our visual system's structure, have enabled a reverse analogy: We can attempt to explain the mind by understanding how neural networks form. In certain architectures, like GANs, diffusion models, or AutoEncoders, the middle layer comprises a vector that holds a high-dimensional representation of the input data in a space termed the "latent space." In this abstract space, we can conduct abstract operations such as "Cat + Glasses = Cat wearing glasses" or "King - Man + Woman = Queen."

Figure 1.7: In the latent space of neural networks, abstract algebraïc operations leads to meaningful results. From [21].

Yet, when we wonder whether human cognition involves computations in high-dimensional spaces, like latent-space arithmetic in AI, the answer is not clear. Some aspects of human cognition, like language processing, visual perception, and certain forms of memory recall and imagination, are thought to involve computations in high-dimensional "semantic" or

"conceptual" spaces. However, human cognition also exhibits substantial flexibility, context-sensitivity, and integration across different modalities and cognitive processes, aspects that are challenging to emulate in AI.

Moreover, the brain's biological and physical properties, embodiment, and environmental interaction heavily influence the representations and computations in the human brain. This adds layers of complexity beyond the high-dimensional latent spaces typically employed in AI. Thus, while there are parallels and shared insights between human cognition and AI models like latent-space arithmetic, it's crucial to acknowledge the unique attributes of human cognition and the ongoing challenges in thoroughly understanding and replicating it. There is much to learn about the intricate workings of the human mind, and while I'm not a specialist, I believe that the concept of computations in a high-dimensional space is a key feature of intelligence that could be used to comprehend complex systems, including biological ones.

In this context, **Vector Databases** serve as essential tools for handling high-dimensional data, particularly in AI and machine learning applications where complex operations on this data are performed. They store, index, and enable efficient querying of large amounts of high-dimensional vectors, often derived from raw data through encoding or embedding processes made with deep neural networks. They operate by implementing algorithms that enable efficient search operations within this high-dimensional space, to enable quick similarity searches or other complex operations. In the same way that latent space arithmetics allow abstract operations to make sense, vector databases allow to cluster together similar piece of data, allowing to deal with complex amounts of data: In a database composed of images, the nearest neighbor images of an image of a cat on a sofa will most probably be other images of cats on a sofa. We will see later very useful practical application of this abstract idea on fluorescent images of cells.

Multimodality and Latent space arithmetics

Multimodality has been an impressive and surprinsingly simple-to-implement breakthrough achieved by AI. One can incorporate Images, Text, and even Sound, and obtain high-dimensional latent state representations where computational operations make sense, allowing for the emergence of models capable of dealing with multiple modalities, such as text-to-images, images-to-text, text-to-speech or speech-to-images models [22–25].

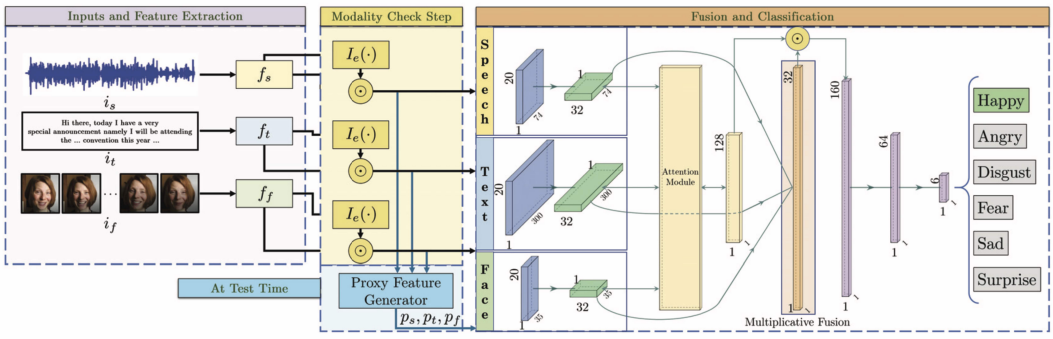


Figure 1.8: Speech + text + images to text architecture: Neural network architectures can deal with multimodality easily: A different input network for each modality allow to blend informations from sound, text and images. From [26].

Biology operates on a multiscale and multimodal basis, necessitating a deep in-

tegration and understanding of various forms of data. AI models could make sense of mechanical, visual, morphological, genetic, transcriptomic, temporal, and contextual data, to discover patterns and relationships on biological systems, solving mysteries that would have remained forever out of reach for humans with traditional methods. Despite the difficulty to interpret their results, AI capabilities constitutes a promising avenue The predictive and generative functions of such models could form the cornerstone of novel therapies. By facilitating an unprecedented understanding of biological phenomena, they could open the door to innovative treatments that target the root causes of diseases rather than just the symptoms, driving forward the frontier of personalized medicine.[7]

[7] By the time I wrote my thesis Google just released Med-Palm M [27] a multimodal version of their large language model for healthcare Med-PaLM [28]. The writing was on the wall.

1.4 Taming complexity in biology

1.4.1 The future of data-gathering and data-analysis

As in many other fields, biology is witnessing and will continue to witness huge developments because of AI. The newest large models allows us to design algorithms that would never be possible to realize otherwise. Biology research, as many other intellectual productions that we believe are reserved to humans because they requires intelligence, is a data-processing tasks: it fundamentally consists in data gathering and data analysis, i.e creation of models from experiments.

People keep saying that we are overwhelmed by a deluge of data. I would qualify this statement by saying that we are overwhelmed by a deluge of inexploitable data, and the deluge of data is overwhelming only because it is inexploitable. We need more data that is easy to analyze, and it appears that it is quite hard to do so, for reasons that are specific to each particular problem. But here is a general principle: It is useless to produce data that cannot be analyzed, and it is useless to design analysis pipelines for data that cannot be produced. As a consequence, data gathering and analysis are two operations that are becoming more and more intertwined in recent years: Algorithms are able to do very high throughput data-analysis if and only if the data respect some rules. People that are being able to control both acquisition and treatment of data in a vertically integrated manner[8], or a very tight and efficient collaboration, really have an edge over projects where the data is analyzed by people who had nothing to do with their production.

Since a few years, inventing new data-analysis methods have been very valuable, and the bulk of the work that I present in this thesis consists of finding clever methods to analyze 2D and 3D microscopy images. A new very recent information is that AI will solve many aspects of data analysis very soon, if the data comes with a stereotypical appeareance. Instead of replacing whole pipelines, AI modules will most probably consistute new parts of it, functionning on-par with more classical algorithmic methods, allowing new pipelines one could not dream of before. As an example, in my work, I repeteadly use cellpose [29] as a black-box instance segmentation AI module, that constitutes only the first step of my pipelines.

For now, the biggest successes in quantitative biology have been done in what we call Omics, that is data obtained from sequencing, because these are very structured information: DNA, RNA and proteins can be described as chains of letters, sequenced and analyzed. In this field, we know what we are looking for, we know how to represent it, we know where to seek it. We just need to do that as fast and precisely as possible with a minimal cost. In imaging however, the quantities that we want to measure are not as obvious. We need more efforts to formulate what we are looking for in a clear mathematical way, but eventually cannot avoid this step. Because of that, in omics as imaging, the future will lie in high throughput data production, where biologists will produce massive amounts of data in a standardized manner, that will then be analyzed almost instantly and shed light on biological questions in a quantitative manner. Generating this data will require to have a clear idea of how the analysis pipeline work, and of the kind of insights we could extract from each piece of data before acquiring it.

This standardisation of data is far from easy, and there is going to be two regimes: First, each question will go through the discovery regime, to test the data-analysis pipeline with a

[8] I immediatly think of the labs of Prisca Liberali, Andrew Oates and Loïc Royer when writing this, but there are many others

biological system in an iterative manner, with small batches[9]. This phase will allow to fine tune the procedure, the biological question and the data-analysis algorithms. Then the second phase arrives, where the procedure has to be "industrialized". There is often many ways to proove something in biology, but some can be high-throughput whereas others can be very hard to analyze. (Of course, a common approach is to use several of them to see if they tell the same story).

Microfluidics is a very popular way to do experiment in a repeatable fashion, allowing to obtain large statistics and to do very accurate tasks at the micrometer scale, and is at the core of the single-cell RNAseq techniques. Both optogenetics and mechanical perturbation with optical tweezers could also be used to perform high throughput experiments as well if programmed correctly. These technologies can be seen as a merging between robotics and microscopy: Motorized stages gave microscopes legs, allowing the microscope to move freely accross a very large field of view. The optical tools gives the microscope arms, allowing it to perturb and manipulate samples at will. Computer vision and more generally AI constitutes the final element of this stack: they are giving microscopes a brain, allowing it to analyze data, plan and perform complex tasks on live.

This high throughput approach is really needed to make better biology, to have a better vision of the whole span of phenotypes one could observe, and to allow new phenomenas to be understood where, as I said before, the interesting signal is currently impossible to obtain because of the noise of the biological experiments.

1.4.2 The problem AI solved in quantitative biology: Image segmentation

Image segmentation is a fundamental process in computer vision that enables the division of an image into multiple segments, or 'regions', with each region corresponding to different objects in the image. It forms the backbone of many modern systems, spanning from autonomous driving to medical imaging. However, when applied to biology, the typical image segmentation techniques often fall short. Biological images present unique challenges, such as varied cell shapes and sizes, intricate structures, and diverse imaging conditions, and most importantly volumetric (3D) data, requiring new segmentation methods.

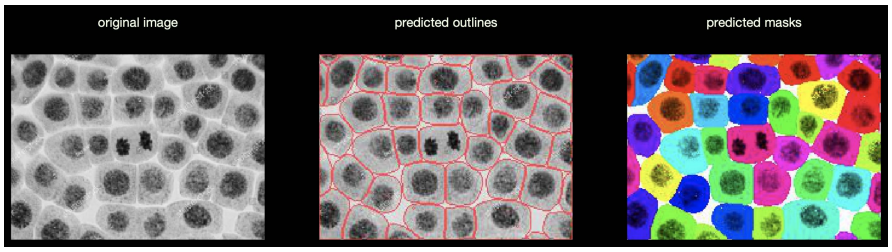


Figure 1.9: Automatic instance segmentation with deep-learning [31].

As in other image analysis tasks, deep-learning proved to be a game-changer. Most architectures today are modifications of a U-net [32], that revolutionized the field. The model's

[9]The "Law of Small Batches" is a concept derived from lean management principles, particularly as applied to manufacturing and software development. The idea is to work on smaller batches of work, rather than large ones, to improve flow, reduce waste, and increase productivity. This is predicated on the principle that large batches can lead to bottlenecks in processes and longer lead times, as well as increased risk of waste due to defects or changes in demand. In contrast, smaller batches allow for quicker feedback and more rapid adjustments, leading to improved efficiency and quality. [30]

architecture consists of an encoder that captures the context of the image and a symmetric decoder for precise localization. With clever data augmentation techniques to leverage the available annotated samples, the network gives surprinsingly good results on segmentation tasks, even when trained on a limited number of images.

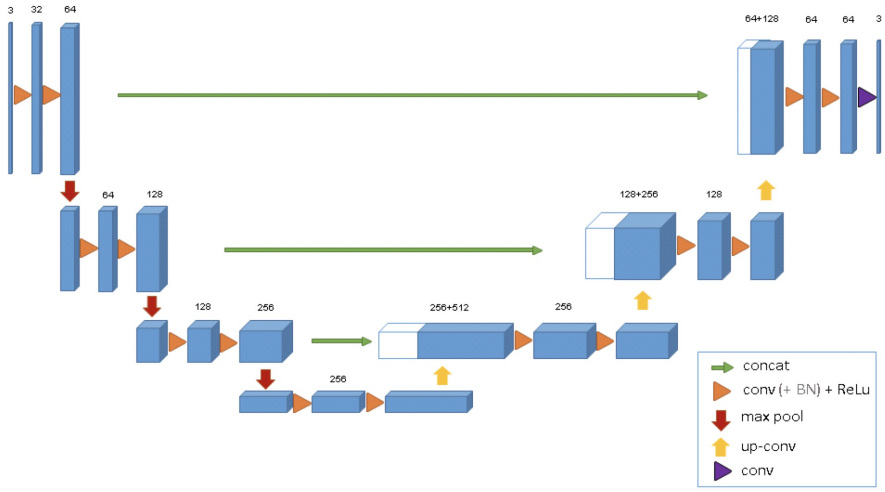


Figure 1.10: The U-net autoencoder structure has become the standard in bioimage analysis. From [32].

Other AI models soon followed. Stardist, developed to detect cell nuclei, introduces an inductive biais by inferring the geometry of star-convex polygons to localize instances in 2D images, overcoming issues related to crowding and false merging of cells [33]. Generalists models, trained on a wide array of data, such as Cellpose [31] also became available, avoiding the need of model retraining or parameter adjustments. Similarly, generalizations such as Omnipose [34] were developped for specific use cases, such as segmenting bacterial of elongated or branched morphology.

To tackle volumetric data, Unet has been generalized to 3DUnet [35, 36], maintaining its primary architecture while replacing all 2D operations with their 3D counterparts for volumetric segmentation. Similarly, Stardist was extended to 3D volumes [37], replacing polygons with star-convex polyhedra to represent cell nuclei, and Cellpose [31] can use its 2D backbone in the XY/YZ/XZ axis to be generalized to 3D instance segmentation. Image segmentation was a very hard problem that has been solved by deep-learning, allowing researchers to devise new methologies leveraging such automatic data-analysis.

1.4.3 Kitchen recipe and algorithms: the marginal cost of intelligence

I will end up with a very important principle when dealing with complex systems, that is the marginal cost of intelligence.

My teacher in computer science used to tell us that a good example of an algorithm is a kitchen recipe. Lets evaluate that at the light of the marginal cost, which is the cost of producing one more unit of service. If you want to reproduce a kitchen recipe, even though the recipe is written and all the ingredients are already there, it will require you some time and effort to reproduce the recipe. The marginal cost is not zero. However, if you want to apply an algorithm on some piece of data that you have, you will need almost no time to do that:

The marginal cost of intelligence is close to zero with algorithms.

This notion is very important: reducing marginal costs is what allow you to make progresses, by letting people put time and effort on new things instead of repeating all the time the same tasks. Over time, humanity has progressed a lot in this domain, reducing the marginal costs of one tasks after another, allowing itself to operate at a higher and higher level, freeing itself from the slavery of the nature.

If we stay in the fields of thought, we can notice a lot of differences between mathematics and philosophy. We have to say that when we compare ancien Grece and today universities, the quality leap of the philosophy produced does not seem huge[10], contrary to the evolution witnessed in mathematics. Why so ? Marginal costs. Once a theorem is proved, it is proved forever, and we can directly use it and build over it, focusing your efforts on other things. In philosophy, the ambiguity of the terms means that everybody always end up rediscussing everything, resulting in a stagnation across time. (Or maybe I am wrong and this is the point of philosophy, discussing endelessly without following any clear goals...).

At the light of this, we can assess the depth of the change brought by AI: today, the marginal cost of intelligence is decreasing incredibly fast, and may be close to zero sooner than we think. It is in this vein that I want to insert my advances: briging the marginal cost of my work down to zero: I have solved problems once and for all, and I want everybody to use my tools without having to solve them again, and rebuild what I built. You specify an input, and it returns you an output, in near-zero time (this near-zero can be up to several days as we are going to see).

Throughout my thesis, I made it a priority to provide freely available Python implementations for all the algorithms I proposed. Now, if someone wants to mesh a 2D contour or a 3D surface, and infer forces on it, they can do so without having to navigate all the complexities of the process. If someone wants to measure flow, they don't need to become proficient in optimal transport; they simply need to ensure that their system meets the required assumptions. Finally, if someone wants to create artificial images or engage in differentiable rendering of microscopy images, they can use my highly optimized GPU implementation directly. All of these new operations can be readily applied in various use cases.

1.4.4 What about top-down approaches ?

A thinker that is sincere enough will himself recognize a good demonstration when he sees one, no matter its form. There is no silver bullet. I praised a lot bottom-up approaches because this was a really logical and conservative way to proceed. What about top-down approaches in this description ? Are they as valuable as bottom-up approaches ? Where do their value come from ?

In statistical physics, renormalization group theory gives some answers to this question. A lot of microscopic systems can be coarse-grained into the same macroscopic description, called a universality class, and will be undistinguishable from one another if we do not probe the system with a sufficiently high resolution. Thus we formally cannot deduce the microscopic nature of the system by looking at it at a larger scale. Top-down is not the inverse of bottom-up, bottom-up approaches are more powerful.

[10] I even prefer the old one.

However, in practise, the physicist will often go back and forth between these two approaches, by what I call retrospective top-down: Looking at the top system (i.e at the universality class), one can constrain the microscopic system, as it has to be compatible with the macroscopic observations. This microscopic system can then be used to do predictions of the quantities at a larger scale.

A paper that I will briefly cite later explain this approach very well [38]: "Finally, our top-down approach could help advance our understanding of locomotion at the molecular level, as it provides strong constraints for bottom-up models that connect microscopic rules to the system-level dynamics of cells".

The principal pitfall of biophysics is what I would call *"the illustrative model"*. The model take a lot of parameters and kind of reproduces what you observed experimentally. And that's it, you do not get any value from it, because in fact you are not very sure this is how the thing works. I believe that one of the reasons such illustrative models exists is because we are studying systems that are too complicated to model. Top-down approaches can be tempting, but it is sometimes hard to extract any valuable insights about the microscopic nature of the system. I will thus prefer to them an fundamentally reductionnist approach that can answer to a lot of questions: To study simple systems, and complexify the system little by little. In the following section, in will explain how this applies in my "wide" domain, which is the mechanics of morphogenesis.

1.4.5 Doublet Biology

Cell biology has been a cornerstone of biological research for many decades. It began in fact with the discovery of the existence of cells: early scientists used microscopes to observe the microscopic structures within living organisms. As our understanding of cells and their components has grown, so too has our ability to dissect the complex processes that govern life. This reductionist, bottom-up approach has enabled us to gain valuable insights into the fundamental workings of biological systems, laying the foundation for advances in medicine, genetics, and countless other fields.

However, as the field advanced, researchers recognized that studying single-cell was not the end of the story: Cells interact with one another, and forms complex networks and multicellular organisms. To gain a deeper understanding of living structures, it is important to study them as a system, and not only component by component. Today, systems biology is one of the most fancy topics in biology. I am astonished by the diversity and the complexity of the systems studied by the researchers, and the unique data that people have access to. The problem, however, is that life is a mess. If you want to study an intricate system because you find it sexy to embrace complexity, you will find one without any difficulty. I firmly believe that too many people study systems that are way too complex. This leads to data that is very difficult to analyze, and it is hard to see what insights we can get from them. I would have loved to collaborate with biologists who did things with $\approx$ 10s of cells, but they were surprinsingly hard to find[11].

This disagrement between the data produced and the analysis and modeling tools made me understand that **we needed to study simpler systems, and focus on the simplest system one can think of beyond a single cell: two cells**. By examining how two cells

[11] Save for our friends Rémi Dumollard and Alex McDougall from the marine station of Villefranche-sur-Mer

interact, we can dissect fundamental processes such as adhesion, mechanochemical feedback loops, and the principles governing multicellular communication. This approach has the potential to shed light on many biological questions that would be difficult to address in more complex systems. I thus want to propose the formal establishment of a new approach called "Doublet Biology". In this approach, we would push the reductionism as far as possible, focusing on simplified, controlled systems, that we can easily dissect and make reproducible. This simplified setting could facilitate the development of predictive models and quantitative understanding of cellular interactions, which could then be extrapolated to more complex systems.

In the following sections, I will delve deeper into existing studies that lay the groundwork for this more integrative form of cell biology. This emergent subfield is currently underexplored and holds immense promise. However, it is essential to recognize that just as biology cannot be entirely encapsulated within cell biology, binary interactions will not fully encompass biological complexities. As such, the evolution of cell biology will undoubtedly involve the incorporation of more integrated approaches, such as the inclusion of extracellular matrix, and more complex multicellular systems. Doublet biology[12] is just the next step, not the final destination.

1.4.6 Sparks of Doublet Biology

I will cite some works in this line of thought. There are some remarkable studies have employed similar methodological approaches that I will not have time to delve into [39–41] Retrospectively, the biggest regreat of my PhD is to have studied systems that are a little bit too complex. Reflecting on my experience, I will take some time to highlight the value of studying biological systems.

The examples that I will cite use doublets to investigate adhesion, cell displacement and interaction, and migration. Let me make explicit the methodology that I like:

- They start studying a doublet, which is the minimal system expressing the emergent behaviour that interests them.
- They do quantitative models, that allow them to have a quantitative understanding of the phenomena at play, as well as more abstract "high level" insights about the behaviour of the system.
- They observe a complex system, often in-vivo, to see how the principles studied in the doublet can be generalized in a more complex system.

Mechanical dissection of cortical adhesion

The first study [42] addresses the mechanisms involved in cell sorting during gastrulation in zebrafish, specifically focusing on the roles of cell adhesion and cortex tension. The authors started by providing a mechanical model of two progenitor cells in contact, where the size of cell-cell contact is determined by the balance of forces at the contact boundary:

$$\cos\theta = \frac{2\gamma_{cc} - \omega}{2\gamma_{cm}} \tag{1.2}$$

[12] I know, Doublet biology does not sound very sexy. Should call it Cell Biology 2.0 to make it attractive ? I fear so...

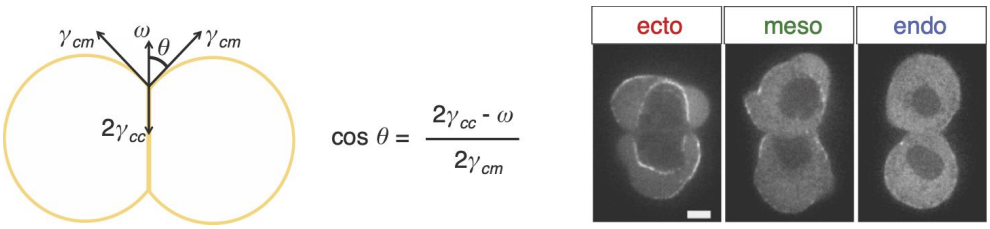


Figure 1.11: Doublets follow a simple model, where tension and adhesion intervenes in the determination of cell shape. Three cell types, ectoderm, mesoderm and endoderm, are present in the embryo allow to study different conditions. From [42].

where θ is the contact angle of the two adhering cells, γ_{cm} the tension at the cell-medium interface, γ_{cc} the cortex tension at the contact, and ω the negative contribution due to adhesion.

Traditionally, it is believed that adhesion increases the contact size, while cortex tension reduces it. The research team then used an innovative approach to examine the mechanical parameters controlling progenitor cell-cell contact formation. By measuring the contact angle of freely adhering cell doublets and examining the ratio of the interfacial tensions at the cell-cell and cell-medium interfaces, they uncovered that ectoderm doublets displayed a larger contact angle and a lower ratio of cell-cell to cell-medium interfacial tensions compared to mesoderm and endoderm doublets.

Upon measuring the cortex tension at the cell-cell and cell-medium interfaces, the study confirmed that cortex tension at these interfaces controls the cell-cell contact angle and, thus, the contact size. Interestingly, they found that the magnitude of adhesion tension ω was considerably smaller than the cortex tension at the cell-cell interface, contrary to previous suggestions. Although adhesion tension seems to have a limited role in cell-cell contact expansion, the formation of adhesive bonds remains essential. These bonds allow mechanical coupling of the contractile cortices of the adhering cells at the contact, supporting stresses normal to the adhesion zone. E-cadherin adhesion thus acts as a mechanical scaffold enabling cortex tension to control contact expansion.

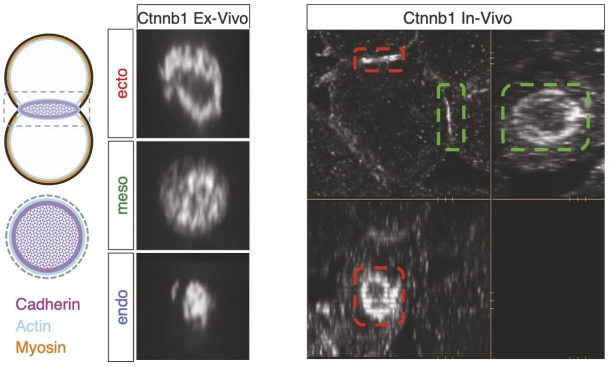


Figure 1.12: Cadherins are assential to provide cell adhesion. From [42].

To better understand the function of mechanical coupling due to adhesion, the researchers measured, on ecto/meso/endoderm the separation force and the fluorescence intensity at the contact. They found that ectoderm doublets exhibited higher separation force at varying contact times compared to mesoderm and endoderm doublets, suggesting that cytoskeletal

anchoring of cadherins is a critical factor determining the difference in separation force and cell-cell contact size between the progenitor cell types. Eventually, they made the interesting finding that the cytoskeletal anchoring of cadherin led to the formation of membrane tethers at the separation between two cells. These tethers can be interpreted as a signature of cadherin-mediated cytoskeletal anchoring.

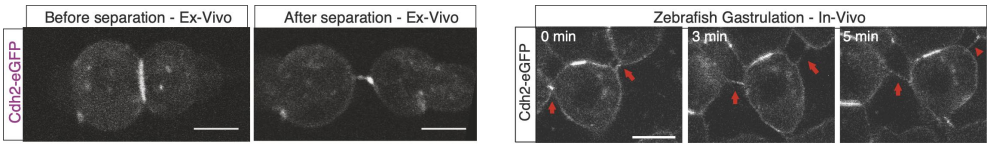


Figure 1.13: Membrane tethers forms at separation, both ex-vivo and in-vivo. From [42].

After this ex-vivo study on doublets, the authors look at the actual in vivo situation during zebrafish gastrulation, observe the same behaviours, and thus postulate that progenitor cell-sorting is driven by the contact-formation mechanisms that they observed on the doublet, and can make interesting postulates about the causality relationships between cortex tension and anchoring strength of the mechanosensitive adhesion complexes.

Dynamics of confined cell interactions, one cell at a time.

I will briefly discuss two studies that describes a series of experiments and theoretical models that together elucidate the dynamics of cell migration in a confined space, with a focus on disentangling the stochastic and deterministic elements of cell behavior. The first study is focused on what happens on a single cell [38]. Then the authors wanted to integrate this knowledge and started studying interactions between cells, and made the great decision to study it with only two cells [43].

The first paper [38], focuses on understanding the dynamics of individual cell migration when they are confined in a structured environment. For this purpose, the researchers designed a two-state micropattern system, consisting of two adhesive sites connected by a thin constriction. They confined individual cells in this micropattern, and monitored their movements.

The researchers collected a large set of data containing trajectories of individual cells (both

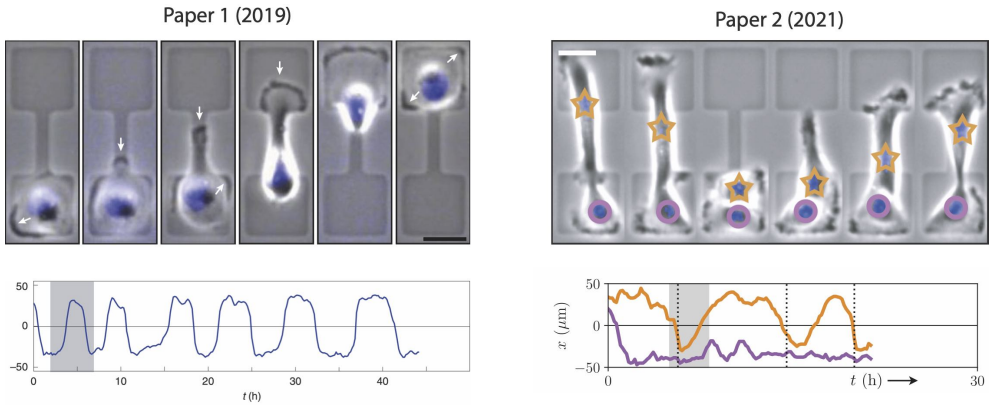


Figure 1.14: Integrating the knowledge at a system level can be done with only two cells to start with. From [38, 43].

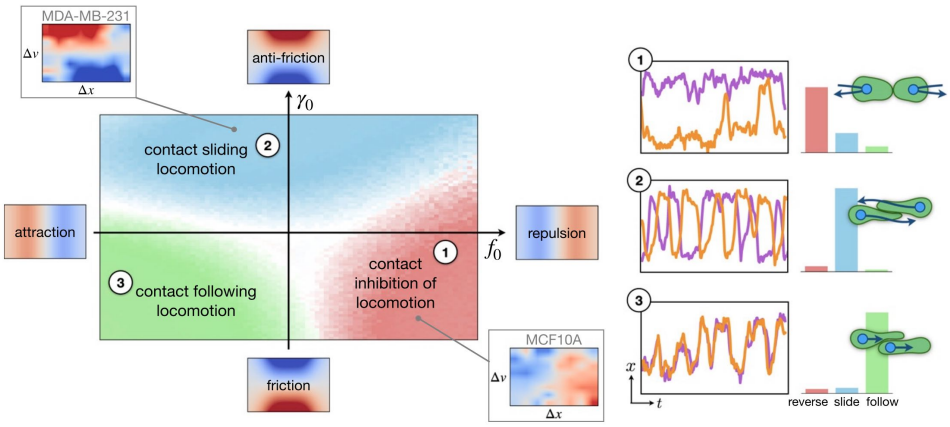


Figure 1.15: The two cell lines studied lie on a larger phase spaces, where maybe other cell types could be mapped. From [43].

cancerous MDA-MB-231 and non-cancerous MCF10A cells). These trajectories highlighted that the cells perform repeated stochastic transitions between the two adhesive sites, driven into the thin constriction. The authors fitted the position data to deduce a simple differential equation, composed of a deterministic term $F(x, v)$ and of a stochastic term $\sigma(x, v)$:

$$\frac{\mathrm{d}v}{\mathrm{d}t} = F(x, v) + \sigma(x, v)\eta(t), \text{ where } \eta \text{ represents a white gaussian noise.} \tag{1.3}$$

This equation demonstrated that confined cell migration can be accurately captured by a relatively simple second-order Langevin equation, including a two-dimensional nonlinear deterministic term and a state-dependent noise term. Interestingly, the two cell lines exhibited distinct deterministic dynamics: the MDA-MB-231 (cancerous) cells displayed a limit cycle (a repetitive path in phase space), whereas the MCF10A cells displayed bistable dynamics (two stable states).

Building upon their earlier work, the authors extended their investigation to include the dynamics of cell-to-cell interactions [43]. They modified their experimental setup into a "cell collider," which allowed two cells to collide and interact in the confined micropattern space. They adapted the stochastic differential equation to account for cohesive interaction and friction:

$$\frac{\mathrm{d}v}{\mathrm{d}t} = F(x, v) + f(|\Delta x|)\Delta x + \gamma(\Delta x)\Delta v + \sigma(x, v)\eta(t) \tag{1.4}$$

where $f(|\Delta x|)\Delta x$ is the cohesive interaction term and $\gamma(|\Delta x|)\Delta v$ is the friction term.

By observing these interactions, they discovered that distinct cell types exhibited diverse interaction behaviors: noncancerous MCF10A cells showed behaviors characterized by repulsion and friction, while the cancerous MDA-MB-231 cells displayed attraction and antifriction behaviors. These divergent dynamics suggested that different types of cells follow unique interaction rules during their migratory processes, potentially influenced by their intrinsic biological state (cancerous vs noncancerous). In fact, these two behaviours are just particular cases of a larger phase space, that may be explored by other systems

In both papers, the researchers used a data-driven approach to develop a theoretical framework that describes the dynamics of cell migration in confined environments, either in the context of a single cell or pairs of interacting cells. In the first study, the research focused

on understanding the dynamics of individual cells, demonstrating the interplay of deterministic and stochastic dynamics in driving the cell into the thin constriction. In the second study, they built upon this foundation to investigate how these dynamics manifest when two cells interact with each other. This simple system allow to really decipher and understand the processes at play, contrary to a system containing more cells that would be harder to interpret.

It takes two to make a collective

Another good example can be provided by a study about collective dynamics in migrating cells [44]. From the author's words, *"The Ciona cardiopharyngeal progenitors (heart and throat precursors) provide the simplest model of collective cell migration".*

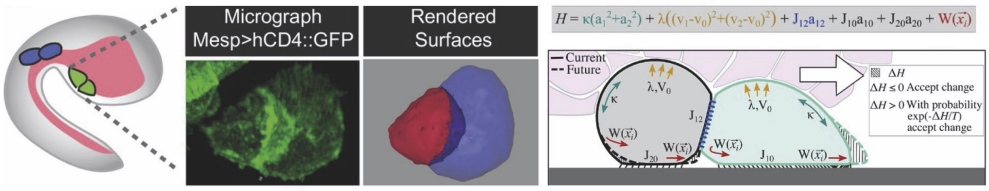


Figure 1.16: Left: localization and shape of the cardiopharyngeal progenitors. Right: The cellular Potts model used to describe the shape of the cell. From [44].

They study the mechanisms of collective cell migration using this simple biological system. The findings demonstrate that cardiopharyngeal progenitors move in pairs composed of a leader and of a trailer. Doing so, they maintain supracellular polarity with differing distributions of cell-matrix adhesion, myosin-based retraction forces, and protrusive forces along the leader-trailer axis (direction of migration). They act in a complementary manner, instead of reproducing the same pattern, showing a minimal system where an emergent collective behaviour can be observed.

The Cellular Potts Model (CPM) is used to examine the force distributions that align with the shapes and collective polarity of migrating cell pairs. This computational model is used to simulate cellular behaviors and interactions at the collective level, such as cell sorting, cell migration, and tissue growth. Rather than representing each cell as a single point or sphere, the CPM represents cells as collections of smaller units or pixels, each with an assigned "cell ID". Cells can change shape and move by changing the cell ID of their pixels and those of their neighbors based on certain energy minimization principles. In this model, the behavior of cells is governed by an effective energy function, or "Hamiltonian", which takes into account cell-cell adhesion, cell volume, cell shape, and other biophysical properties. For a pair of cells of area $\mathrm{a}_1, \mathrm{a}_2$, of volume $\mathrm{v}_1, \mathrm{v}_2$, of target volume v_0 and of area with the substrate a_{10}, a_{20}, and of cell-cell area a_{12}, the Hamiltonian can be expressed as:

$$H = \kappa(\mathrm{a}_1^2 + \mathrm{a}_2^2) + \lambda((\mathrm{v}_1 - \mathrm{v}_0)^2 + (\mathrm{v}_2 - \mathrm{v}_0)^2) + \mathrm{J}_{10}\mathrm{a}_{10} + \mathrm{J}_{20}\mathrm{a}_{20} + \mathrm{J}_{12}\mathrm{a}_{12} + \mathrm{W}(\vec{x}_i) \qquad (1.5)$$

where κ models the surface tension and λ the hydrostatic pressure. J represent the mechanical adhesion, and W describes the protrusive and retractive forces. This Hamiltonian is a generalization of a simpler Hamiltonian for the single-cell case, and it is easy to see how it could be generalized to a larger number of cells.

To have an even simpler system, the authors did an experimental perturbation on one cell, by inducing a dominant-negative inhibitor of $Sar1^{dn}$, that stalls the migration of the transfected cell, leading to only one migrating cell. They use this single cell migration experimental data to validate the Potts model and to build a phenomelogical model of migration of the Trunk Ventral Cell (TVC) on the epidermis, before extending the model to the pair observed in the WT phenotype, with a leader and a trailer cell.

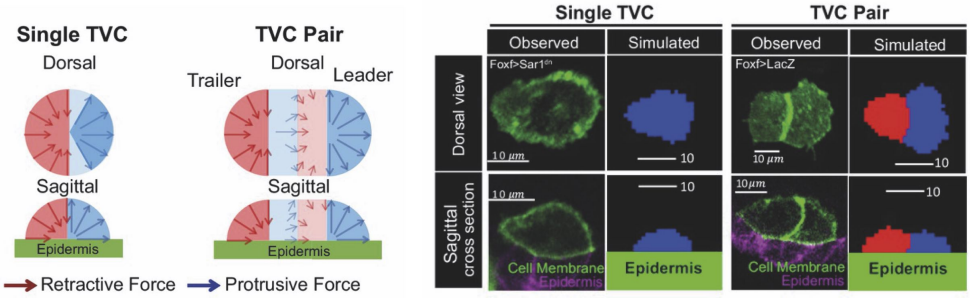


Figure 1.17: Left: Phenomelogical model of cell migration. Right: Experimental validation of the cellular Potts model. From [44].

In order to assess what determines which cell becomes the leader and the trailer, they do pharmacological perturbations. They deduce interesting and significant statistical patterns that are out of the scope of this section, but informs about the cellular pathways involved in the leader-trailer mode of migration. Following the line of thought of integration via progressive complexification, the authors extend their system to the 3-cell case, showing that they naturally adopt the leader-trailer mode of migration. They also manages to perturb their system to obtain four TVC instead of two, and observe that they migrate following the leader-trailer mode of migration.

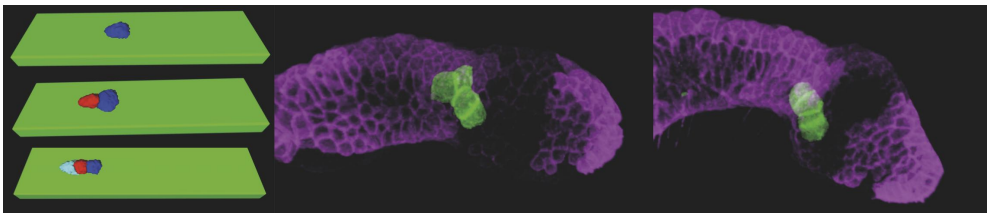


Figure 1.18: Extension with 3 cells, perturbation in vivo to see with 4 cells. From [44].

Eventually, after having looked at the interaction of the doublet with the endorderm, the authors propose that the persistent and directional migration of cell pairs might be due to the emergent physical properties resulting from the polarized supracellular organization of these progenitors. They suggest that these properties could be crucial for the cells to overcome the mechanical resistance of their surrounding environment during their migration. They also highlight the need for balance between cell-cell and cell-matrix adhesion in maintaining the collective polarity and integrity of the migrating cell pairs. Furthermore, they suggest that cell-cell communication is critical for aligning collective polarity with the direction of migration, as observed in simulations and in vivo observations.

The study concludes that supracellular organization may influence the directionality and mechanical interactions of migrating cells with their environment, facilitating more efficient

migration over longer distances. Methodologically, their way to proceed is very logical and progressive: they study what happens in a single cell, see to what extent this generalizes to a doublet, and what emergent properties could appear in this more complex system.

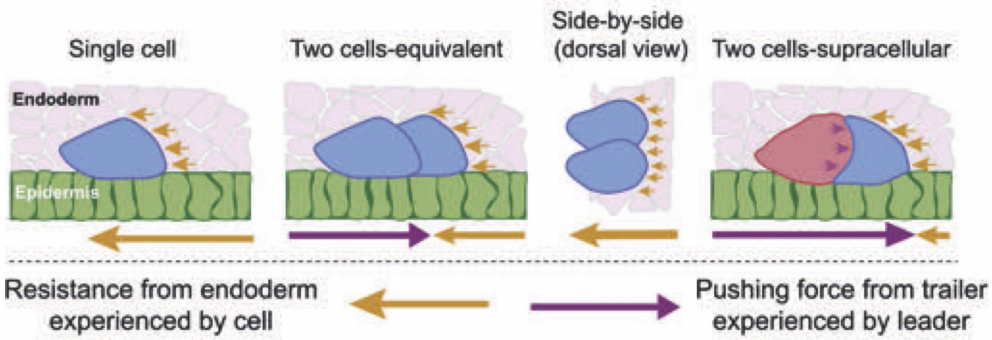


Figure 1.19: Emergent behaviour of the two cells. From [44].

1.4.7 Beyond reductionism - Biology by the numbers

Aside from reductionism, the thermodynamics-like approach of measuring macroscopic properties and study the system through their lens can offer many insights. Determining which macroscopic quantities to measure is hard however, and I think that the most interesting quantities may be backed with some physical theories, mostly from solid and fluid mechanics (viscosity, elasticity, stresses) or statistical mechanics (cell-cell contacts, cell shapes, topological charges [45, 46]...). Measurings such quantities allow to proceed to a whole kind of mathematical operations between them, allowing to formulate predictions, among which:

- **Order relation: If $a < b$ and $b < c$, then $a < c$.** In case of 1D data, measuring mathematical quantities allow to track the temporal evolution of a quantity, and make a lot of observations, assertions and predictions about them.

- **Distances:** In case of data of higher dimensions, one can study the topology of the space, looking about the relative distance between each other, which allow to quantify similarity and dissimilarity. It can also be relevant to study the different modes of variation of the data.

- **Symmetries:** One can also look at invariances in the patterns, and at conserved quantities, as they can be very informative about the nature of the system. (speak about noether theorem).

However, such measurements can be far more abstract that just physical quantities. As an example, it is possible to extract value from transcriptomics or imaging data (by using self-supervised learning to create informative embeddings) using the same methodology.

1.4.8 Early successes of quantitative biology in high dimensions

Single-cell RNAseq

Single-cell RNA sequencing (RNAseq) is a revolutionary technique that was developed as a response to the limitations of bulk RNA sequencing, which could only provide an average

gene expression profile of a multitude of cells, thus obscuring the nuances and heterogeneities within individual cells. Single-cell RNAseq emerged as a solution, enabling researchers to decode the transcriptome—the complete set of RNA transcripts produced by the genome—of individual cells, thereby providing insights into the unique functional characteristics of each cell and shedding light on complex biological processes and diseases. **A key enabler of this transformative technology is the application of microfluidics, manipulation and control of fluids at the microscopic level**. Circuits are designed to isolate individual cells and execute single-molecule barcoding, a technique crucial for identifying and distinguishing the transcripts of each individual cell.

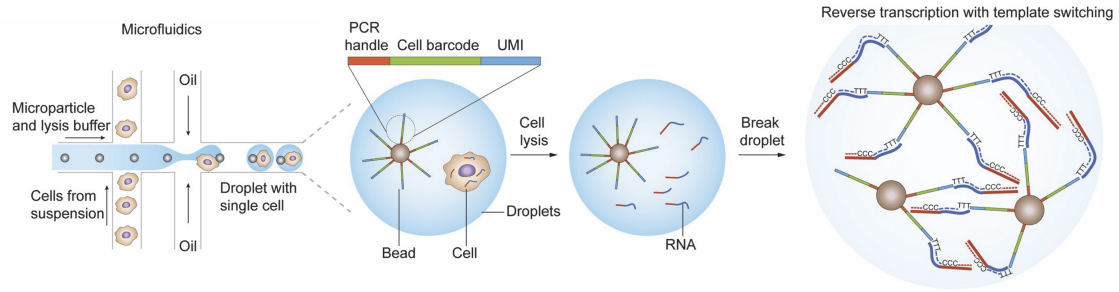


Figure 1.20: Microfluidics barcoding allow to trace the host cell of each RNA fragment after cell lysis and transcription, allowing RNA-sequencing at the single-cell resolution. From [47].

To visualize and analyze the high-dimensional data produced by single-cell RNAseq in a more comprehensible manner, both **dimensionality reduction and clustering algorithms are vital**. Dimensionality reduction algorithms such as t-SNE [48] or UMAP [49] have become particularly valuable in this field, as well as clustering algorithms such as k-nearest neighbors [50, 51]. t-SNE and UMAP work by reducing high-dimensional data into two or three dimensions, while preserving the distances between data points.

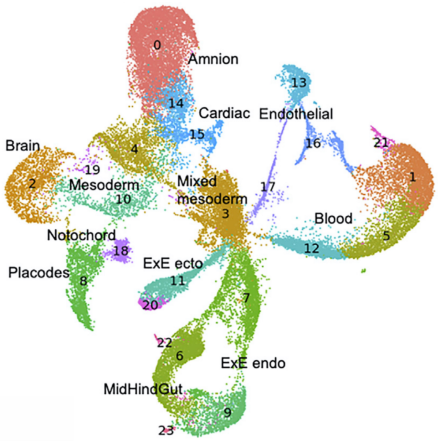


Figure 1.21: UMAP of an artificial "embryoid" created from stem-cell cultures. Different cell types appears in clusters of different colors. From [52].

Single-cell RNAseq embodies a top-down approach, starting with a complex, high-dimensional model that is then distilled into simpler, more interpretable conclusions. This surprising ease of interpretation, one of the great advantages of single-cell RNAseq, facilitates a seamless oscillation between microscopic details and more coarse-grained, system-level, mesoscopic

properties. It is currently at the basis of new studies shedding a new light on old problems, allowing an unprecedented understanding of cellular behaviors and dynamics at both the individual cell level and the broader system level.

Self-supervised methods in biology

As said before, methods leveraging artificial intelligence and machine learning have been increasingly employed to streamline and automate various analysis processes. However, especially in biology, the generation of annotated training data is challenging or resource-intensive.

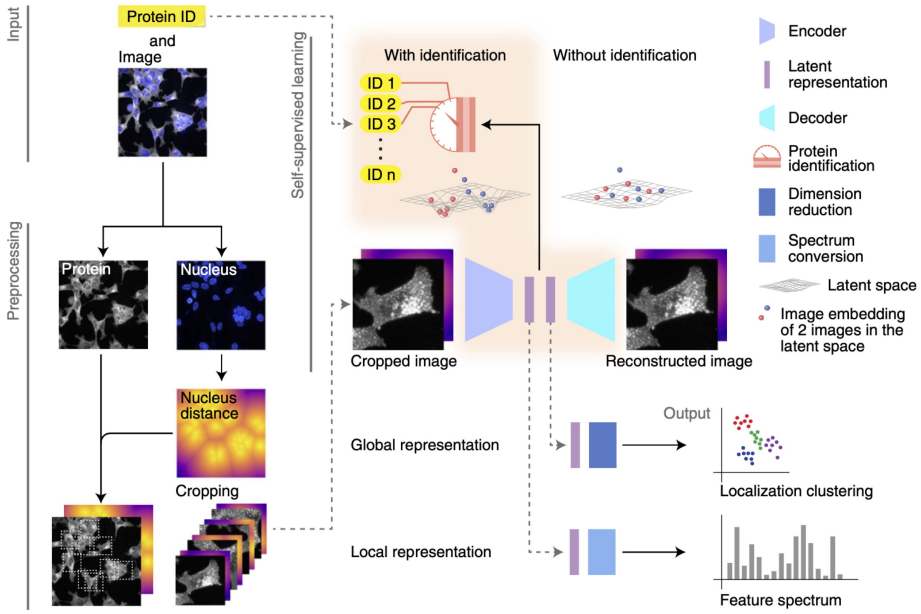


Figure 1.22: The self-supervised approach works by learning a pretext task, in this case to predict protein distribution and identity. From [53].

When looking at other fields of information processing, a prominent methodology is self-supervised learning, which allow neural networks to learn without the need of annotated data [54–58]. **Self-supervised learning works by pretraining the neural network on a pretext task, that can be formulated based solely on unlabeled data** (such as completing holes in a text), allowing the filters of the neural network to generate embeddings, i.e high dimensional representation of our input data, that capture high-level image semantics. These representations can then be fine-tuned or used directly for a downstream task with available ground truth. This method can be particularly impactful in fields like live-cell microscopy where data annotation can often be laborious, and it becomes crucial to discern meaningful representations of microscopic images.

Cytoself uses self-supervised learning for protein localization profiling and clustering, enhancing the understanding of cellular architecture and the diversity of protein localization. The model comprises two pretext tasks for individual cropped images: the first task encodes and decodes the image, extracting lower-dimensional representations, while the second task predicts the protein identity associated with the image from these representations, assuming protein localization is correlated to protein identity. The output of interest here is the 'localization encodings,' or distilled image representations, not the reconstructed image or predicted protein identity. These embeddings can be used to construct a vector database of

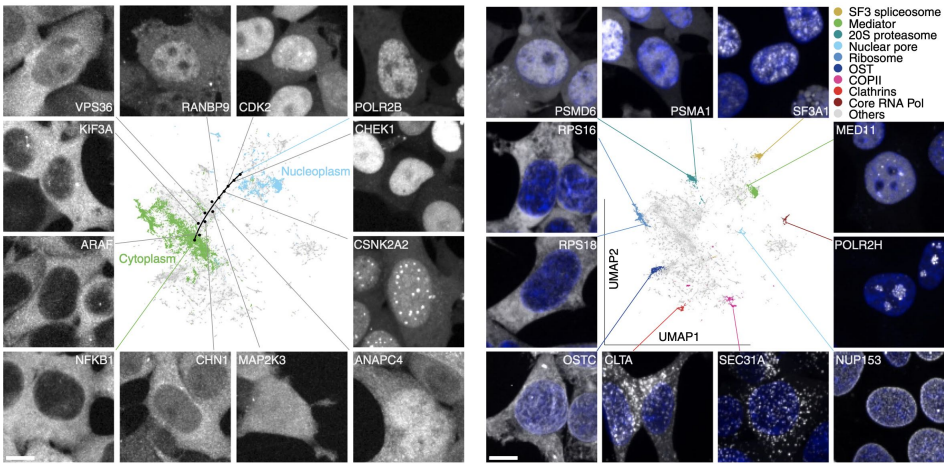


Figure 1.23: The Umap algorithm allows to visualize proteins clustered according to their localization. From [53].

protein expression within a cell, where collocalized proteins appear close to each other. It will help better understand the role of some understudied proteins, by helping to make functional predictions.

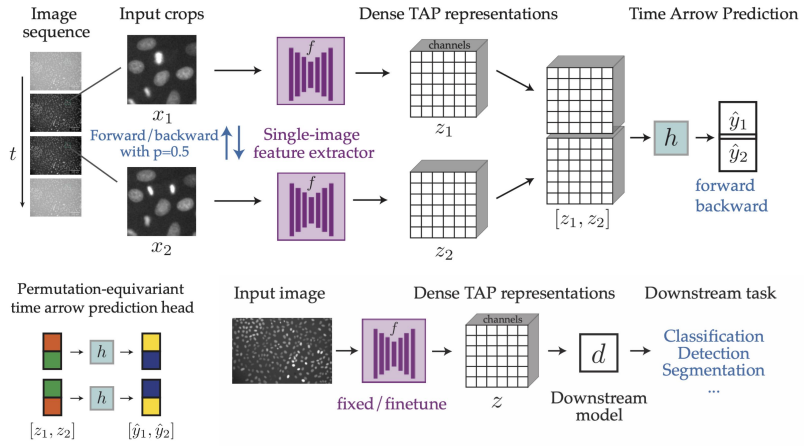


Figure 1.24: TAP architecture: A permutation-equivariant time arrow prediction neural network extracts features from biological images in order to predict the order of the frames in an image sequence. These representations are then used to learn downstream tasks. From [59].

Another interesting self-supervised learning approach in live-cell microscopy is explored with the TAP architecture [59]. This study takes advantage of the fact that biology is an out-of-equilibrium process where irreversible events, such as cell-divisions, differentiation, apoptosis or morphogenetic movements happens. These events are informative of time-reversal-invariance symmetry breaking, and can thus be used to pretrain a neural network to learn dense image representations from raw, unlabeled live-cell microscopy videos. The pretext task is predicting the correct order of time-flipped image regions via a single-image feature extractor and a subsequent time arrow prediction head. As expected, these dense representations capture inherently time-asymmetric biological processes such as cell divisions on a pixel-level. The generated representations could then be used to perform detection, segmentation or cell state classification tasks, giving good performance even when only limited ground truth an-

notations were available.

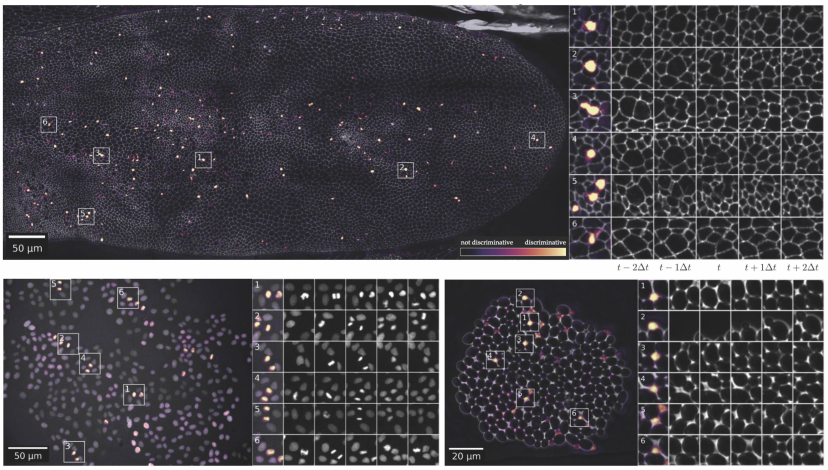


Figure 1.25: TAP returns embeddings that capture time-assymetric biological events. From [59].

Self-supervised methods holds a significant promise in deep-learning, by reducing the need of labeled data, paving the way for more efficient automated biological data analysis.

Bibliography

[1] Majid Rabbani and Paul W Jones. *Digital image compression techniques*, volume 7. SPIE press, 1991.

[2] Ming Yang and Nikolaos Bourbakis. An overview of lossless digital image compression techniques. In *48th Midwest Symposium on Circuits and Systems, 2005.*, pages 1099–1102. IEEE, 2005.

[3] Stéphane Mallat. *A wavelet tour of signal processing*. Elsevier, 1999.

[4] Erwin Schrodinger. *What is life? The physical aspect of the living cell*. At the University Press, 1951.

[5] Claude Elwood Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

[6] Ludwig Boltzmann. *Lectures on gas theory*. Courier Corporation, 2012.

[7] Edwin T Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4):620, 1957.

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

[9] Richard Aristotle et al. Categories. 2001.

[10] Aristotle Aristotle et al. *Posterior analytics*, volume 1. Clarendon Press Oxford, UK, 1994.

[11] Thomas Aquinas, Cajetanus (kardinaal.), and Armandus de Bellovisu. *De ente et essentia*. Polirom, 1998.

[12] George EP Box. Science and statistics. *Journal of the American Statistical Association*, 71(356):791–799, 1976.

[13] Michael Winding, Benjamin D Pedigo, Christopher L Barnes, Heather G Patsolic, Youngser Park, Tom Kazimiers, Akira Fushiki, Ingrid V Andrade, Avinash Khandelwal, Javier Valdes-Aleman, et al. The connectome of an insect brain. *Science*, 379(6636):eadd9330, 2023.

[14] Jeffrey De Fauw, Joseph R Ledsam, Bernardino Romera-Paredes, Stanislav Nikolov, Nenad Tomasev, Sam Blackwell, Harry Askham, Xavier Glorot, Brendan O'Donoghue, Daniel Visentin, et al. Clinically applicable deep learning for diagnosis and referral in retinal disease. *Nature medicine*, 24(9):1342–1350, 2018.

[15] Daniel Kahneman. *Thinking, fast and slow*. macmillan, 2011.

[16] Alan M Turing. *Computing machinery and intelligence*. Springer, 2009.

[17] Noam Chomsky. Rules and representations. *Behavioral and brain sciences*, 3(1):1–15, 1980.

[18] Yann LeCun, Bernhard Boser, John Denker, Donnie Henderson, Richard Howard, Wayne Hubbard, and Lawrence Jackel. Handwritten digit recognition with a back-propagation network. *Advances in neural information processing systems*, 2, 1989.

[19] Howard E Egeth and Steven Yantis. Visual attention: Control, representation, and time course. *Annual review of psychology*, 48(1):269–297, 1997.

[20] Roger BH Tootell, Nouchine K Hadjikhani, Wim Vanduffel, Arthur K Liu, Janine D Mendola, Martin I Sereno, and Anders M Dale. Functional analysis of primary visual cortex (v1) in humans. *Proceedings of the National Academy of Sciences*, 95(3):811–817, 1998.

[21] Ajay Jain. *URL: https://ajayj.com*.

[22] Piotr Bojanowski, Rémi Lajugie, Francis Bach, Ivan Laptev, Jean Ponce, Cordelia Schmid, and Josef Sivic. Weakly supervised action labeling in videos under ordering constraints. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 628–643. Springer, 2014.

[23] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021.

[24] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022.

[25] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022.

[26] Trisha Mittal, Uttaran Bhattacharya, Rohan Chandra, Aniket Bera, and Dinesh Manocha. M3er: Multiplicative multimodal emotion recognition using facial, textual, and speech cues. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 1359–1367, 2020.

[27] Tao Tu, Shekoofeh Azizi, Danny Driess, Mike Schaekermann, Mohamed Amin, Pi-Chuan Chang, Andrew Carroll, Chuck Lau, Ryutaro Tanno, Ira Ktena, Basil Mustafa, Aakanksha Chowdhery, Yun Liu, Simon Kornblith, David Fleet, Philip Mansfield, Sushant Prakash, Renee Wong, Sunny Virmani, Christopher Semturs, S Sara Mahdavi, Bradley Green, Ewa Dominowska, Blaise Aguera y Arcas, Joelle Barral, Dale Webster, Greg S. Corrado, Yossi Matias, Karan Singhal, Pete Florence, Alan Karthikesalingam, and Vivek Natarajan. Towards generalist biomedical ai, 2023.

[28] Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. Large language models encode clinical knowledge. *Nature*, pages 1–9, 2023.

[29] Wang T. Michaelos M. et al. Stringer, C. Cellpose: a generalist algorithm for cellular segmentation. *Nat Methods*, 18:100–106, 2021.

[30] Thomas R Eisenmann, Eric Ries, and Sarah Dillard. Hypothesis-driven entrepreneurship: The lean startup. *Harvard business school entrepreneurial management case*, (812-095), 2012.

[31] Carsen Stringer, Tim Wang, Michalis Michaelos, and Marius Pachitariu. Cellpose: a generalist algorithm for cellular segmentation. *Nature methods*, 18(1):100–106, 2021.

[32] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015.

[33] Uwe Schmidt, Martin Weigert, Coleman Broaddus, and Gene Myers. Cell detection with star-convex polygons. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part II 11*, pages 265–273. Springer, 2018.

[34] Kevin J Cutler, Carsen Stringer, Teresa W Lo, Luca Rappez, Nicholas Stroustrup, S Brook Peterson, Paul A Wiggins, and Joseph D Mougous. Omnipose: a high-precision morphology-independent solution for bacterial cell segmentation. *Nature methods*, 19(11):1438–1448, 2022.

[35] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2016: 19th International Conference, Athens, Greece, October 17-21, 2016, Proceedings, Part II 19*, pages 424–432. Springer, 2016.

[36] Huimin Huang, Lanfen Lin, Ruofeng Tong, Hongjie Hu, Qiaowei Zhang, Yutaro Iwamoto, Xianhua Han, Yen-Wei Chen, and Jian Wu. Unet 3+: A full-scale connected unet for medical image segmentation. In *ICASSP 2020-2020 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 1055–1059. IEEE, 2020.

[37] Martin Weigert, Uwe Schmidt, Robert Haase, Ko Sugawara, and Gene Myers. Star-convex polyhedra for 3d object detection and segmentation in microscopy. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 3666–3673, 2020.

[38] David B Brückner, Alexandra Fink, Christoph Schreiber, Peter JF Röttgermann, Joachim O Rädler, and Chase P Broedersz. Stochastic nonlinear dynamics of confined cell migration in two-state systems. *Nature Physics*, 15(6):595–601, 2019.

[39] Benoit G Godard, Rémi Dumollard, Edwin Munro, Janet Chenevert, Céline Hebras, Alex McDougall, and Carl-Philipp Heisenberg. Apical relaxation during mitotic rounding promotes tension-oriented cell division. *Developmental Cell*, 55(6):695–706, 2020.

[40] Benoit G Godard, Remi Dumollard, Carl-Philipp Heisenberg, and Alex McDougall. Combined effect of cell geometry and polarity domains determines the orientation of unequal division. *Elife*, 10:e75639, 2021.

[41] Jean-Léon Maître, Hervé Turlier, Rukshala Illukkumbura, Björn Eismann, Ritsuya Niwayama, François Nédélec, and Takashi Hiiragi. Asymmetric division of contractile domains couples cell positioning and fate specification. *Nature*, 536(7616):344–348, 2016.

[42] Jean-Léon Maître, Hélène Berthoumieux, Simon Frederik Gabriel Krens, Guillaume Salbreux, Frank Jülicher, Ewa Paluch, and Carl-Philipp Heisenberg. Adhesion functions in cell sorting by mechanically coupling the cortices of adhering cells. *science*, 338(6104):253–256, 2012.

[43] David B Brückner, Nicolas Arlt, Alexandra Fink, Pierre Ronceray, Joachim O Rädler, and Chase P Broedersz. Learning the dynamics of cell–cell interactions in confined cell migration. *Proceedings of the National Academy of Sciences*, 118(7):e2016602118, 2021.

[44] Yelena Y Bernadskaya, Haicen Yue, Calina Copos, Lionel Christiaen, and Alex Mogilner. Supracellular organization confers directionality and mechanical potency to migrating pairs of cardiopharyngeal progenitor cells. *Elife*, 10:e70977, 2021.

[45] Yonit Maroudas-Sacks, Liora Garion, Lital Shani-Zerbib, Anton Livshits, Erez Braun, and Kinneret Keren. Topological defects in the nematic order of actin fibres as organization centres of hydra morphogenesis. *Nature Physics*, 17(2):251–259, 2021.

[46] Pau Guillamat, Carles Blanch-Mercader, Guillaume Pernollet, Karsten Kruse, and Aurélien Roux. Integer topological defects organize stresses driving tissue morphogenesis. *Nature materials*, 21(5):588–597, 2022.

[47] Byungjin Hwang, Ji Hyun Lee, and Duhee Bang. Single-cell rna sequencing technologies and bioinformatics pipelines. *Experimental & molecular medicine*, 50(8):1–14, 2018.

[48] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[49] Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. Umap: Uniform manifold approximation and projection. *Journal of Open Source Software*, 3(29):861, 2018.

[50] Evelyn Fix and Joseph Lawson Hodges. Discriminatory analysis. nonparametric discrimination: Consistency properties. *International Statistical Review/Revue Internationale de Statistique*, 57(3):238–247, 1989.

[51] Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27, 1967.

[52] Shadi Tarazi, Alejandro Aguilera-Castrejon, Carine Joubran, Nadir Ghanem, Shahd Ashouokhi, Francesco Roncato, Emilie Wildschutz, Montaser Haddad, Bernardo Oldak, Elidet Gomez-Cesar, et al. Post-gastrulation synthetic embryos generated ex utero from mouse naive escs. *Cell*, 185(18):3290–3306, 2022.

[53] Hirofumi Kobayashi, Keith C Cheveralls, Manuel D Leonetti, and Loic A Royer. Self-supervised deep learning encodes high-resolution features of protein subcellular localization. *Nature methods*, 19(8):995–1003, 2022.

[54] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[55] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[56] Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

[57] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[58] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[59] Benjamin Gallusser, Max Stieber, and Martin Weigert. Self-supervised dense representation learning for live-cell microscopy with time arrow prediction. *arXiv preprint arXiv:2305.05511*, 2023.

Chapter 2

Cells and bubbles

In this chapter, we describe in greater details cell biology, the mechanics of acto-myosin cortex and its emerging properties. We explain current experimental techniques to measure cellular stresses and forces, before discussing succintely rigidity transitions in tissues. We draw an analogy with between tissues and foams, and introduce vertex models and the isoperimetric problem in 2D and 3D. We introduce the toy model of the cell-doublet, that we explore in depth, both analytically and numerically. Then we build a minimalistic implementation of gradient descent for shape optimization, that we progressively improve. Finally, we introduce our current implementation, discussing data structures for representing 3D geometries, topology transition solutions, and optimal mesh qualities.

Dans ce chapitre, nous décrivons plus en détail la biologie cellulaire, la mécanique du cortex d'actomyosine et ses propriétés émergentes. Nous expliquons les techniques expérimentales actuelles pour mesurer les contraintes et les forces cellulaires, avant de discuter succinctement des transitions de rigidité dans les tissus. Nous établissons une analogie entre les tissus et les mousses, et introduisons les modèles de vertex et le problème isopérimétrique en 2D et 3D. Nous présentons le modèle jouet du doublet cellulaire, que nous explorons en profondeur, à la fois analytiquement et numériquement. Ensuite, nous construisons une mise en œuvre minimaliste de la descente de gradient pour l'optimisation de forme, que nous améliorons progressivement. Enfin, nous introduisons implémentation réelle, en discutant des structures de données pour représenter les géométries 3D, des solutions de transition topologique, et des caractéristiques de maillage optimales.

2.1 Cell and tissue Mechanics

As the building blocks of all livings organisms, cells exhibit an array of intricate and dynamic mechanical behaviors that underpin life's diversity and complexity. Cell morphogenesis, that aims at understanding what determines their shape is a very rich and interdisciplinary field of study that bridges biology, physics and mechanics to elucidate how physical forces and mechanical properties govern a cell's behavior, morphology, and function. The architecture and dynamics of cells are regulated by a complex interplay of biological structures, that we will define and describe in the following section.

2.1.1 Mechanical aspects of the elementary components of a cell

Membranes: Lipid bilayers

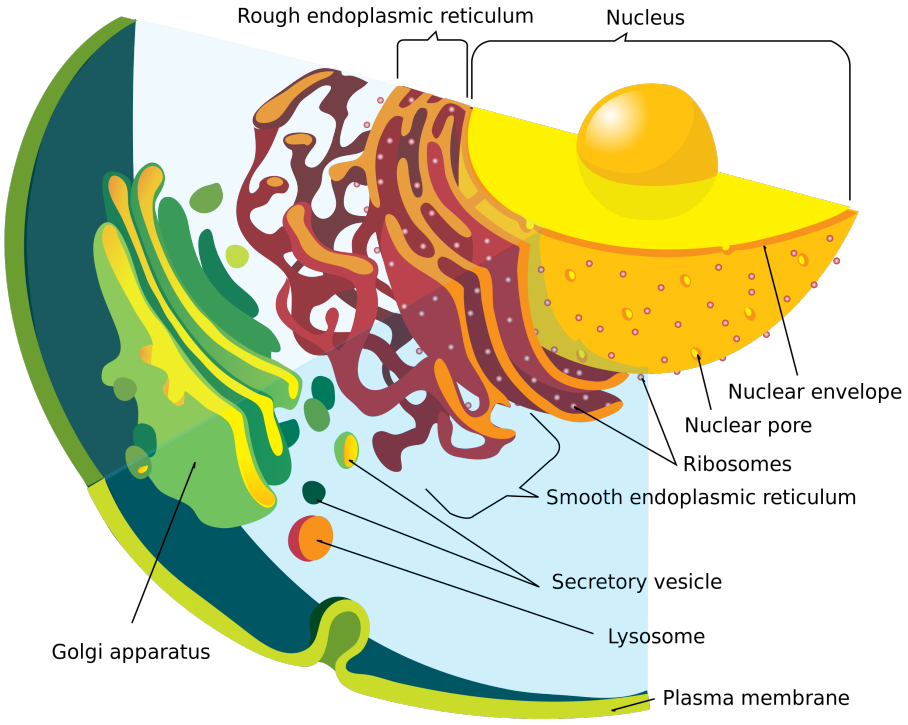


Figure 2.1: Diagram of the membrane system of an eukaryotic cell. From [1].

Cells membranes are composed of lipid bilayers, much like soap bubbles. They are composed of amphiphilic lipids: one end of a lipid molecule is attracted to water (hydrophilic) while the other end is repelled by it (hydrophobic) [2,3]. This fundamental property of the lipids results in a spontaneous formation of bilayers in aqueous environments, similar to the behavior of soap bubbles in air. These self-assembled structures display characteristic behavior such as area minimization, while being able to exhibit surprinsingly diverse geometries depending on their surrounding environment.

A closer look into the cellular structure reveals that lipid bilayers not only demarcate the outer boundary of cells but also form the encapsulating structures of nuclei, organelles, and vesicles. The cell's architectural complexity is allowed by the versatility of lipid bilayers, allowing compartmentalization of various biological functions within a single cell. Vesicles are responsible for the transport of proteins and other macromolecules between different cellular

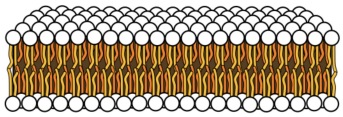


Figure 2.2: Bilayers are a stable self-organized configuration of amphiphilic lipids. From [1].

compartments. This process, also known as intracellular trafficking, is vital for maintaining cellular homeostasis and enabling responsive adaptation to environmental changes [4].

Cytoplasm

The cytoplasm is the internal medium of cells. It is often described as acting like an incompressible viscous fluid, serving as the cellular environment in which organelles reside and chemical reactions happens. It would be misleading to perceive the cytoplasm as an empty fluid where some interesting things happens from time to time; in reality, it is a crowded and dynamic environment. Within the viscous milieu of the cytoplasm are numerous organelles, each carrying out their unique functions. These encompass the nucleus, mitochondria, endoplasmic reticulum, Golgi apparatus, and lysosomes, among others. Intriguingly, the cytoplasm also hosts membraneless organelles, which lack a surrounding lipid bilayer. These unique structures have crucial roles in diverse cellular processes, including mechanics.

The nucleus

The nucleus is the largest organelle within the cell, and plays a central role in multiple cellular processes. As a repository of genetic information and the site of DNA transcription, it serves as a fundamental control center, orchestrating various aspects of cellular function.

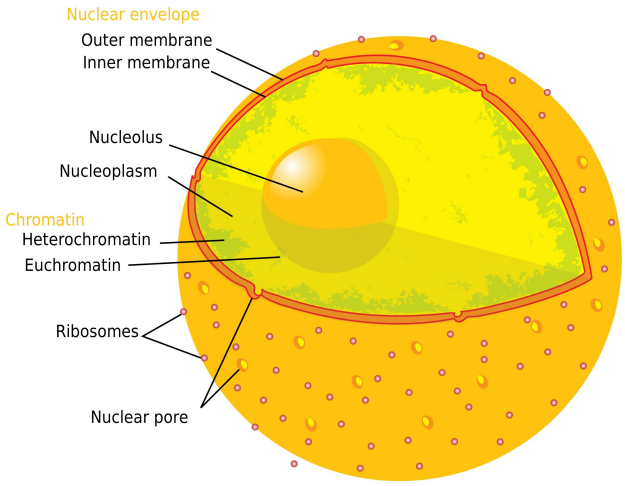


Figure 2.3: Diagram of a eukaryotic cell nucleus. From [1].

In addition to its conventional biological roles linked to transcription and regulation, the nucleus also acts as a mechanical sensor. This feature enables it to trigger specific cellular responses to mechanical stimuli, such as ameboid displacement. Ameboid movement is a rapid and adaptable form of cell motility observed in many cell types, particularly those

of the immune system, which is often stimulated in response to physical alterations in the cellular environment. In the nuclei, the cell membrane is naturally wrinkled. When the nuclei is compressed, these wrinkles disappear, leading to the activation of the cPLA$_2$-AA pathway, that recruits myosin II to the cell cortex, raising contractility and leading to ameoboid cell migration [5, 6].

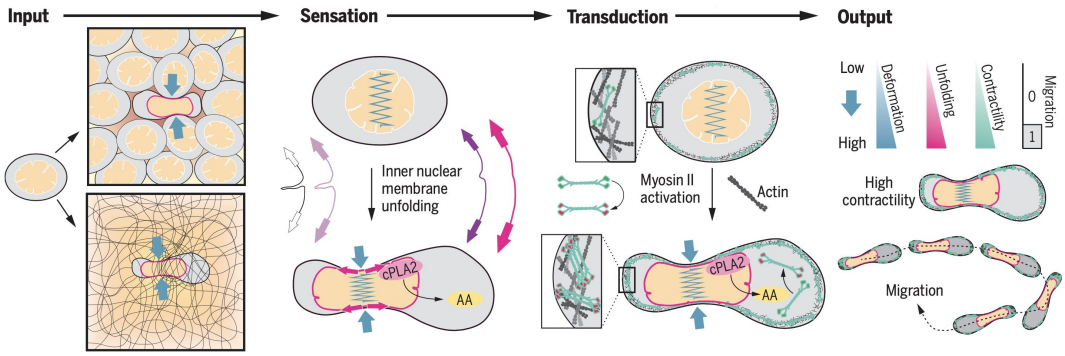


Figure 2.4: Diagram of a nuclear mechanosensation mechanism. From [6].

Furthermore, the nucleus can have a direct mechanical impact on cell shape, particularly under conditions of significant cellular deformation. In this case, the nucleus experiences compression, which subsequently influences the overall cell shape. However, the fact that the nucleus compression activates a migration pathways means that having a stable tissues may incompatible with having compressed nuclei. As a consequence, in steady tissues, nuclei are most often not compressed, and thus they do not play a mechanical role. [1]

Polymers and Entropic Springs

The cytoplasm hosts a myriad of macromolecules, including DNA, RNA, and proteins, that are formed in a sequential manner, element by element, ultimately giving rise to polymers. These polymers then fold upon themselves to acquire their unique and intricate shapes that is necessary to excert their function. Interestingly, these polymers can behave akin to what is known as 'entropic springs'. An entropic spring is a system where the force arises not from a change in energy but from a shift in entropy. When polymers are confined within a restricted space, as they are in the cell, they naturally adopt coiled configurations. This coiling increases the randomness, or entropy, of the system, thereby maximizing it [7–9]. Any external force f [10] that attempts to extend or unfold the polymer is met with a restoring force, in a response similar to the one of a mechanical spring. This entropic spring behavior can play a significant role in several cellular functions.

Figure 2.5: Perfect polymer are entropic springs. From [10].

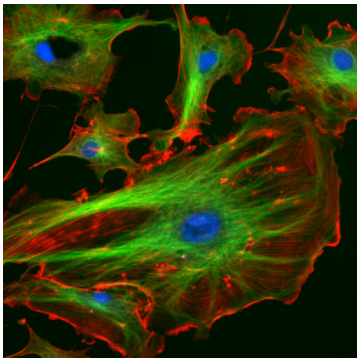


Figure 2.6: Cytoskeleton of a cell (Red: actin, Green: microtubules, Blue: nuclei)

Cytoskeleton

The cytoskeleton, as the name implies, is a skeletal network that extends throughout a cell, from the cell nucleus to the cell membrane. This network gives the cell its shape, provides mechanical resistance to deformation, and assists in numerous cellular functions such as division, polarization, apoptosis, intracellular transport and cell migration. In eukaryotic cells, the cytoskeleton is composed of three main types of filaments: microfilaments, intermediate filaments, and microtubules.

- Actin Filaments: Often referred to as microfilaments, these are the thinnest filaments in the cytoskeleton, with a diameter of around 7 nm [11]. Composed primarily of a protein named actin, these filaments are flexible and resistant to buckling, playing a pivotal role in muscle contraction, cell division, and cell motility.

- Intermediate Filaments: As their name suggests, these filaments are intermediate in size, typically 10 nm in diameter. They consist of a diverse group of proteins, including keratins, vimentin, and neurofilaments. These structures provide mechanical strength to cells and tissues, helping them withstand physical stress.

- Microtubules: These are the largest filaments, with a diameter of about 25 nm [12]. Built from tubulin proteins, microtubules are stiff hollow tubes that serve as tracks for the transport of organelles and vesicles within the cell. They also form the spindle apparatus during cell division.

These different types of filaments do not exist in isolation but interact and are regulated through a plethora of accessory proteins. These proteins modulate the assembly of filaments, link them to other cellular structures, or facilitate their interaction with motor proteins for movement along these filaments.

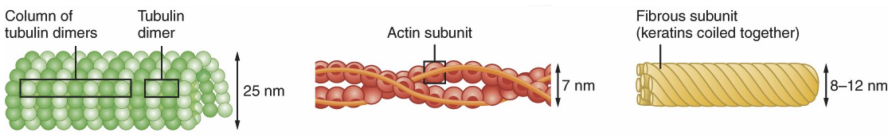


Figure 2.7: Structure of some filaments composing the cell cytoskeleton. From [13].

[1]Please read that with attention, this is important.

The turnover rate of these cytoskeletal elements, the process of continuous assembly and disassembly, varies significantly. Actin filaments have the fastest turnover, happening within seconds to minutes, followed by microtubules that take minutes to hours, and finally intermediate filaments with a turnover time ranging from hours to days. These timescales are approximate and subject to variation. For instance, different cell types, developmental stages, and cellular conditions (e.g. cell division) can all influence the dynamics of these cytoskeletal elements.

Element	Diameter	Persistence length	Turnover timescales
Actin filaments	7nm	17μm	seconds - minutes
Intermediate filaments	$\approx$ 10nm	$< 1\mu$m	$\approx$ hours - days
Microtubules	23nm	1-5mm	minutes - hours

The intricate composition, dynamic nature, and complex regulation of the cytoskeleton makes it a cornerstone of cell mechanics. Historically, the study of the behavior or intermediate filaments has been a bit neglected by researchers, constitute today a very active field of research.

2.1.2 Emerging properties of the acto-myosin cortex

Actin filaments

Actin is one of the most abundant proteins in eukaryotic cells and plays a central role in determining the shape and mechanics of cells. This versatile protein is crucial for many cellular functions, including determining cell shape, motility, intracellular trafficking, maintaining cell junctions, and possesses a rich dynamic behavior and spatial organization.

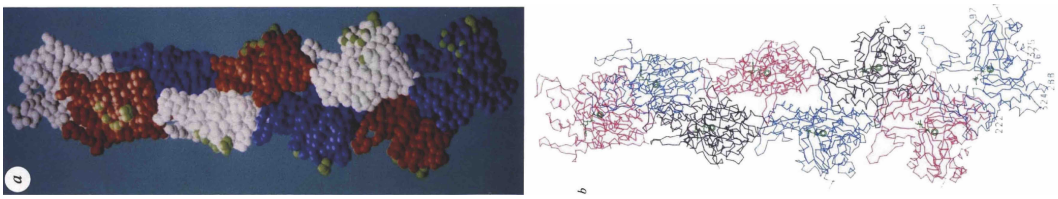


Figure 2.8: Atomic model of the actin filament. Actin monomers assembles following a double-helix structure. From [14].

Actin proteins (globular actin or G-actin) assemble into slender structures known as microfilaments or actin filaments (F-actin). These filaments have a distinctive double-helical structure, with each approximately 7 nm in diameter. Composed of two strands of polymerized actin monomers, each strand runs in the same direction, a characteristic often referred to as filament polarity. This polarity is essential to many of the functions that actin filaments perform. Actin filaments are not static entities, and undergo a continuous process known as treadmilling, in which monomers are added at the positive end (barbed end) of the filament and removed from the negative end (pointed end). This dynamic nature allows actin filaments to help cells rapidly respond to changes in their environment.

The actin cortex

At the cell surface, actin filaments interact with myosin motor proteins and a variety of actin-binding proteins that are crucial to help its nucleation, such as Arp2/3 [15]. The assembly

of these components forms a structure known as the actomyosin cortex. Because of the rapid turnover of its constituent proteins and its capacity to form a complex, branching meshwork, the actin cortex have very dynamic attributes.

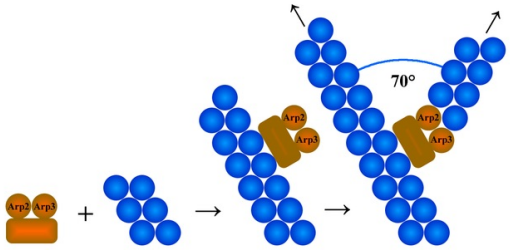


Figure 2.9: The complex Arp2/3 plays a crucial role in the formation of the actin network by allowing the creation of branching filaments.

These filaments work in conjunction with myosin motor proteins, which generate force by moving along the actin filaments, resulting in contraction and extension of the cortex. The cortex isn't a standalone structure, and is connected to the cell membrane through specific anchoring molecules, creating a stable yet flexible interface between the rigid cytoskeleton and the fluid-like cell membrane. This interconnection allows the cell to change shape while maintaining its structural integrity.

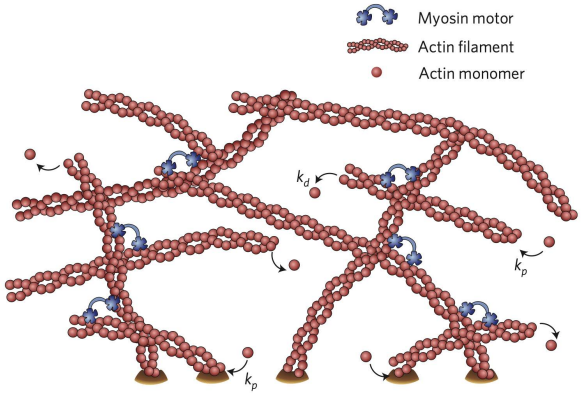


Figure 2.10: Model of the Actomyosin cortex. actin polymerizes with a rate k_p, and depolymerizes with a rate k_d. From [16].

The actin cortex also houses molecular pathways that regulate myosin recruitment. The presence of myosin in the cortex is governed by specific signaling events that dictate when and where myosin is required, ensuring the efficient and controlled generation of force within the cell. An interesting feature within the context of the actin cortex is the formation of blebs [17–19]. Blebs are bulbous protrusions of the cell membrane that occur when the cortex is detached from the membrane. The peculiar behavior of these stuctures devoid of cytoskeleton highlight the critical role the cortex plays in maintaining cell shape and responding to various mechanical stresses.

Acto-myosin cortex as a condensate

In recent years, the concept of membraneless organelles has revolutionized our traditional understanding of cellular organization. Unlike conventional organelles encased within a membrane, membraneless organelles are distinct, membrane-free structures [20]. Numerous examples of these unique organelles can be found both in the cytoplasm and nucleus, and they play crucial roles in a plethora of cellular processes.

Membraneless organelles allow to compartmentalize, and regulate biochemical reactions. They act as 'chemical reactors' within the cell, capable of sequestering or concentrating specific molecules, thereby accelerating chemical reactions and enhancing an efficient use of the cytoplasmic volume. By encapsulating different biochemical processes, they facilitate concurrent yet independent operations within the same cellular space, leading to rapid and targeted responses to changes in cellular states or external stimuli.

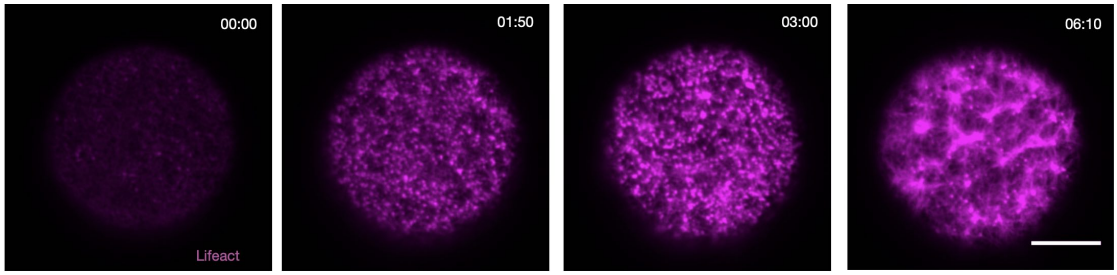


Figure 2.11: Movie of a C-elegans oocyte actin cortex maturation. F-actin, N-WASP and the ARP2/3 complex form an active micro-emulsion that catalyses actin nucleation. From [21].

One specific membraneless organelle of interest, when studying cellular mechanics, is the condensate related to acto-myosin cortex activation. Within cell membranes, it has been observed using super-resolution (TIRF) microscopy, that actin is distributed in clusters also containing other nucleators such as ARP2/3 [21]. These clusters are, apparently, understood as condensates that play a key role in the acto-myosin cortex activation process.

Corse-grained properties

The complex and dynamic behaviors of cellular components, like actin and myosin, can be understood through the lens of statistical physics. At the nanoscopic scale, the behavior of these components is well defined. At an intermediate scale, we have seen that actin forms an emulsion to catalyse its nucleation. At the larger, mesoscale level, the cortex present different emerging properties. Statistical physics provides the tools necessary to understand these divergences, employing various active hydrodynamic theories, such as active polar gels, and active nematic gels [16, 22, 23].

For instance, during cell division, the actin cortex forms a contractile ring that tightens around the cell's middle, resulting in the creation of two daughter cells [25, 26]. The cortex's role in cell migration is also paramount. By rapidly assembling and disassembling actin filaments at the cell's leading edge, the cortex can extend and retract portions of the cell, propelling it in a particular direction. In the litterature, a widely accepted approximation consist of studying the mechanical behaviour of the cell only by studying the properties of the actin cortex. Despite being a simplification, this approximation has proven successful in modeling

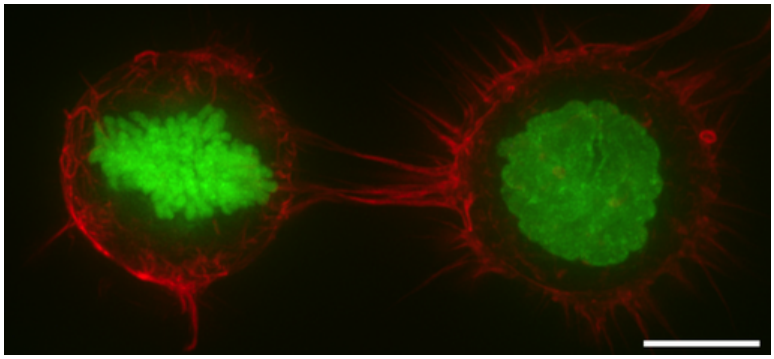


Figure 2.12: Hydrodynamic theories allow to study coarse-grained emerging properties of the actomyosin cortex. From [24].

observed cell behavior both in vivo and ex vivo. It serves as a minimal model - if it's good enough, there's no need to add more complexity.

Foam physics, On growth and form

D'Arcy Wentworth Thompson (1860-1948), a pioneer in biology, is famous for his groundbreaking work *On Growth and Form* [27] published in 1917, where he explored how physical and mathematical principles shape biological development, or morphogenesis, heralding the era of biophysics. One of his most significant contributions was the analogy of describing cells as a foam-like structure.

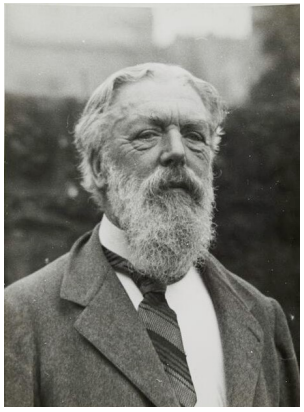
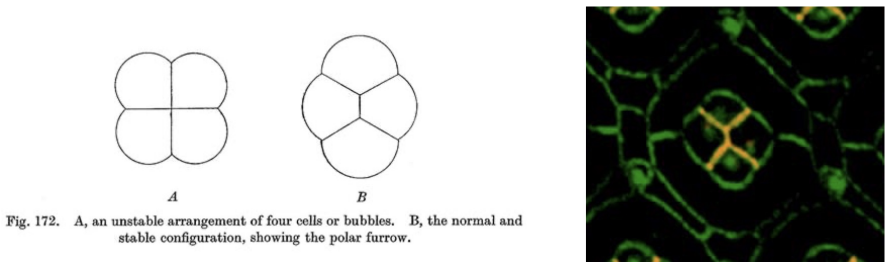


Figure 2.13: D'Arcy Thomson was appointed full professor at age 24, so he never got the chance to do a PhD. Life was easier at these times. He did some good work nevertheless.

Thompson observed that, like individual soap bubbles, cells adopt shapes that minimize their surface area. In both cases, these forms represent a state of minimal energy. He emphasized that the spatial organization of cells in a tissue and soap bubbles in a foam share striking similarities. For instance, the way that three soap bubbles meet along a line at an angle of approximately 120 degrees was found to mimic the angles at which three cells meet in epithelial tissues. He further noted that, just as soap foams rearrange to minimize total surface area when soap bubbles grow or shrink, so too do cells rearrange within a tissue in response to growth or remodeling.

This analogy was the basis for later studies that further developed this "cellular foam" model,



A B

Fig. 172. A, an unstable arrangement of four cells or bubbles. B, the normal and stable configuration, showing the polar furrow.

Figure 2.14: Left: Picture from d'Arcy Thomson book. Right: Confocal slice of cells from Drosophila's Ommatidia [28].

applying it to the understanding of tissue mechanics and morphogenesis. This assumption is of course only valid at equilibrium, excluding any dynamical aspects of morphogenesis. Thompson's foam analogy continues to provide valuable insights and has paved the way for more complex modelling approaches of cell-cortex behavior.

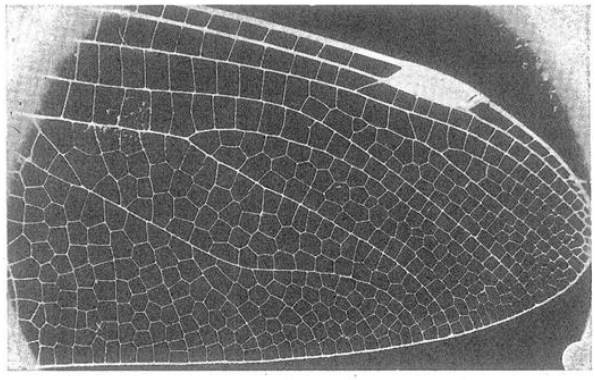


Fig. 162. Part of a dragonfly's wing.

Figure 2.15: Dragonfly wing. From [27].

If we relate the foam model to the components of the acto-myosin cortex, which is responsible for the mechanics of cell membranes, actin filaments and myosin motors play roles analogous to the molecules at the surface of a bubble. Just as intermolecular forces contribute to surface tension in a bubble, interactions between actin filaments and myosin motors generate contractile forces within the actomyosin cortex. The main postulate is that the fine-grained dynamics of the actomyosin cortex can be averaged in space and time and lead to simple coarse-grained equilibrium behavior ressembling a foam when considering cells evolving on long timescales. We will build large parts of our work using this foam physics analogy, but first we will discuss a bit other crucial properties of tissues.

Cadherins and interactions with the environnement

Within a tissue or an organism, cells are not isolated entities but are in constant dynamic interaction with their surroundings. These interactions are predominantly mediated through cell-cell junctions and cell-extracellular matrix (ECM) adhesions, which plays an integral role in maintaining the structural integrity of tissues. Cadherins are transmembrane proteins that consist of an extracellular domain for homophilic binding to cadherins on adjacent cells, a transmembrane domain, and an intracellular domain.

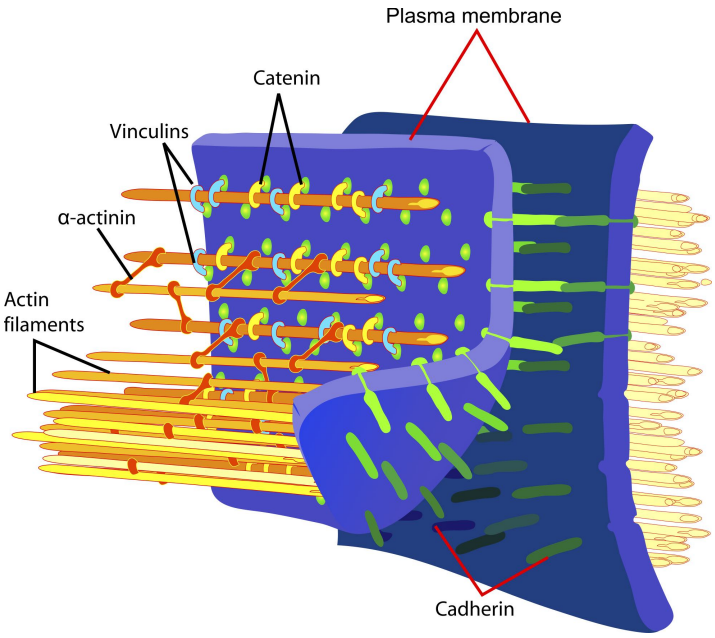


Figure 2.16: Primary associations of structural proteins at cadherin-based adherens junctions. Actin filaments are interconnected through α-actinin and associated with the membrane via vinculin. The vinculin's head domain is connected with E-cadherin through α-,β-, and γ-catenins. The vinculin's tail domain forms a bond with both membrane lipids and actin filaments [29]. From [1].

Cadherin-mediated interactions underpin the formation and maintenance of tissue and organ structure and function. Yet, the role of cadherins extends beyond merely acting as molecular 'glue'. They also serve as scaffolding for actin's contractile forces, facilitating the transmission of mechanical forces arising from actin-myosin contractility [30]. When two neighboring cells adhere, cadherins from membrane bind with each other, forming an adhesive junction. Their intracellular domains connect to the actin cytoskeleton through catenin proteins, creating a cadherin-catenin-actin complex that allows force transmission across cells. Additionally, this complex undergoes force-induced conformational changes, enhancing cell-cell adhesion and enabling cells to adapt to varying tissue tension and withstand mechanical stress.

Aside from cell-cell junctions, cells also interact with their surroundings through cell-ECM adhesions, which are critical in maintaining tissue structural integrity and directing diverse cellular functions. These adhesions generate and transmit traction forces, which can impact various aspects of cell mechanics, such as cell shape, structural organization, and migration.

Extracellular Matrix Interactions

The extracellular matrix (ECM) is an essential component of tissue and organ structure. It is a complex network composed of diverse bioactive components, including proteins such as collagen, elastin, fibronectin, and laminin, as well as polysaccharides like hyaluronic acid. These components work together to provide mechanical support, maintain cell-to-cell communication, and contribute to various physiological processes such as cellular differentiation, proliferation, migration, and tissue repair. The unique composition of the ECM within different tissues dictates its specific properties and functions, contributing to the diversity observed in living systems .

Many components of our body look like composite materials, the ECM being the matrix

tying-up all the disparate components with each other, collagen being our most important protein, accounting for $\approx 25\%$ of our dry body mass. A comprehensive understanding of cell mechanics requires to take into account these additional forces and structures, however they are very varied and interact with cells in very varied ways, and are thus complicated to model. In our studies, we focus on a simple and general description of cell mechanics, in order to discover general principles, and never take into account the role of extracellular matrix.

Volume control and osmotic pressure

Cells manage their volume through a variety of mechanisms that act to balance the inward and outward movement of water and ions, a process known as osmoregulation. This is crucial to prevent either excessive swelling or shrinking, which could lead to cell damage or even death. Cellular volume control is a complex process involving multiple mechanisms that work in concert to maintain the balance of water and ions in the cell.

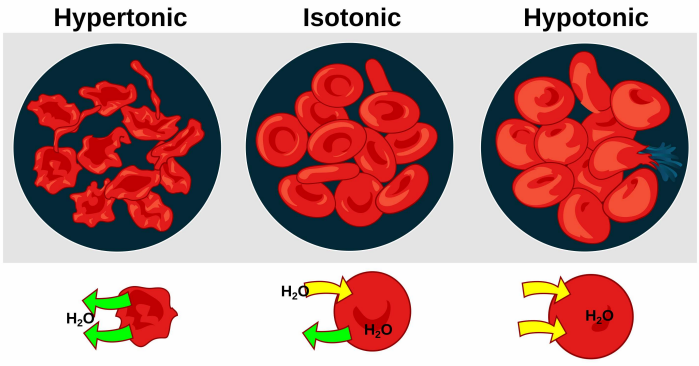


Figure 2.17: Cells swells in a hypotonic environnement, and shrink in an hypertonic environnement. From [1].

Osmotic pressure is the pressure resulting from the difference of chemical potential between different solutions. It can be illustrated by a simple experiment depicted bellow, where movement of water across a semi-permeable membrane is produced, due to a difference in solute concentration. Cells maintain their osmotic pressure through a meticulously balanced control of its concentration of ions mainly comprising sodium, potassium, calcium, and chloride. Within the cytoplasm, maintaining a controlled concentration of these ions is crucial in numerous cellular functions such as signal transduction and pH balance. Cells maintain the proper balance of ions inside its volume through the use of ion channels and pumps embedded in their membranes. These ATP-dependent specialized proteins can open or close in response to various signals and can transport ions against their concentration gradient.

Besides, the cell membrane is semi-permeable, and water can freely move across it, driven by osmosis. In addition to these channels and pumps, cells use organelles such as vacuoles to manage their water content. In plant cells, the central vacuole can store water and change its volume to adjust the cell's overall volume. In animal cells, smaller vacuoles and vesicles can move substances in and out of the cell, including water, to help maintain osmotic balance.

The pressure within the cell is equilibrated by viscous flow, which is a fast process compared to many other diffusion-driven processes within a cell. Thus in most applications, the hypothesis of a uniform pressure across the cell is very reasonable. Cell internal pressure is the consequence of the cell membrane and cortex trying to minimize their surface at a fixed

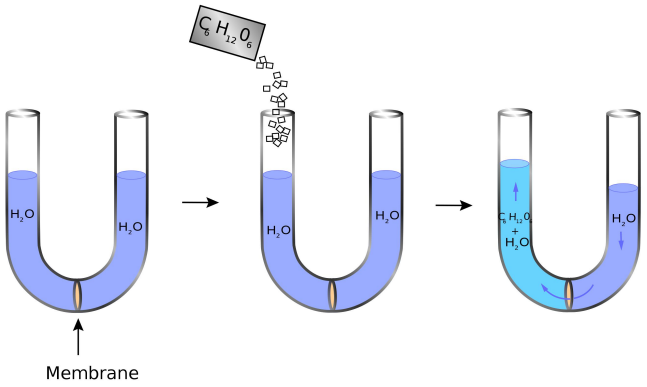


Figure 2.18: The membrane is semi-permeable and only let water flow. Adding $C_6H_{12}O_6$ ions to this solution leads to water flows until the osmotic pressure equilibrates. From [31].

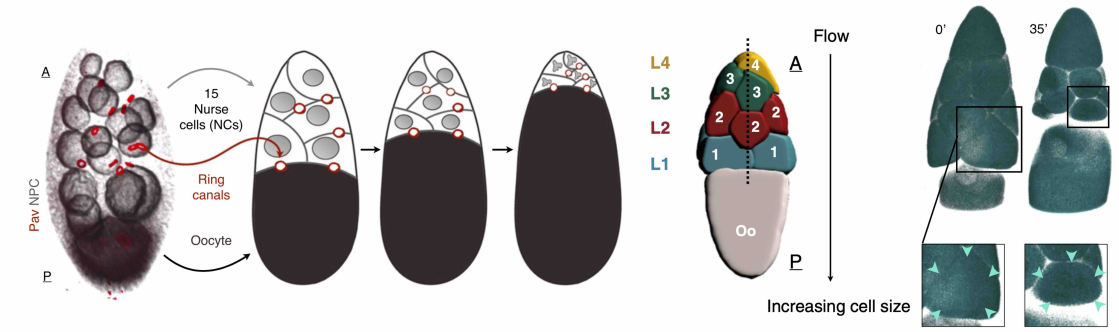


Figure 2.19: Left: Topology of the Drosophila egg chamber. Ring cells are connected with each other through ring canals, down to the oocyte. Middle: The cytoplasm flow according to Laplace law, from the smallest cells to the biggest ones. Right: 3D rendering of a confocal image of cell membranes (Clip170::GFP). From [32].

volume. A good example of this is given by the equilibrium situation, where we can model cells membranes as foam bubbles, that have a fixed surface tension, and therefore follows Laplace law. Situations similar to the two-bubble experiment [33], where the smaller bubble empties itself into the larger one due to differences in pressure, can be found in the biological world. In the Drosophila egg chamber, nurse cells (smaller cells) transfer their cytoplasmic content to the oocyte (the bigger cell), leading to its growth [32], according to these laws. The analogy between such behaviors thus reinforces the credibility of a model of internal cell pressure driven by volume control and surface energy minimization.

2.1.3 Experimental measurements of Stresses and forces

Micropipette aspiration

Micropipette aspiration is a versatile and powerful technique that has been widely used in the field of biophysics to explore and quantify the mechanical properties of individual cells and tissues. It works by applying a known negative pressure P_c through a precisely controlled pipette to a cell or a tissue fragment, which is then aspirated into the micropipette. The portion of membrane inside the pipette, initially possessing a curvature radius R_c, reaches a new mechanical equilibrium, characterized by its curvature radius R_p. As we know the applied pressure P_c, we can retrieve the surface tension via the Young-Laplace equation (applied twice to eliminate the unknown internal cell pressure):

$$\gamma_{cm} = \frac{P_c}{2}\left(\frac{1}{R_p} - \frac{1}{R_c}\right) \tag{2.1}$$

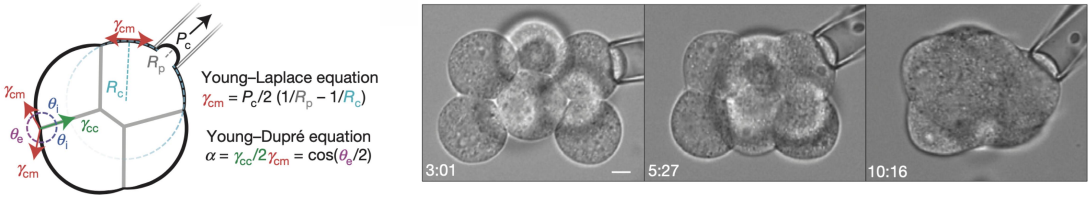


Figure 2.20: Micropipette aspiration allow to measure the surface tensions of external membranes. From [34].

The micropipettes used in this process typically range in size depending on the specific application, but they generally have an inner diameter between 20-50 micrometers. It has been observed that too small pipettes tends to detach the actin from the cortex, leading to the formation of blebs and to innacurate measurements of the surface tension. Beyond surface tension, observing the time-dependent deformation of a tissue sample during micropipette aspiration also allows for the measurement of viscosity in tissues [35].

Atomic Force Microscopy

Atomic force microscopy (AFM) is a high-resolution scanning technique that can measure the surface topography of a material at the atomic level [36–38]. The fundamental principle behind AFM is to scan the surface with a mechanical probe a mechanical probe (also known as a cantilever with a sharp tip at the end). As the cantilever is brought close to the sample surface, forces between the atoms on the probe tip and those on the sample surface cause the cantilever to deflect. These forces can be attractive or repulsive, depending on the distance between the tip and the sample. A laser beam is focused onto the backside of the cantilever, and as the cantilever moves up and down due to the atomic forces from the sample, the reflected laser beam moves correspondingly. A position-sensitive photodetector measures this movement and thus generates a signal that is proportional to the cantilever deflection, which is then converted into a distance.

AFM can be used to probe cell tensions by measuring the mechanical properties of the cells, such as their elasticity or stiffness [39, 40]. This is achieved by pressing the cantilever against the cell surface and observing the deflection. The AFM tip is moved towards the cell surface until it deforms the cell. The force exerted by the cell (in response to the deformation) and the deformation depth are then recorded, providing a measure of the cell's mechanical response.

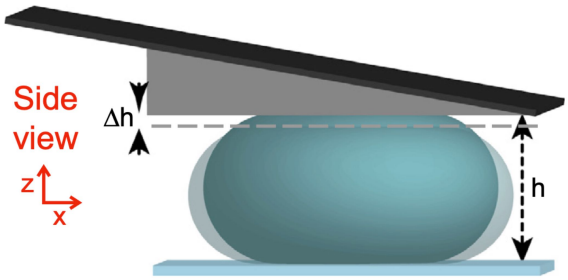


Figure 2.21: Atomic Force Microscopy allow to proble to mechanical properties of the cell. From [39].

From this, one can derive parameters like the Young's modulus (a measure of stiffness) of the cells, which is directly related to the tension within the cell. High tension corresponds to a high Young's modulus (i.e., the cell is stiffer), and low tension corresponds to a low Young's modulus (i.e., the cell is softer) [41].

Studies have shown that the mechanical properties of cells, such as their elasticity or rigidity, can affect cell differentiation [42–44]. AFM, with its ability to measure these mechanical properties at a high resolution, has been instrumental in establishing a quantitative understanding of the mechanical factors that influence cell differentiation. For instance, stem cells, which are undifferentiated and have the ability to differentiate into various cell types, have been shown to be softer than their differentiated counterparts. Such discoveries have had wide implications in regenerative medicine and tissue engineering.

Droplet Microinjection

Mechanical forces play a critical role during tissue morphogenesis and organ formation in embryos. They influence cellular behavior and can sculpt embryonic tissues, maintaining organ architecture. However, since recently, it has not been possible to measure cellular forces within developing three-dimensional tissues in vivo, making it challenging to understand how these forces shape embryonic organs.

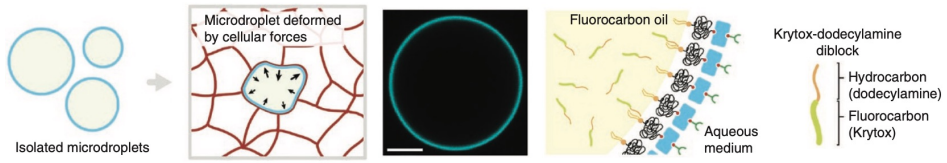


Figure 2.22: The microdroplets are fluorescent, allowing their shape deformations to be tracked in time using images from confocal microscopy. They are coated with adhesion-receptor ligands (Right: in green) to adhere to neighboring cells. From [45].

In [45], the authors developed a method involving the injection of fluorescent, cell-sized lipid microdroplets made with biocompatible oil, into the embryonic tissues. These microdroplets serve to probe the mechanics of the embryo without disrupting development. The microdroplets are coated with adhesion receptor ligands and their shape deformations can be measured using fluorescence microscopy and numerical image analysis. These shape deformations represents proxy measurements of local stresses. With this method, the authors performed the quantification of cell-generated mechanical stresses within living embryonic tissues.

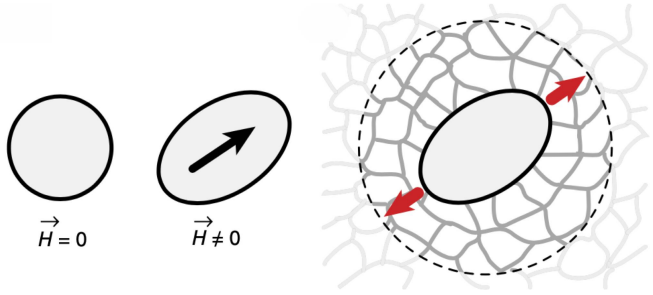


Figure 2.23: A magnetic droplet acts as a tunable mechanical actuator to probe the mechanical aspects of the microenvironnements of tissues. From [46].

In a follow-up study [46], the authors extended the method with ferrofluid droplets, that act as mechanical actuators and allow to probe the viscoelastic responses of tissues, by measuring the dynamics of the deformation of the droplets. These methods constitutes a novel way of probing quantitatively the mechanics of the cellular microenvironnement in-vivo, allowing for a better understanding on how these properties may affect cell migration, differentiation, and even tumor progression.

2.1.4 Emerging properties of tissues: Rigidity transitions

Jamming, glassy and rigidity phase transitions

Phase transitions are physical phenomena characterized by an abrupt change in a macroscopic property of the system, called the **order parameter**, under a smooth and continuous variation of a **control parameter**. The existence of observed rigidity transition, where the tissue fluidizes or rigidifies itself, lead to profound analogies with a rich array of systems where such rigidity transitions have been observed, mostly under the concept of jamming and glass transition, where, depending on the conditions, the system can exhibit both solid-like and liquid-like behavior. Although these phenomena share some similarities in the way they affect the macroscopic properties of materials, they refer to distinct processes and occur under different conditions.

Jamming refers to a transition from a flowing state to a rigid or "jammed" state that occurs in disordered systems of particles when the density of particles, or the size of the fluctuations, passes a certain threshold [47]. This concept was first introduced in [48], by analogy with traffic jam, and shortly applied to physics and material science [49, 50]. Examples of systems where jamming can occur include granular materials (such as sand or grain), foams, and emulsions. When the particles in these systems are loosely packed and the density is low, they can flow and rearrange themselves easily. However, as the density increases, the particles become so closely packed together that they become trapped in a disordered, rigid configuration and cannot move easily. This is the jammed state, and it's characterized by a sudden increase in the system's shear modulus (resistance to shear stress).

On the other hand, the glass transition is a phenomenon that occurs in amorphous (non-crystalline) materials such as polymers, certain molecular liquids, and glasses. As these materials are cooled below a certain temperature (known as the glass transition temperature, Tg), they transition from a liquid state to a glassy or vitrified state. Unlike a traditional phase transition (like freezing), this process does not involve a change in the microscopic arrangement of particles; instead, it's associated with a drastic increase in the material's viscosity

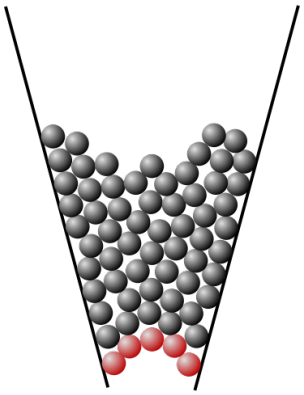


Figure 2.24: Particles too closely packed together become trapped in a disordered, rigid configuration and cannot move easily.

and a decrease in molecular mobility. The resulting glassy state is mechanically rigid but retains the disordered molecular structure of a liquid. The physical mechanism behind this unique property of glass is related to its energy landscape, which is characterized by a rough, rugged terrain filled with numerous local potential energy barriers. As the temperature decreases, these barriers become increasingly insurmountable for the constituent particles, be they atoms, ions, or molecules. As a result, the glassy system is trapped in one of these local minima, much like a mountaineer unable to descend a rugged landscape due to the presence of steep cliffs.

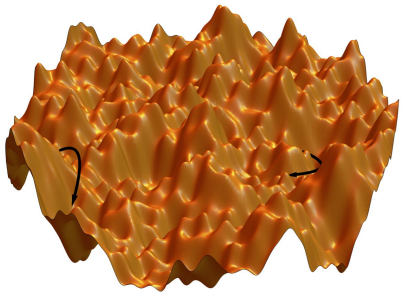


Figure 2.25: Glassy systems are characterized by a rough energy landscape. From Chiara Cammarota.

In both jamming and glass transitions, the system is characterized by an inability to reach the state of minimum energy, or ground state. While both jamming and the glass transition result in a system transitioning to a rigid state, they are fundamentally different phenomena. Jamming is induced by changes in density or applied stress and can occur at any temperature, whereas the glass transition is induced by changes in temperature and can occur at any density. Furthermore, the glass transition involves a significant change in molecular or particle mobility over a broad range of timescales, whereas jamming results in a sudden locking of the particles' positions. The framework of rigidity transitions has been applied to understand the dynamics of biological systems, such as embryonic development, tissue homeostasis, wound healing, and cancer progression [51]. Rigidity transitions are everywhere, and the application of this concept brought from statistical mechanics to biology is one of the most impressive successes of modern biophysics.

Rigidity transitions in biology

From a phenomenological point of view, tracking cells in epithelia offers many insights about their behavior. As amorphous solids, cells exhibit a "caged behavior", where they are trapped into a potential well, and escapes from it only episodically, the local thermal fluctuations being to little to allow for frequent neighbor exchange and fluidity. In a tissue, the crossing of energy barrier corresponds to a T1 topology transition, i.e a neighbor exchange between cells, that allow them to relax.

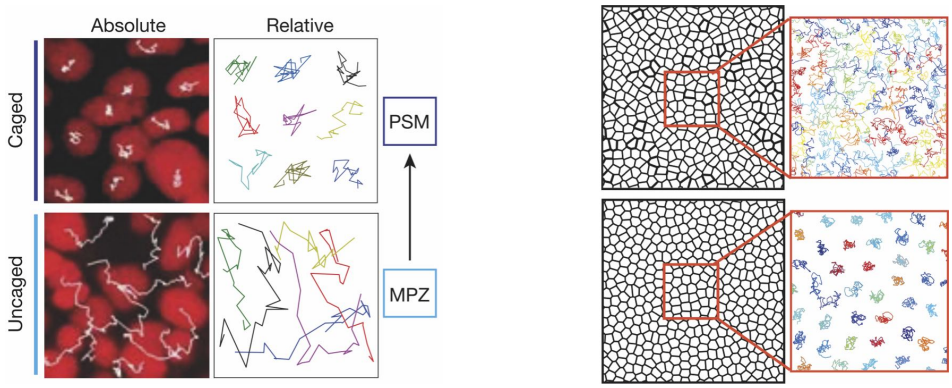


Figure 2.26: Caged behavior can be observed both in-vivo (Left: Zebrafish Pre-Somitic Mesoderm (PSM) exhibit a caged behavior while cells from the Mesodermal Progenitor Zone (MPZ) move more freely. Right: Vertex model simulations reproduce these two observed phenotypes). From [52, 53].

From a modelling point of view, this analogy between the rigidity transition in tissues and the one encountered in soft-matter physics seems quite evident in retrospective, as vertex models used in biophysics were first used in foam physics. Two seminal papers [53,54] observe a density independent transition 2D vertex models, analog to glass transitions. They characterizes the system by using an adimensional shape index for each cell, relating its perimeter P and its area A:

$$p_0 = \frac{P}{\sqrt{A}} \tag{2.2}$$

This control parameter leads to a rigid system once it crosses a certain threshold $p_0^* = 3.81$. It offers clear structural signature to relate the microscopic and the macroscopic behavior of a system, that can be tested experimentally [55].

Several new criterias, such as ones based on the graph structure and Maxwell's rigidity criterion has been proposed since then [56], and unifying theories are emerging [57, 58]. Magnetic droplets [46], that we introduced before, are able to probe the material properties of different regions of the zebrafish embryo, and have shown successfully demonstrate how rigidity transitions are able to sculpt embryonic tissues [52]. New experimental observations also corroborate the importance of fluctuations in cell sorting [**?**, 59], and offer a more comprehensive understanding of tissue mechanics. Rigidity transitions are a very good example of a collective process that happens at large scale and that really shapes the tissue, where a complex system can be described in a coarse grained manner. As in thermodynamics, we can rely on simple observables (the shape index, the graph structure) to obtain insightful informations about the coarse-grained behavior of the system.

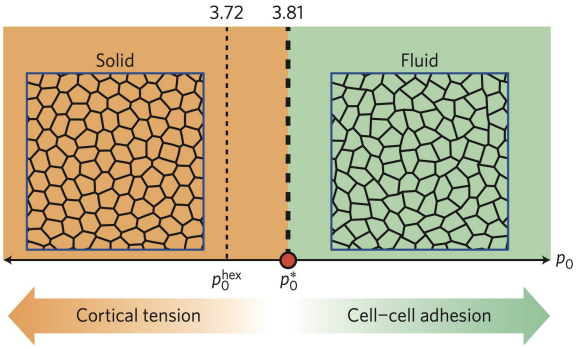


Figure 2.27: Particles too closely packed together become trapped in a disordered, rigid configuration and cannot move easily. From [53].

The principle of jamming and rigidity transitions in tissues provides an insightful perspective on the organization and dynamics of cellular systems. It is a general principle applicable to the morphogenesis of many systems. To obtain a quantitative understanding of the intricate and system-specific mechanisms that underlie development, we now have to integrate this principle with other concepts, such as morphogen gradient patterning [60].

2.2 Bubbles and Tissues

As explained before, the analogy between bubbles and tissues has been proposed as early as in the early 20th century by d'Arcy Thomson [27]. He observed that the patterns seen in clusters of cells, such as in a developing embryo or plant tissue, often resemble the patterns formed by soap bubbles, because both cells and soap bubbles are subject to similar physical constraints and forces, that is, area minimization under the constraint of volume conservation.

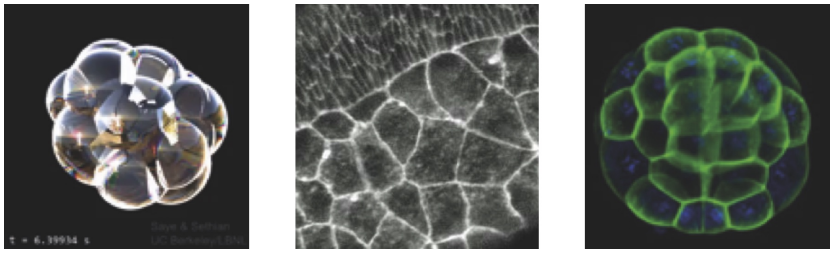


Figure 2.28: Some biological tissues ressembles closely to foam bubbles, and can be explained using the same physics. From [61, 62].

This simple analogy has been the basis of many physical approaches modelling tissues,where the shape of cells effectively minimizes a surface energy, due to surface-tension driven forces, called vertex models.

2.2.1 Vertex models

Vertex models have been a common mathematical framework used to describe the behavior of cell packings in biological tissues. They provide a way to quantify and simulate the geometric and mechanical properties of cells and tissues, making them a powerful tool in the study of biological processes like tissue morphogenesis, wound healing, and tumor growth. These models are often quite complicated to implement. Indeed, as cells are modeled as polygons, one has to take care of collisions as well as more intricate topology operations, as we will see later in the introduction.

The term "vertex model" comes from the fact that these models represent the junctions between cells (where cell membranes meet) as vertices in a graph. The edges of the graph then correspond to the cell membranes, and the spaces enclosed by these edges represent the cells themselves. This provides a simplified, abstract representation of the tissue, focusing on the arrangement and interaction of the cells.

In the first (and maybe simplest) version of a 2D vertex model, proposed by [63–66], each cell is represented as a polygon. In this model, epithelia are composed of a sheet of cells of similar height that are connected via cell-cell adhesion. This is justified by the observation of the widely studied wing disc of Drosophila. In this animal, the adhesion molecule Cadherin and components of the actin cytoskeleton are enriched apicolaterally. Cell packing geometry can be defined by the network of adherens junctions.

This network is described by a vertex model with N_c polygonal cells numbered by $\alpha \in [1, ...N_c]$ and N_v vertices, numbered $i \in [1, ...N_v]$ at which cell edges meet. Stationary and stable network configurations satisfy a mechanical force balance, which implies that at each vertex, the total force $\vec{F_i}$ vanishes. We describe these force balances as local minima of an energy function $\mathcal{E}$, that describes forces due to cell elasticity, actin-myosin bundles, and adhesion

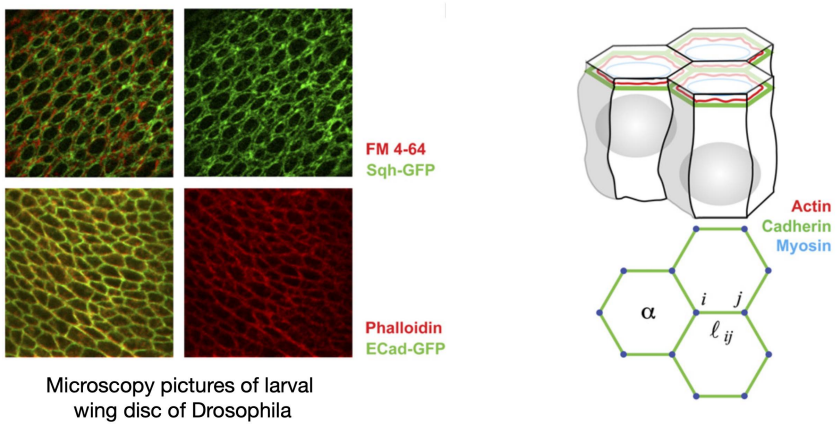


Figure 2.29: Vertex models aims at reproducing cell dynamics observed in the wing disc of Drosophila. From [63].

molecules:

$$\mathcal{E} = \sum_{\alpha} \left(\frac{K}{2}(A_\alpha - A_0)^2 + \frac{\Gamma}{2} L_\alpha^2 \right) + \sum_{<i,j>} \Lambda l_{ij} \tag{2.3}$$

The force moving the vertices positions $\vec{x}_i$ is defined by : $\vec{F}_i = -\dfrac{\partial E}{\partial \vec{x}_i}$.

A_α is the area of the polygon α, A_0 is the target area, which is determined by cell height and cell volume, and l_{ij} denotes the length $|\vec{x}_i - \vec{x}_j|$ of the junction linking vertices i and j and the sum over $< i, j >$ is over all edges.

K is an elastic coefficient, Γ describes the contractility of the perimeter L_α which could reflect, for example, the mechanics and contractility of the actin-myosin ring, and Λ_{ij} describes line tensions at junctions between individual cells. Biologically speaking, line tensions expresses the joint effect of myosins and cadherins on the contractility of the actomyosin cortex. They can be reduced by increasing cell-cell adhesion or reducing actin-myosin contractility.

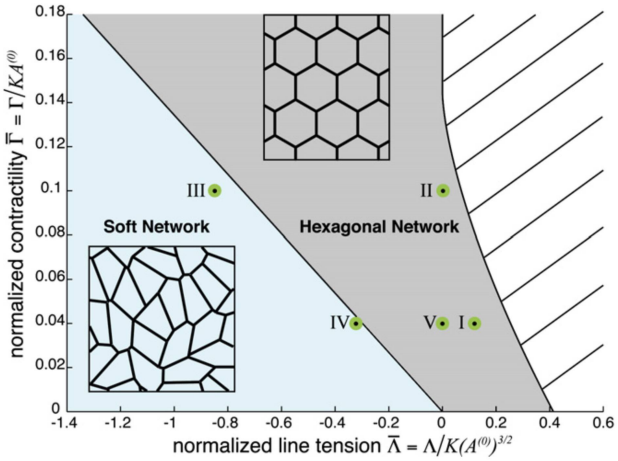


Figure 2.30: Three stable states can exist, depending on the parameters: an hexagonal network, a soft network, or an unstable collapsed network. From [63].

This simple model is able to capture the collective behavior of cells in a tissue, taking into account the physical and mechanical interactions between the cells. By adjusting the parameters in the model, one can simulate different conditions and study how these affect the behavior of the tissue. It appears that there is only three possible states depending of the line tensions and the contractility : an hexagonal network, a soft network, or an unstable collapsed network.

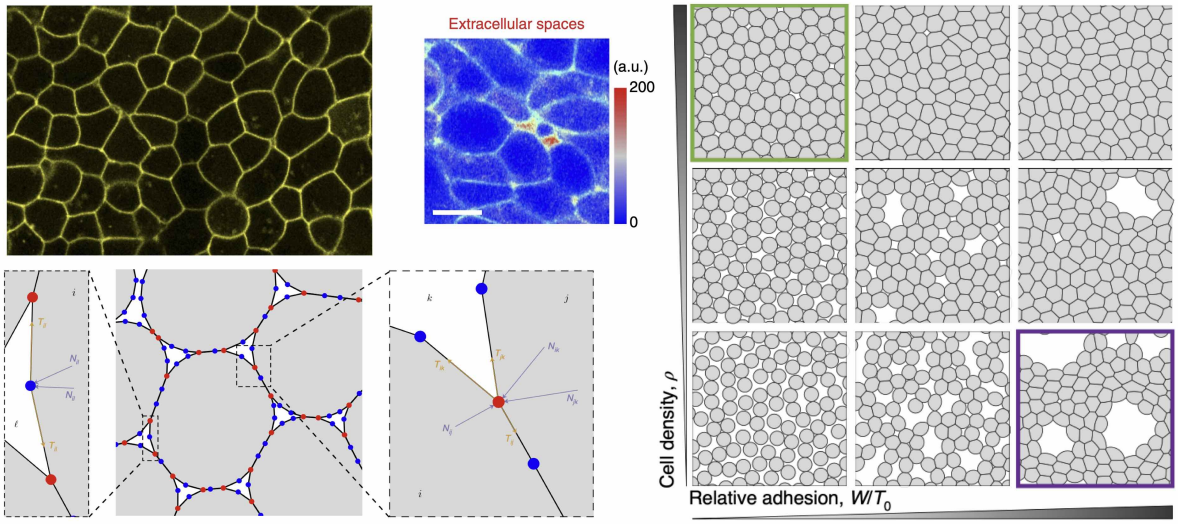


Figure 2.31: New models, closer to the biological objects studied are emerging, taking into account new features like intercellular spaces and more complex cell shapes, allowing for a more precise comparison between experiments and modeling, allowing to disentangle the effects of cell density and of adhesion in jamming. From [67].

Vertex models have been used to study a wide range of phenomena, from the folding of epithelial tissues into complex shapes during morphogenesis, to the invasive behavior of cancer cells in a tumor. They provide a bridge between the microscopic properties of individual cells and the macroscopic behavior of the tissue as a whole, making them a key tool in the field of computational biology. An essential feature of these models is their geometry. Going from the polygonal shapes to a description of cells where every cell is a manifold that is modelled via a closed polygonal line [67] (or a triangle mesh in 3D [68]), one can describe more complex cell shapes, thereby capturing the true variety of cell shapes observed in biological tissues and considering the potential role that these different shapes can play in tissue behavior and development.

Another advancement made possible by such numerical model is the incorporation of extracellular spaces [68]. Traditional vertex models often assume that cells are tightly packed with no extracellular space. In contrast, describing cells as manifolds allows to take into account the presence of extracellular spaces, which are the gaps between cells filled with extracellular fluid and matrix. This addition is important because extracellular spaces can significantly influence the mechanical properties of the tissue and the behavior of cells.
We are going to do simulations based on the minimization of surface energy, but in 3D, with a geometrical description of the cell as individual manifolds, i.e bounded volumes. This is a description as close to the reality as one can get, there is nothing beyond this: Cells are really 3D bounded volumes. Before introducing our computational model, we will first study the mathematics of the bubbles and area minimization surfaces, from which the physics of active foams emerges.

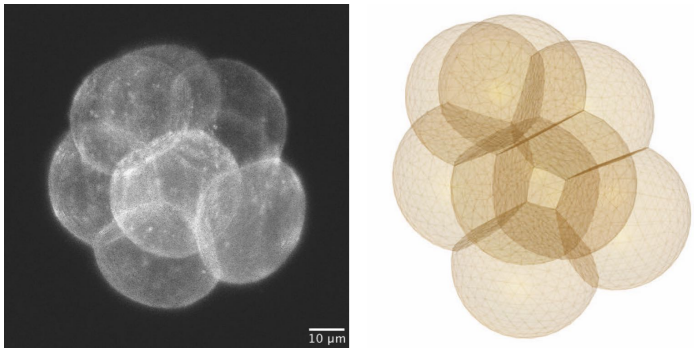


Figure 2.32: We will describe cells as 3D bounded volumes. Their geometry can be expressed in a discretized fashion using triangle meshes.

2.2.2 Introduction: Bubbles and the isoperimetric problem

Bubbles, particularly bubble clusters and foams, have been a topic of interest across a multitude of fields, including biology, physics, engineering, and even culinary arts. The fascination with bubbles dates back to Ancient Greece with the isoperimetric problem - a mathematical problem that involves finding the shape that would enclose the maximum area with a given perimeter, for which severely incomplete geometric proofs have been proposed: Archimedes 250 B.C., Zenorodus 175 B.C., Pappus 325 A.D., and continues to interest numerous mathematicians, witnessing new advances in recent years.

The proof of the isoperimetric problem was presented only two thousand years later, in the late 1830s by Jakob Steiner [69]. The three-dimensional counterpart - the problem of the sphere being the shape enclosing the maximum volume with a given surface area - was solved by Hermann Schwarz in 1884 [70]. The shapes of other multi-cell bubble clusters have been progressively determined, and the theory has witnessed several recent breakthroughs [71, 72]. However, the high mathematical difficulty of these problems spurred the development of physical simulation tools designed to characterize foams by solving an optimization problem.

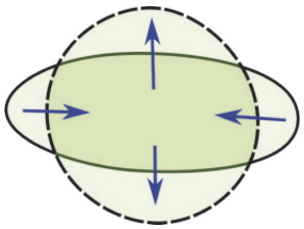


Figure 2.33: The isoperimetric problem involves finding the shape with the maximum area given a specific perimeter. Though it was first formulated in Ancient Greece, a solution was not found until more than 2000 years later.

2.2.3 The isoperimetric problem in 2D

The heuristic argument of the regular polygon

We will give a heuristic argument to justify the following lemma (This argument is based on a proof of Jakob Steiner [69]):

Among all 2n-sided polygons with the same length L , the regular 2n-gon has the largest area.

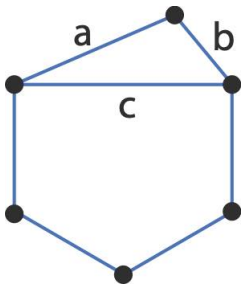


Figure 2.34: For a+b fixed, c fixed, the area of the following polygon is maximal when a=b.

Suppose that two adjacent edges had different lengths a,b. Then we could cut off this from our polygon, and consider its area A, that we can compute with Heron's formula, after having defined the half perimeter $d = (a + b + c)/2$:

$$A = \sqrt{d(d-a)(d-b)(d-c)} \tag{2.4}$$

If $a + b = l$ is fixed, we can write $b = l - a$, and compute the derivative of the area:

$$\frac{\partial A}{\partial l_1} = \frac{-\sqrt{d(d-c)(l-2a}}{2\sqrt{(d-a)(d-l+a)}} \tag{2.5}$$

Thus the area is maximized when $a = l/2$, i.e when the triangle is isocele. By symmetry, this implies that on the area-maximizing 2n-gon of length L , all the edges are equal.

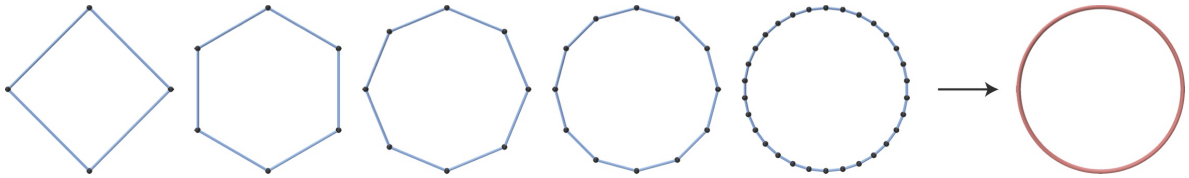


Figure 2.35: Regular N-polygons eventually converge to a circle.

As the set of regular polygons converges into a circle, the unique solution of the isoperimetric problem will be the circle.

The variational calculus

Another approach can be obtained using variational calculus [70]. Let's parametrize our curve with as a function $r : [0, 1] \mapsto \mathbb{R}^2$:

$$r(t) = (x(t), y(t)) \text{ with } \mathrm{r}(0) \ = r(1) \tag{2.6}$$

Its perimeter can be computed by integrating its coordinate:

$$P = \int_0^1 \sqrt{(x'(t))^2 + (y'(t))^2}\mathrm{d}t \tag{2.7}$$

And its area can be obtained using Green's theorem:

$$A = \frac{1}{2}\int_0^1 [xy' - x'y]\mathrm{d}t \tag{2.8}$$

We want to find, for a fixed area A, the curve r that minimizes the perimeter P. We can solve this problem using a Lagrange multiplier, by maximizing the following functional:

$$\mathcal{J}(r) = P(r) + \lambda(A(r)) \tag{2.9}$$

$$\mathcal{J}(r) = \int_0^1 \sqrt{(x'(t))^2 + (y'(t))^2} + \frac{\lambda}{2}[xy' - x'y]\mathrm{d}t \tag{2.10}$$

We call the functional $\mathcal{L}(r, r') = \mathcal{L}(\begin{bmatrix} x \\ y \end{bmatrix}, \begin{bmatrix} x' \\ y' \end{bmatrix}) = \sqrt{(x'(t))^2 + (y'(t))^2} + \frac{\lambda}{2}[xy' - x'y]$.

The Euler-Lagrange theorem states that for any function r that is stationnary for J, $t \mapsto \dfrac{\partial \mathcal{L}}{\partial r'}$ is derivable and:

$$\frac{\partial L}{\partial r} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathcal{L}}{\partial r'}\right) = 0 \tag{2.11}$$

This gives us a pair of Euler-Lagrange equations:

$$\frac{\mathrm{d}}{\lambda \mathrm{d}t}\left(y(t)\frac{x'(t)}{\sqrt{(x'(t))^2 + (y'(t))^2}}\right) = 0 \tag{2.12}$$

$$\frac{\mathrm{d}}{\lambda \mathrm{d}t}\left(x(t) - \frac{y'(t)}{\sqrt{(x'(t))^2 + (y'(t))^2}}\right) = 0 \tag{2.13}$$

Without any loss of generality, we can parametrize r with the arclength (i.e $\sqrt{(x'(t))^2 + (y'(t))^2} = 1$) and thus obtain:

$$\lambda y + x' = b \quad \text{and} \quad \lambda x - y' = a \tag{2.14}$$

We reparametrize $X = x - a, Y = y - b$ and combine both equations into a second-order linear differential equation:

$$\lambda^2 x + X'' = 0 \quad \lambda Y = -X' \tag{2.15}$$

Which gives us $X = \frac{1}{\lambda}\cos(t - t_0)$ and $Y = \frac{1}{\lambda} sin(t - t_0)$, the equation of a circle.
A minimal curve, assuming existence, is thus necessarily a circle.

Curvature and Laplace's Law

For a curve $r : \mathbb{R} \mapsto \mathbb{R}^2$ that is $\mathcal{C}^2(\mathbb{R})$, the curvature is defined at any point as the inverse of the radius of its osculating circle at this given point.

The functional J expresses the physics of a 2D bubble, which is a 1D contour that tries to minimize is interfacial energy while keeping its area fixed. The first part of the functional $\mathcal{J}$ can be interpreted as the energy of a 2D bubble, that depends on the length of the contour of the curve, its perimeter P:

$$\mathcal{E} = \gamma P, \tag{2.16}$$

where γ is the interfacial tension of the curve.

Laplace's law relates the curvature between two regions of pressure p_1 and p_2, of pressure differential $\Delta p = p_2 - p_1$ and the interfacial tension of a soap film between these two regions. If we write κ the curvature of the curve, we have:

Laplace's Law (1D curve in 2D space):

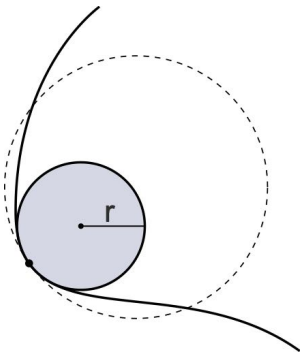


Figure 2.36: The curvature of the curve κ is the inverse of the radius of its oscillating circle, also called its radius of curvature r. From [73].

$$\boxed{\Delta p = \kappa\gamma} \tag{2.17}$$

We will follow the demonstration of [74], that we adapted to the 1D setting, that derives Laplace's law from a computation of energy established in [75]. For a curve at mechanical equilibrium, the amount of work δW needed to move the contour of the curve to an infinitesimal distance is equal to zero. This work can be expressed as the sum of two components. Let's call δW_p the work exerced by the pressures and δW_γ the work exerced by the interfacial tension.

$$\delta W_p = \int_r (p_2 - p_1)\delta\xi \mathrm{d}l \tag{2.18}$$

$$\delta W_\gamma = \int_r \gamma\delta l \tag{2.19}$$

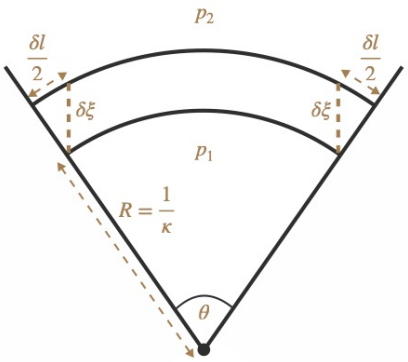


Figure 2.37: Parametrization of the curve.

We need to compute the length variation of the curve δl

$$\mathrm{d}l = R\mathrm{d}\theta = \frac{\mathrm{d}\theta}{\kappa} \tag{2.20}$$

$$\mathrm{d}l* = (\frac{1}{\kappa} + \delta\xi)\mathrm{d}\theta \tag{2.21}$$

Which allows to compute δ_l:

$$\delta l = \mathrm{d}l' - \mathrm{d}l = \delta\xi\mathrm{d}\theta = \delta\xi\mathrm{d}l\kappa \tag{2.22}$$

Thus the whole work δW writes:

$$\delta W = \int_r (p_2 - p_1)\delta\xi \mathrm{d}l - \gamma\delta\xi \mathrm{d}l\kappa = \int_r \delta\xi \mathrm{d}l(p_2 - p_1 - \gamma\kappa) \tag{2.23}$$

As this is true for any elementary deformation $\delta\xi$. The fundamental lemma of calculus of variation thus states that:

$$\Delta p = \gamma\kappa, \tag{2.24}$$

which is Laplace's Law.

2.2.4 The isoperimetric problem in 3D

In three dimensions, we need to consider surfaces, which are maps $\mathbb{R}^2 \mapsto \mathbb{R}^3$. In such surfaces, there are two principal curvatures κ_1, κ_2, and two osculating circles.

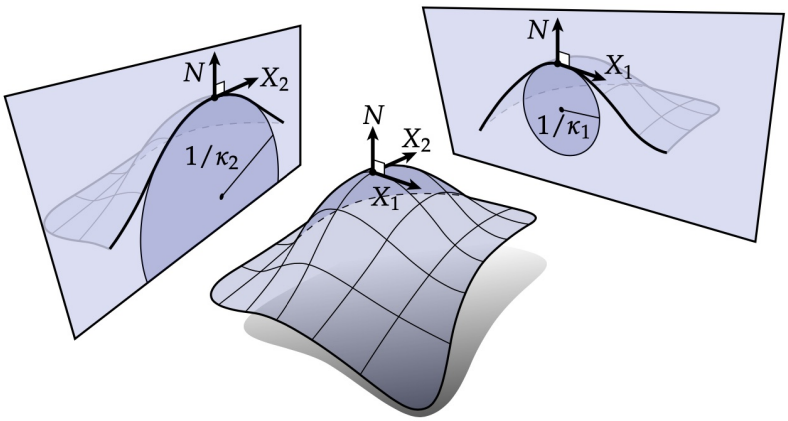


Figure 2.38: The curvature of the curve κ is the inverse of the radius of its oscillating circle, also called its radius of curvature r. From [73].

These two principal curvatures forms two quantities that we will encounter a lot during this phd:

$$\textbf{Gaussian Curvature: } g = \sqrt{\kappa_1\kappa_2} \tag{2.25}$$

$$\textbf{Mean Curvature: } h = \frac{\kappa_1 + \kappa_2}{2} \tag{2.26}$$

In 3D, as in 2D, a foam is physically defined by its surface energy $\mathcal{E}$ and its surface tension γ, and the total area of its surfaces S:

$$\mathcal{E} = \gamma S. \tag{2.27}$$

To obtain the shape of a foam, one has to minimize this surface energy, while keeping the volume of each bubble fixed. This, in essence, states that bubbles will always try to minimize their surface area.

As in 2D, we have Laplace's Law (see [74] for a derivation), that reads:

$$\Delta p = \gamma(\kappa_1 + \kappa_2) = 2\gamma h, \tag{2.28}$$

which leads to the fact that all the surfaces have a constant mean curvature. In two-dimensional constructs, mean constant curvature curves are circular arcs. However, in 3D, constant mean curvatures are richer, and are not limited to spherical caps, and can include for instance a "saddle structure", with a constant mean curvature but a varying gaussian curvature.

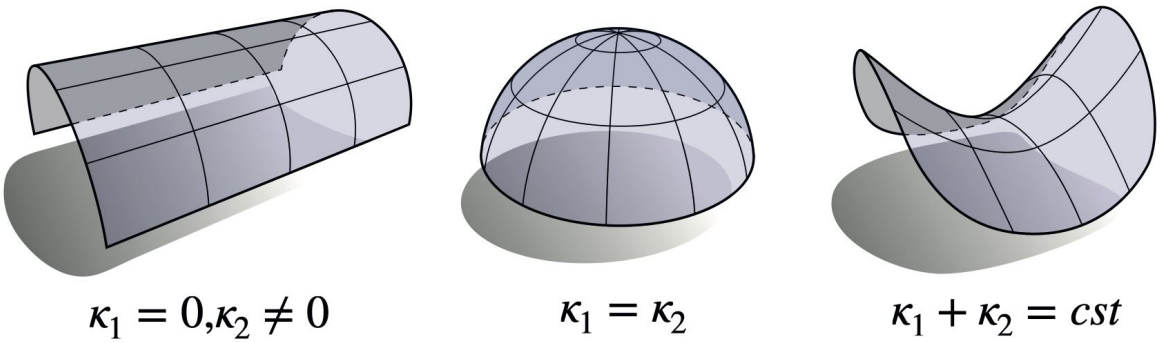


Figure 2.39: Left: A cylinder has zero gaussian curvature but non-zero mean curvature. Middle: A bubble has constant mean and gaussian curvature. Right: A saddle surface has a constant mean curvature but a varying gaussian curvature. From [73].

When the topology is richer, we can also observe other types of surfaces, as the famous catenoid structure that can be observed experimentally:

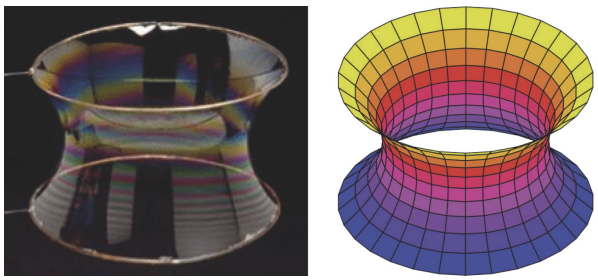


Figure 2.40: A catenoid has a constant mean curvature. From [76].

To describe the geometry of bubbles, one have to consider not just the tension in the bubble walls, but also at the lines and points where they meet, a concept known as Plateau's Laws. Plateau's Laws describe the structure of soap films and bubbles in equilibrium and can be summarized as follows:

- Soap films are always smooth and continuous.
- They have a constant mean curvature
- The edges of soap films, known as Plateau borders, meet in threes along an edge and they do so at an angle of arccos(-1/2) = 120 degrees.
- The Plateau borders meet in fours at a vertex, and they do so at the tetrahedral angle of about 109.5 degrees.

Despite this seemingly simple geometrical behaviours, dealing with clusters becomes increasingly complicated as we move from two to three dimensions, from homogeneous (same surface tension) to heterogeneous (different surface tensions), and from monodisperse (same volumes)

to polydisperse (different volumes) bubbles. In this case, one has to define the surface tension of each soap film γ_m, its surface S_m, and the volume of each bubble V_i^0, and the mechanical problem can be seen as an optimization of the following lagrangian:

$$\mathcal{L} = \sum_m \gamma_m S_m - \sum_i P_m(V_m - V_m^0) \tag{2.29}$$

Where the pressure appears as Lagrange multipliers of the volume-conservation constraint.

The third plateau law expresses the mechanical equilibrium between three surfaces that want to minimize their surface energy. In the case of non-uniform surface tensions, this has to be generalized into the Young-Dupré relationship:

$$\sum_m \vec{\gamma}_m = \vec{0} \tag{2.30}$$

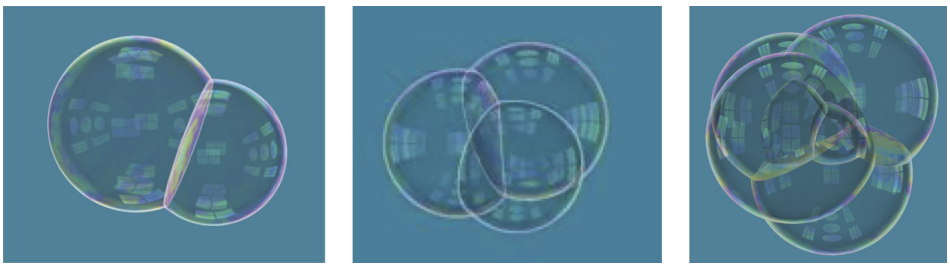


Figure 2.41: Clusters of up to four cells are composed of assemblies of spherical caps. However, for a six cells cluster, we can observe in the middle a saddle surface, of constant mean curvature that is not a spherical cap. From [77].

For clusters composed of four cells or fewer, the geometry can be described as an assembly of spherical caps. However, for five or more cells, the geometry remains uncertain. Because of the mathematical complexity of the bubbles clusters, numerical simulations have emerged as a powerful tool to advance our understanding. One such example is Sullivan's numerical proof which has exhibited a counter example of a six-cell clusters with a saddle interface [71]. Thus, in the case of six cells, interfaces are not necessarily spherical. Heterogeneous clusters add more complexity, with non-spherical and different junction types in 3D.

Understanding bubble clusters carries great importance for practical applications. It pertains to the science of packing and area/volume ratio considerations, which have implications for foam design in engineering, crystallography, and even culinary applications, such as creating the best foamy cappuccino. Understanding bubble clusters also allows us to perform force inference algorithms, which is particularly important in biological systems and one of the main advances of my PhD work.

2.3 The toy model of the doublet

One of the simplest model of heterogenous foam is the doublet. A doublet is naturally axisymmetric, which drastically limits the number of variables necessary to describe its geometry. The parametrization below is a refinement based on the model [68]. We will parametrize the system, and give state equations characterising the equilibrium of the system. Solving this system of equations will then allow us to find the geometrical parameters corresponding to the given surface tensions and volumes.

Geometric parametrization

The parametrization of the doublet has been performed by Fabrice Delbary and Hervé Turlier.

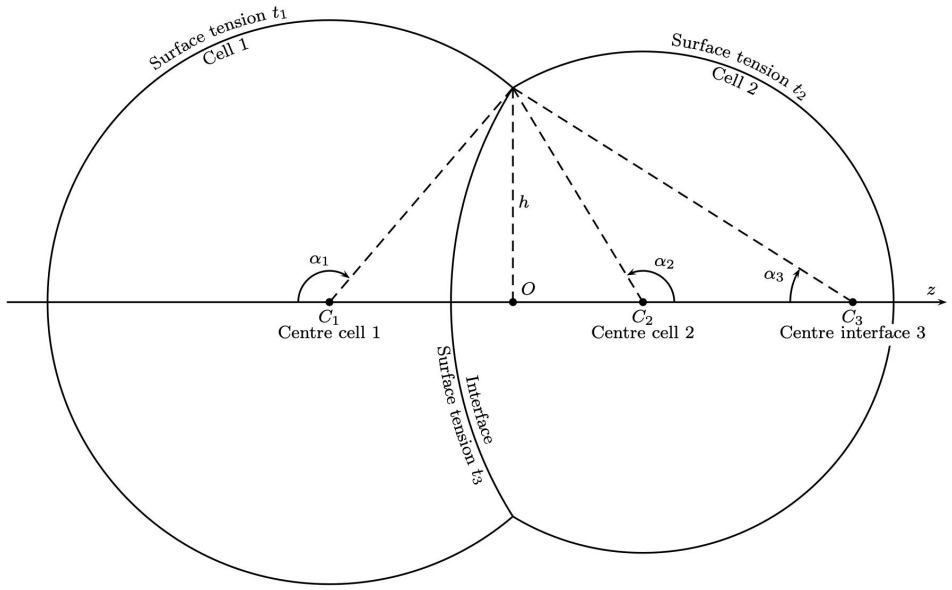


Figure 2.42: Cell doublet geometric parametrization.

We consider two spherical cells 1 and 2 of respective centre C_1 and C_2, joined by a spherical interface 3 of centre C_3 such that $C_3 \in (C_1, C_2)$. The centre of the intersection circle $\mathcal{C}$ between the cells is denoted by O and its radius by h. We assume that the cells are not tangent, that is $h > 0$. We consider an oriented (Oz) axis with origin O and unit direction vector $\mathbf{e_z} = (C_2 - C_1)/|C_2 - C_1|$. For $\alpha \in (-\pi, \pi) \setminus \{0\}$, $S_{h,\alpha}$ denotes the spherical cap of centre $z_{h,\alpha} = -h \cot \alpha$ and radius $|r_{h,\theta}|$ where $r_{h,\theta} = -h \csc \alpha$ defined by:

$$S_{h,\alpha} = \{z_{h,\alpha} + r_{h,\alpha}(\sin\theta \mathbf{e_\varphi} + \cos\theta \mathbf{e_z}) \; ; \; \mathbf{e_\varphi} \cdot \mathbf{e_z} = 0, \; |\mathbf{e_\varphi}| = 1, \; \theta \in [\pi - |\alpha|, \pi + |\alpha|]\}.$$

In other words, for $\alpha \in (-\pi, 0)$, $S_{h,\alpha}$ is the spherical cap of centre $z_{h,\alpha}$ and radius $|r_{h,\theta}|$ located on the left of the origin O and for $\alpha \in (0, \pi)$, $S_{h,\alpha}$ is the spherical cap of centre $z_{h,\alpha}$ and radius $|r_{h,\theta}|$ located on the right of the origin O. In the limit case $\alpha = 0$, we have $S_{h,0} = \mathcal{C}$. The cell 1 and 2 are respectively defined by S_{h,α_1} and S_{h,α_2} with $\alpha_1, \in (-\pi, 0)$ and $\alpha_2 \in (0, \pi)$. The interface 3 is defined by S_{h,α_3} with $\alpha_3 \in (\alpha_1, \alpha_2)$. We denote by $\vec{\Psi} \equiv (h, \alpha_1, \alpha_2, \alpha_3)$ the vector of variables parametrizing the doublet geometry, which spans the following open subset of $\mathbb{R}^4$:

$$D = \{\vec{\Psi} = (h, \alpha_1, \alpha_2, \alpha_3) \; ; \; h > 0, \; \alpha_1 \in (-\pi, 0), \; \alpha_2 \in (0, \pi), \; \alpha_3 \in (\alpha_1, \alpha_2)\}.$$

The surface tensions of each surfaces are denoted by $\gamma_1, \gamma_2, \gamma_3$. The respective volumes of cells 1 and 2 are denoted by v_1 and v_2. If we denote by A_i the areas of the surfaces i, the surface energy $\mathcal{E}$ of the system is given by

$$\mathcal{E}(\vec{\Psi}, \gamma_1\gamma_2, \gamma_3) = \sum_i A_i\gamma_i = \frac{\pi}{3}h^2 \left(\frac{\gamma_1}{1 + c_1} + \frac{\gamma_2}{1 + c_2} + \frac{\gamma_3}{1 + c_3} \right), \tag{2.31}$$

where :

$$\begin{aligned} c_1 &= \cos\alpha_1, \qquad & s_1 &= \sin\alpha_1 \\ c_2 &= \cos\alpha_2, \qquad & s_2 &= \sin\alpha_2 \\ c_3 &= \cos\alpha_3, \qquad & s_3 &= \sin\alpha_3. \end{aligned}$$

The expression for the volumes of cells 1 and 2 are:

$$v_1(\vec{\Psi}) = 2\pi h^3 \left(-\frac{s_1(2+c_1)}{(1+c_1)^2} + \frac{s_3(2+c_3)}{(1+c_3)^2} \right)$$
$$v_2(\vec{\Psi}) = 2\pi h^3 \left(\frac{s_2(2+c_2)}{(1+c_2)^2} - \frac{s_3(2+c_3)}{(1+c_3)^2} \right)$$

In the following, for sake of simplicity, we define $E = \frac{\mathcal{E}}{2\pi}$ and $v_k = \frac{\mathcal{V}_k}{2\pi}$ for $k \in \{1,2\}$ so that we have

$$E = h^2 \left(\frac{t_1}{1+c_1} + \frac{t_2}{1+c_2} + \frac{t_3}{1+c_3} \right), \tag{2.32}$$

and

$$v_1 = h^3 \left(-\frac{s_1(2+c_1)}{(1+c_1)^2} + \frac{s_3(2+c_3)}{(1+c_3)^2} \right),$$
$$v_2 = h^3 \left(\frac{s_2(2+c_2)}{(1+c_2)^2} - \frac{s_3(2+c_3)}{(1+c_3)^2} \right).$$

To impose conservation of the volume, we will find the solution of the Lagrangian:

$$\mathcal{L}(h, \alpha_1, \alpha_2, \alpha_3) = E - P_1(v_1 - v_1^0) - P_2(v_2 - v_2^0) \tag{2.33}$$

2.3.1 State equations

Deriving the Lagrangian $E - P_1 w_1 - P_2 w_2$ with respect to α_1 and α_2, we easily get the Lagrange multipliers

$$P_1 = -\frac{t_1 s_1}{3h}, \qquad P_2 = \frac{t_2 s_2}{3h}.$$

Deriving with respect to h and α_3 and replacing by the values of P_1 and P_2, we get the balance of forces at the joining points, and find back Young-Dupré relations:

$$t_1 c_1 + t_2 c_2 + t_3 c_3 = 0, \qquad t_1 s_1 + t_2 s_2 + t_3 s_3 = 0.$$

Finally, finding the possible extrema of the energy E amounts to seek for the solutions of the following system with unknown $h, \alpha_1, \alpha_2, \alpha_3$

$$t_1 c_1 + t_2 c_2 + t_3 c_3 = 0, \tag{2.34a}$$
$$t_1 s_1 + t_2 s_2 + t_3 s_3 = 0, \tag{2.34b}$$
$$h^3 \left(-\frac{s_1(2+c_1)}{(1+c_1)^2} + \frac{s_3(2+c_3)}{(1+c_3)^2} \right) = v_1, \tag{2.34c}$$
$$h^3 \left(\frac{s_2(2+c_2)}{(1+c_2)^2} - \frac{s_3(2+c_3)}{(1+c_3)^2} \right) = v_2. \tag{2.34d}$$

We see that there is a zero mode on the equations (2.34a) and (2.34b). We can thus fix one surface tension, lets say $t_3 = 1$, without any loss of generality.

2.3.2 Computation of mathematical quantities

Explicit computation

We call the vector containing the state equations the state vector, denoted $\vec{\mathcal{S}}$. It contains the state equations that we have to respect.

$$\vec{\mathcal{S}}(\vec{\Psi},\vec{\Pi}) = \begin{pmatrix} t_1c_1 + t_2c_2 + t_3c_3 \\ t_1s_1 + t_2s_2 + t_3s_3 \\ h^3\left(-\frac{s_1(2+c_1)}{(1+c_1)^2} + \frac{s_3(2+c_3)}{(1+c_3)^2}\right) - v_1^0 \\ h^3\left(\frac{s_2(2+c_2)}{(1+c_2)^2} - \frac{s_3(2+c_3)}{(1+c_3)^2}\right) - v_2^0 \end{pmatrix}$$

The state equations can thus be rewritten :

$$\vec{\mathcal{S}}(\vec{\Psi},\vec{\Pi}) = \vec{0} \tag{2.35}$$

As said in the introduction, for given parameters $\vec{\Pi}$ we can find the optimal parameters $\vec{\Psi}^*$, that minimizes all the state equations:

$$\vec{\Psi}^* : \vec{\Psi} \text{ such that } \vec{\mathcal{S}}(\vec{\Psi}^*,\vec{\Pi}) = \vec{0} \tag{2.36}$$

With this simple model, we can explore the phase space of the doublet, and see which configurations are available.

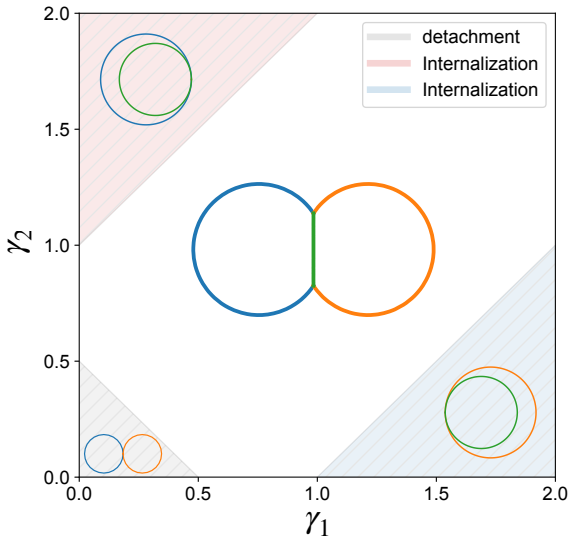


Figure 2.43: With γ_3 fixed to one, we can see several behaviours depending on the values of the two surface tensions γ_1 and γ_2: Cell-Cell contact, detachment and internalization (1 internalized by 2 or 2 internalized by 1).

We demonstrated that our model of a doublet, representing an active foam with two cells, is capable of recovering Young-Dupré and Laplace relations and present an interesting phase diagram. However, the problem's complexity was significantly reduced by the presence of a symmetry that enabled an efficient two-dimensional parametrization. Extending this approach to systems with three, four, or more cells appears to be highly challenging. Consequently, to model more complex systems, we propose employing three-dimensional simulations, in which we optimize the surface energy of a surface mesh.

2.3.3 Numerical representation of 3D geometries

Describing surfaces in a computer involves dealing with a mathematical representation of these surfaces. Though mathematically, every surface represented in a fine enough manner will eventually converge into its continuous representation, in practice, this is a highly critical choice. Indeed, choosing a way to represent the surface will impact the computation of its

geometrical properties, the way to deal with intersections, and more critically, the body of existing work one can rely on to build its simulation engine. The two primary methods for surface representation in computer graphics are explicit and implicit modeling. The choice between these two depends on the application requirements and the level of complexity involved.

Explicit Representation

In explicit representation, each point on the surface is defined directly. For instance, in 3D graphics, a common explicit representation is a mesh of polygons (usually triangles or quadrilaterals), where the vertices of the polygons correspond to points on the surface. Each polygon is defined by its vertices, and together, the polygons form the surface.

The advantage of this representation is its simplicity and the ease with which it can be rendered graphically. It is a method commonly used in computer graphics, physics simulation and video games. However, it has some disadvantages. It might be challenging to maintain the continuity and smoothness of the surface, especially when modifying it, since each polygon is defined independently. Moreover, complex operations like boolean operations can be difficult to perform.

Non-Uniform Rational Basis Splines (NURBS) are another way to describe explicitely the surface. NURBS are a type of parametric curves or surfaces, which are defined by a set of control points with associated weights. The non-uniform aspect refers to the parameterization of the curve or surface, which does not necessarily progress at a uniform rate along the shape. This allows for greater flexibility in shaping the curve or surface. The word rational refers to the use of weights, which provide even further control over the shape of the curve or surface. NURBS are widely used in computer graphics, especially in CAD (Computer-Aided Design) and 3D modeling software, due to their versatility and precision in representing complex organic and mechanical shapes.

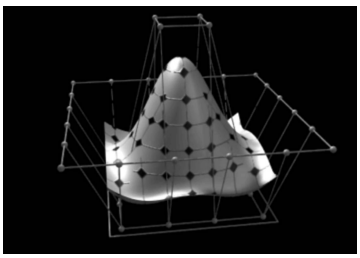


Figure 2.44: Non-Uniform Rational B-Splines (NURBS) are used for modeling smooth curves and surfaces in computer graphics and CAD. From [78].

Implicit representation

On the other hand, implicit representation defines a surface as the set of points that satisfy a particular equation. For example, a sphere can be implicitly represented as the set of points (x, y, z) that satisfy the equation $x^2+y^2+z^2-r^2 = 0$, where r is the radius of the sphere.

The most known form of implicit surfaces are level sets [79]. The level set method is a numerical technique for describing shapes using a scalar field. The fundamental idea is to represent the surface as the zero level set of a higher dimensional function, typically referred

to as the level set function $\phi(x,t)$, where x represents the spatial position and t the time. It is described by the following equation:

$$\phi(x,t) = 0 \tag{2.37}$$

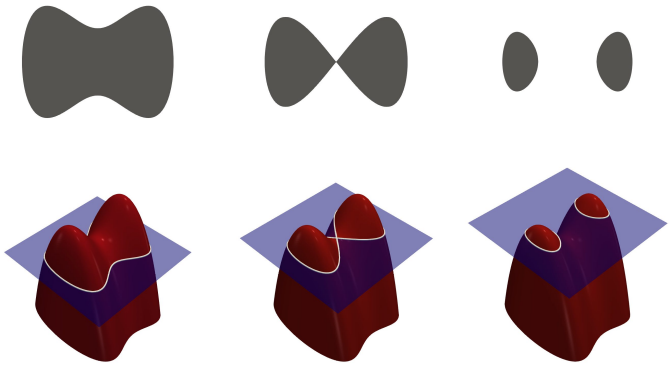


Figure 2.45: Illustration of the level set method in 2D. From [80].

A very popular level-set is the signed distance function. A signed distance function (SDF) is a function whose value at any point in space gives the shortest (Euclidean) distance to a specific surface, and is positive if the point is outside the surface, negative if the point is inside, and zero if it's on the surface. In mathematical terms, the signed distance function (SDF), denoted D, for a point p in space and a closed surface S in the 3D space, is defined as:

$$D(p) = \text{sign}(d(p)) \cdot \min_{q \in S} \|p - q\| \tag{2.38}$$

Where:

- p is a point in space,
- q is a point on the surface S,
- $\|p - q\|$ is the Euclidean distance between p and q,
- $\min_{q \in S}$ denotes taking the minimum over all q on the surface S,
- $d(p)$ is an auxiliary function that provides the unsigned distance to the surface, and
- $\text{sign}(d(p))$ is the sign function that is negative if the point p is inside S and positive if it is outside.

One advantage of implicit representations is that they are compact, and can represent smooth and complex shapes easily. Boolean operations (union, inclusion, intersection, and difference) are also straightforward with implicit representations.

However, implicit representations can be more challenging to render graphically compared to explicit representations. To draw an implicitly represented surface, we typically need to use techniques like ray tracing or convert the implicit representation into an explicit one, a process that can be computationally expensive.

Both explicit and implicit representations have their strengths and weaknesses, and the choice between them depends on the specific requirements of the task at hand. In this thesis, we will follow the approaches of previous works in the modelling of foam physics, and decide to model our cells using triangle meshes.

Surface Evolver

The Surface Evolver [81] is a software widely used in the field, that allows to model bubble clusters under various conditions. In our lab, we use a more sophisticated software that take into account different materials and allow for complex topology changes [68,82]. This software allowed us to explore various sophistications of the model, as its confinement inside a shell, or the addition of line tensions, far more rapidely than with a mathematical analysis. In the following of the introduction, we will implement a minimal version of the surface evolver, to show the user the power and the difficulties of numerical modeling of heterogeneous foams.

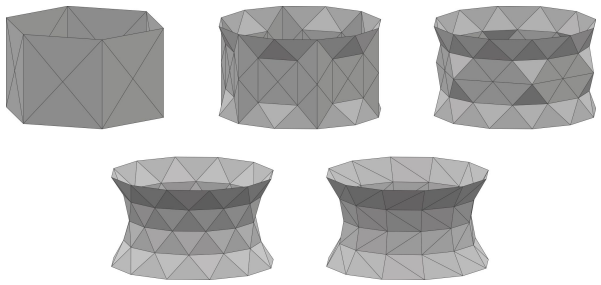


Figure 2.46: Simulation software can minimize surface energy via gradient descent and reproduce minimal surfaces observed experimentally with soap-films. From [81].

2.4 Why are bubbles not cubic ? A toy implementation of gradient descent to perform shape optimization

We will start by providing a minimal python implementation of a foam model, for a single cell described by a manifold l. Let's list all the ingredients that we need:

- A mesh on which we perform the optimization
- A way to compute the effective energy
- A way to compute the gradient of this energy with respect to the vertices
- A gradient-descent procedure

And that's it ! By far, the most challenging step is to compute the gradient of the energy. In this minimal implementation, we will show the power of reverse-mode differentiation to compute such gradients, and propose an implementation in PyTorch [83].

Motivation: A short introduction to PyTorch

Automatic differentiation (AD) is a powerful computational technique for evaluating derivatives of functions expressed as computer programs, enabling the efficient and accurate computation of gradients, Jacobians, and Hessians. First introduced by [84] and later refined by [85], AD has become a fundamental tool for optimization, machine learning, and numerical simulations. Unlike symbolic differentiation, which can lead to complex and cumbersome expressions, and finite difference methods, which suffer from numerical instability and approximation errors, AD computes derivatives algorithmically by applying the chain rule of calculus to elementary operations, while maintaining the original function's computational structure [86].

Over the past decade, the field of deep learning has seen tremendous advances, attracting the attention of major industry players like Google and Facebook, which have developed deep learning libraries to simplify the development of new architectures. One such library, PyTorch [83], provides high-level abstraction for various tasks related to optimization. We have implemented a minimal foam model in PyTorch to harness its powerful optimization capabilities. The library offers high level abstraction of many useful routines, among which:

- Automatic computation of the gradients with graph-decomposition of the computations and backpropagation
- Gradient-based optimization routines (Methods with momentum, Adam...)
- Use of graphical processing units to parallelize computations (Basically PyTorch is just a Python API to highly optimized C++/CUDA rountines)

These features make PyTorch an indispensable tool for optimizing complex functionals, and our work relies heavily on this. Automatic differentiation is typically implemented in two primary modes: forward and reverse. Forward mode computes the derivative of each intermediate variable with respect to the input variables, whereas reverse mode calculates the derivative of the output variables with respect to each intermediate variable. While forward mode is efficient for functions with few inputs and many outputs, reverse mode is more suitable for functions with many inputs and few outputs, as often encountered in machine learning [87]. By leveraging the automatic differentiation engine of PyTorch, our optimization procedures

benefit from its continuous improvements. In practice, using the right tools, such as PyTorch, is as important as having the right ideas, enabling us to iterate faster with cleaner code and bring our ideas to fruition.

Initial mesh We define our mesh by providing a list of vertices $\{\vec{x_i}\}_1^{n_v}$ and a list of triangles (also called faces), $\{t_i\}_1^{n_t}$, where each triangle t is composed of three vertices: $t = (\vec{x}_\alpha, \vec{x}_\beta, \vec{x}_\gamma)$.

2.4.1 Effective energy

When minimizing a numerical quantity, we note Φ the quantity to be minimized.

Mathematical expression

In a first approach, we do impose volume conservation using lagragian multipliers as we did for the doublet. We provide a simpler formulation of the energy that contains two terms: the surface energy, and the volume conservation penalization.

$$\mathcal{E} = \gamma_l A_l \tag{2.39}$$

$$\mathcal{E}_v = k(V_l - V_l^0)^2 \tag{2.40}$$

$$\Phi = \mathcal{E}_{eff} = \mathcal{E} + \mathcal{E}_v = \gamma_l A_l + k(V_l - V_l^0)^2 \tag{2.41}$$

To simplify the expression, we choose $\gamma_l = 1$ and $V_l^0 = 1$, the energy is thus:

$$\Phi = A_l + k(V_l - 1)^2 \tag{2.42}$$

We need to compute the volume and the area of our manifold l.

Numerical computation of the area

Each interface area and its gradient can be decomposed into a sum over its triangles, so we will start with the formula for a single triangle only, defined by its three vertices: $t = (\vec{x}_\alpha, \vec{x}_\beta, \vec{x}_\gamma)$. The area A_t of the triangle t is:

$$A_t = \frac{1}{2}||(\vec{x}_\beta - \vec{x}_\alpha) \wedge (\vec{x}_\gamma - \vec{x}_\alpha)|| \tag{2.43}$$

The total area of the manifold l is thus:

$$A_l = \sum_{t \in l} A_t \tag{2.44}$$

As the expression is based on a sum over all the triangles, numerically, we can do the computation for each triangle in a parallel fashion using numpy or torch., and perform the sum in the end. We call such implementation a *vectorialized* implementation.

Numerical computation of the volume

It is also possible to obtain a expression for the volume based on a sum over all the triangles, but its derivation, based on Stokes theorem, is less straightforward, and is detailed in the appendix A.

$$V_l = \frac{1}{6} \sum_{(\vec{x}_\alpha, \vec{x}_\beta, \vec{x}_\gamma) \in l} \det(\vec{x}_\alpha, \vec{x}_\beta, \vec{x}_\gamma) \tag{2.45}$$

2.4.2 Gradient-descent procedures

Gradient descent

Gradient descent is a fundamental optimization technique for minimizing an objective function Φ. It updates the position of each vertex $\vec{x}_i$ based on the gradient with respect to $\vec{x}_i$:

$$\vec{x}_i \leftarrow \vec{x}_i - \eta \frac{\partial \Phi}{\partial \vec{x}_i} \tag{2.46}$$

where η is a step size, called the learning rate. Choosing the appropriate learning rate is crucial, as a large learning rate can lead to faster optimization but may result in instability, while a small learning rate ensures more stable convergence at the cost of a greater number of iterations. Some methods use a scheduled decrease of the learning rate, taking smaller steps as the solution approaches the local minimum.

Momentum method

A popular variation of the gradient descent method is the momentum method, which incorporates a momentum term to accelerate convergence and reduce oscillations. The momentum method can be expressed as:

$$\vec{v}_i \leftarrow \mu \vec{v}_i - \eta \frac{\partial \Phi}{\partial \vec{x}_i} \tag{2.47}$$

$$\vec{x}_i \leftarrow \vec{x}_i + \vec{v}_i \tag{2.48}$$

where $\vec{v}_i$ is the velocity vector, and μ is the momentum coefficient. However, the momentum method has limitations, such as its sensitivity to the choice of learning rate and momentum coefficient.

Adam

To address these limitations, the Adam (Adaptive Moment Estimation) optimization algorithm was introduced by [88]. Adam combines the benefits of both momentum and adaptive learning rates, updating the position of each vertex as follows:

$$\vec{m}_i \leftarrow \beta_1 \vec{m}_i + (1 - \beta_1) \frac{\partial \Phi}{\partial \vec{x}_i} \tag{2.49}$$

$$\vec{v}_i \leftarrow \beta_2 \vec{v}_i + (1 - \beta_2) \left(\frac{\partial \Phi}{\partial \vec{x}_i} \right)^2 \tag{2.50}$$

$$\hat{\vec{m}}_i \leftarrow \frac{\vec{m}_i}{1 - \beta_1^t} \tag{2.51}$$

$$\hat{\vec{v}}_i \leftarrow \frac{\vec{v}_i}{1 - \beta_2^t} \tag{2.52}$$

$$\vec{x}_i \leftarrow \vec{x}_i - \eta \frac{\hat{\vec{m}}_i}{\sqrt{\hat{\vec{v}}_i} + \epsilon} \tag{2.53}$$

where $\vec{m}_i$ and $\vec{v}_i$ are the first and second moment estimates, β_1 and β_2 are exponential decay rates, t is the time step, and ϵ is a small constant to prevent division by zero. Adam's adaptive learning rates and momentum-based approach make it particularly well-suited for large-scale optimization problems and noisy gradients, often encountered in machine learning and other complex applications.

The Adam Zoo

Adam proved to be computationally efficient, with low memory requirements and invariance to diagonal rescaling of gradients, making it well-suited for large-scale problems involving data and/or parameters. Furthermore, the algorithm was able to handle non-stationary objectives and problems with very noisy and/or sparse gradients.

In the same paper, [88] proposed AdaMax, a variant of the Adam algorithm that relies on the infinity norm. The main difference between Adam and AdaMax lies in the way they update their second moment estimates. While Adam uses an exponentially decaying average of squared gradients, AdaMax employs an exponentially decaying average of the infinity norm of the gradients. This modification makes AdaMax more robust to extreme gradient values and provides better performance in certain optimization scenarios.

Continuing the development of Adam variants, [89] introduced AdamUniform, close to Adamax, which applies the infinity norm over time. This leads to more smooth results in the context of mesh-based optimization. Most recently, [90] proposed VectorAdam, a rotation-equivariant modification of the Adam optimization algorithm. The authors observed that traditional Adam is not rotation-equivariant for vector-valued parameters, leading to significant artifacts and biases. By accounting for the vector structure of optimization variables, VectorAdam resolves these issues and offers equivalent or even improved rates of convergence for machine learning and traditional geometric optimization problems. We refer the reader to the corresponding papers for a proper introduction and definition of these methods.

2.4.3 The torch implementation

```
import numpy as np
import torch

"""
Vertices are imported as a [nv,3] float array | Faces are imported as a [nt, 3] int array
They are converted as torch tensor, and we indicate that we will need the gradient with respect to the vertices by putting Verts.requires_grad to True
"""
Verts,Faces = np.load("Mesh_cube.npy",allow_pickle=True)
Verts = torch.tensor(Verts)
Faces = torch.tensor(Faces[:,:3])
Verts.requires_grad = True

# We provide two routines to compute areas and volumes of arrays
def compute_volume_manifold(Verts,Faces):
    Coords = Verts[Faces]
    cross_prods = torch.cross(Coords[:,1],Coords[:,2],dim=1)
    determinants = torch.sum(cross_prods*Coords[:,0],dim=1)
    Vol = torch.sum(determinants)/6
    return(Vol)

def compute_area_manifold(Verts,Faces):
    Coords = Verts[Faces]
    cross_prods = torch.cross(Coords[:,1]-Coords[:,0],Coords[:,2]-Coords[:,0],dim=1)
    Areas = 0.5*torch.norm(cross_prods,dim=1)
    return(torch.sum(Areas))

Volume_target = torch.tensor(1.0) #Equal to initial volume
```

```
k = torch.tensor(100.0) #Stiffness
optimizer =torch.optim.Adam([{'params': Verts,'lr':0.001}]) #Choice of gradient
    descent scheme

for i in range(10000):
    optimizer.zero_grad() #Put Verts.grad to 0
    Surface_energy = compute_area_manifold(Verts,Faces)
    Volume_penalization = (k+i)*(Volume_target-compute_volume_manifold(Verts,
        Faces))**2
    Effective_energy = Surface_energy + Volume_penalization
    Effective_energy.backward() #Update Verts.grad
    optimizer.step() #Update Verts according to the gradient descent schemes
```

2.4.4 Results

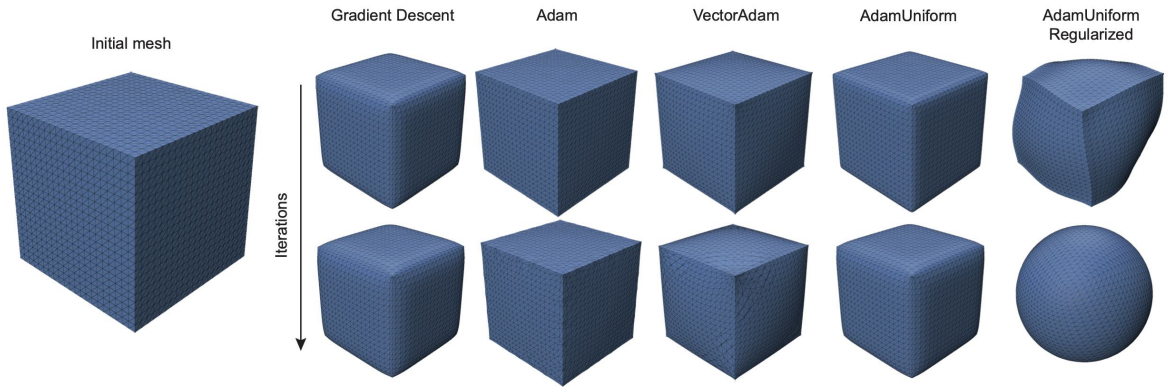


Figure 2.47: Results after 10000 iterations with different gradient descent approaches. Non-regularised approaches does not perform well, no matter the gradient-based optimization scheme.

Running the optimization reveal that non-regularized approaches of various gradient-based optimization schemes performs badly, irrespective of the optimization technique employed. The problem can be understood simply: The optimization leads to degenerate triangles, that lead to an incorrect numerical determination of the geometrical quantities of interest. We propose a solution involving gradient regularization via diffusion, which gives the expected result.

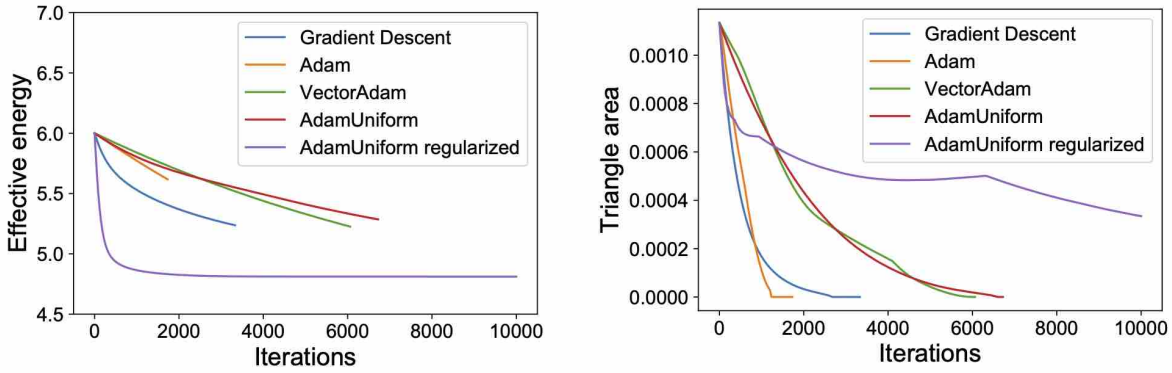


Figure 2.48: Non-regularized optimizations lead to triangles with degenerate shapes.

With this regularized solution, we end up with a final volume $V_f = 0.998$ and a final area $A_f = 4.785$. We can do a little check. The initial volume of our cube is of $V_0 = 1$, and its

area is $A_0 = 6$. If we end up with a perfect sphere, the radius R of the sphere is related to the volume and the area via the formulas:

$$V_s = \frac{4}{3}\pi R^3 \text{ and } A_s = 4\pi R^2 \tag{2.54}$$

With $V_s = 1$, we have $R = \left(\frac{3}{4\pi}\right)^{\frac{1}{3}}$ and thus the final area should be $A_s = 4\pi\left(\frac{3}{4\pi}\right)^{\frac{2}{3}} = (4\pi)^{\frac{1}{3}}\,3^{\frac{2}{3}} \approx 4.836$. Quite close ! We note that on our implementation both the area and the volumes are a little lower than the exact values for a sphere, because the volume conservation is not a rigid constraint. The error obtained in the end could be lower but raising the term k in the volume penalization energy, but this raises another problem which is: Is the simulation physical ? In practice, if we neglect any source of noise (starting with the thermal noise), a biological system would just fall into a local minimum of energy. Our gradient-based optimization scheme leads us to a local minmum as well, but we are not sure that they are the same. However here, in the case of a single cell, the local minimum is also the global minimum.

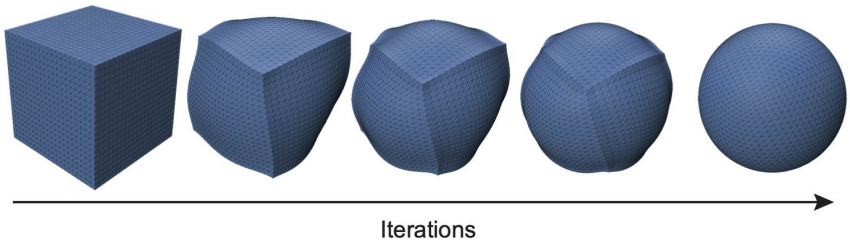


Figure 2.49: Our regularised optimization converges to the expected bubble shape: a perfect sphere.

2.5 Improvements of the implementation

2.5.1 Surface energy and gradients

We provide a more appropriate way of formulating the problem, where the volume is conserved at each iteration of the optimization [68,81]. We find a local minimum of the energy by doing a gradient-based optimization. We consider a mesh describing one cell. The surface energy and Lagrangian function are defined as follows:

$$\mathcal{E} = \gamma_l A_l \tag{2.55}$$

$$\mathcal{L} = \mathcal{E} - p_l(V_l - V_l^0) \tag{2.56}$$

Where p_l and V_l^0 are respectively the pressure and the target volume value of the cell. As before, we choose $\gamma_l = 1$ and $V_l^0 = 1$, thus the lagrangian is:

$$\mathcal{L} = A_l - p_l(V_l - 1) \tag{2.57}$$

From the Lagrangian function, one can calculate the force $\vec{f_k}$ on each vertex of the mesh $\vec{x} \in \{\vec{x}_\alpha\}_{\alpha=1}^{n_v}$ as follows

$$\vec{f} = -\frac{\partial \mathcal{L}}{\partial \vec{x}} = -\frac{\partial A_l}{\partial \vec{x}} + p_l \frac{\partial V_l}{\partial \vec{x}} \tag{2.58}$$

We note that when the mesh is at equilibrium, we have $\vec{f} = \vec{0}$, and thus have, for each vertex $\vec{x}$:

$$\frac{\partial A_l}{\partial \vec{x}} = p_l \frac{\partial V_l}{\partial \vec{x}}, \tag{2.59}$$

that relates the surface tensions and the pressures. We will later demonstrate that they lead to Laplace-like equations in the discrete setting.

2.5.2 Volume-conservation implementation

We define three scalars O, Q and R:

$$\mathrm{O} = \left(\sum_i \frac{\partial V_l}{\partial \vec{x_i}} \cdot \frac{\partial V_l}{\partial \vec{x_i}} \right) \tag{2.60}$$

$$\mathrm{Q} = \left(\sum_i \frac{\partial \mathcal{E}}{\partial \vec{x_i}} \cdot \frac{\partial V_l}{\partial \vec{x_i}} \right) \tag{2.61}$$

$$\mathrm{R} = V_l - V_l^0 \tag{2.62}$$

Force correction

If we displace each vertex using a gradient descent, to the first order, the volume change induced by each vertex displacement will be:

$$\delta V_{\vec{x}_i} = \vec{f}_{\vec{x}_i} \cdot \frac{\partial V_l}{\partial \vec{x}_i} \tag{2.63}$$

And thus the total volume change will be :

$$\delta V_l = \sum_i \delta V_{\vec{x}_i} = \sum_i \vec{f}_{\vec{x}_i} \cdot \frac{\partial V_l}{\partial \vec{x}_i} \tag{2.64}$$

We can anihilate this volume change but changing the force correction by choosing instead as a force:

$$\vec{f}_{\vec{x}_i} - \frac{\left(\sum_j \vec{f}_{\vec{x}_j} \cdot \frac{\partial V_l}{\partial \vec{x_j}} \right) \cdot \frac{\partial V_l}{\partial \vec{x}_i}}{\left(\sum_i \frac{\partial V}{\partial \vec{x_i}} \cdot \frac{\partial V}{\partial \vec{x_i}} \right)} \tag{2.65}$$

The displacement for each vertex will be

$$\delta V_{\vec{x}_i} = \left(\vec{f}_{\vec{x}_i} - \frac{\left(\sum_j \vec{f}_{\vec{x}_j} \cdot \frac{\partial V_l}{\partial \vec{x_j}} \right) \cdot \frac{\partial V_l}{\partial \vec{x}_i}}{\left(\sum_i \frac{\partial V}{\partial \vec{x_i}} \cdot \frac{\partial V}{\partial \vec{x_i}} \right)} \right) \cdot \frac{\partial V_l}{\partial \vec{x}_i} \tag{2.66}$$

and thus the total displacement will be:

$$\delta V_l = \sum_i \delta V_{\vec{x}_i} = \sum_i \left(\vec{f}_{\vec{x}_i} - \frac{\left(\sum_j \vec{f}_{\vec{x}_j} \cdot \frac{\partial V_l}{\partial \vec{x_j}} \right) \cdot \frac{\partial V_l}{\partial \vec{x}_i}}{\left(\sum_i \frac{\partial V}{\partial \vec{x_i}} \cdot \frac{\partial V}{\partial \vec{x_i}} \right)} \right) \cdot \frac{\partial V_l}{\partial \vec{x}_i} = 0 \tag{2.67}$$

If we define $F_l = \mathrm{QO}^{-1}$, the force correction is simply:

$$\vec{f}_{\vec{x}} \leftarrow \vec{f}_{\vec{x}} - F_l \cdot \frac{\partial V_l}{\partial \vec{x}} \tag{2.68}$$

Motion correction

The motion correction has for goal to correct the volume if it is not equal to the target volume. This corrects for drift in volume, which happen for optimizations with many time-steps because of numerical noise and because the force correction is only first order.
Lets define $M = \mathrm{QO}^{-1}$, This motion correction will be implemented by displacing the vertices:

$$\vec{x} \leftarrow \vec{x} - M_l \cdot \frac{\partial V_l}{\partial \vec{x}} \tag{2.69}$$

.
M is expressed as :

$$M = \frac{V_l - V_l^0}{\left(\sum_i \frac{\partial V_l}{\partial \vec{x_i}} \cdot \frac{\partial V_l}{\partial \vec{x_i}}\right)} \tag{2.70}$$

at the first order, the volume change is thus:

$$\delta V_l = \sum_i \delta V_{\vec{x}_i} = -\sum_i \left(\frac{\partial V_l}{\partial \vec{x}_i} \frac{V_l - V_l^0}{\left(\sum_j \frac{\partial V_l}{\partial \vec{x_j}} \cdot \frac{\partial V_l}{\partial \vec{x_j}}\right)} \right) \cdot \frac{\partial V_l}{\partial \vec{x}_i} = V_l^0 - V_l \tag{2.71}$$

2.5.3 Gradient descent with a line-search

The gradient descent procedures described before were interesting, and all useful in practice, but we rely on a more robust procedure, that is called a line search: It is a naive gradient descent, with the peculiarity that the learning rate l_r will be adapted (lowered or raised) for each step: We start from a initial learning rate, and try to update the position of the vertices.

- If our new energy is bigger than the original one, our learning step is two big, and thus we will decrease it progressively until the new energy is lower than the original one
- If our new energy is lower than the original one, we can try to increase our learning step by multiplying it with a factor β, as long as our new energy continues to decrease.

2.5.4 The new implementation for a single cell

Let's implement all these elements in Python. As before, we rely on PyTorch to compute the gradient of the energy. To implement the volume control, we need to define the matrices P, Q, R, that are all 1×1 matrices (scalars). As before, we obtain the gradients via automatic differentiation, but this time we use another pytorch routine, torch.autograd.function.jacobian, that gives the jacobian of a given function. We provide a short implementation of our algorithm (80 lines of codes for a physics-simulation algorithm is really not much).

```
import numpy as np
import torch

"""
Vertices are imported as a [nv,3] float array | Faces are imported as a [nt, 3] int array
They are converted as torch tensor, and we indicate that we will need the gradient with respect to the vertices by putting Verts.requires_grad to True
"""
Verts,Faces = np.load("Mesh_cube.npy",allow_pickle=True)
Verts = torch.tensor(Verts)
Faces_label = torch.tensor(Faces[:,3:])
Faces = torch.tensor(Faces[:,:3])
```

```
Verts.requires_grad = True

# We provide two routines to compute areas and volumes of arrays
def compute_volume_manifold(Verts,Faces):
    Coords = Verts[Faces]
    cross_prods = torch.cross(Coords[:,1],Coords[:,2],dim=1)
    determinants = torch.sum(cross_prods*Coords[:,0],dim=1)
    Vol = torch.sum(determinants)/6
    return(Vol)

def compute_surface_energy(Verts,Faces):
    Coords = Verts[Faces]
    cross_prods = torch.cross(Coords[:,1]-Coords[:,0],Coords[:,2]-Coords[:,0],
        dim=1)
    Areas =  0.5*torch.norm(cross_prods,dim=1)
    return(torch.sum(Areas))

with torch.no_grad():
    Volume_target = compute_volume_manifold(Verts,Faces) #Equal to initial
        volume

for i in range(10):
    ###
    #Computation of the gradient with respect to the energy and the volumes
    ###
    E_grad = torch.autograd.functional.jacobian(lambda x:
        compute_surface_energy(x,Faces),Verts)
    V_grad = torch.autograd.functional.jacobian(lambda x:
        compute_volume_manifold(x,Faces),Verts)

    ###
    #Computation of the volume control corrections
    ###
    O = torch.sum(torch.sum(V_grad*V_grad,dim=1))
    Q = torch.sum(torch.sum(E_grad*(V_grad),dim=1))
    R = (compute_volume_manifold(Verts,Faces)-Volume_target)
    F = Q/O
    M = R/O

    ###
    #Application of the force and volume control corrections:
    ###
    Verts.grad=E_grad - F*V_grad
    with torch.no_grad(): Verts-=V_grad*M

    ###
    #Line_search step:
    ###
    Current_surface_energy = compute_surface_energy(Verts,Faces)
    lr = 0.01
    beta = 1.5
    Verts_new = Verts - lr*Verts.grad
    New_surface_energy = compute_surface_energy(Verts_new,Faces)
    nsteps = 0

    if New_surface_energy<=Current_surface_energy:
        while New_surface_energy<Current_surface_energy:
            Current_surface_energy = New_surface_energy
            lr = lr*beta
            Verts_new = Verts - lr*Verts.grad
            New_surface_energy = compute_surface_energy(Verts_new,Faces)
```

```
            nsteps+=1
    else :
        while Current_surface_energy<New_surface_energy:
            lr = lr*(1/beta)
            Verts_new = Verts - lr*Verts.grad
            New_surface_energy = compute_surface_energy(Verts_new,Faces)
            nsteps-=1

    ###
    #Gradient descent step
    ###
    with torch.no_grad(): Verts-= lr*Verts.grad
```

In conclusion, PyTorch proves to be an advantageous tool for quickly testing new concepts, given its ease of use and versatility. Generalizing this approach to several cells is not very complicated. However, when aiming for robust solutions, we must turn to more sophisticated approaches. Two significant challenges emerge in this context - managing clusters containing multiple cells and handling collisions.

2.6 Real-life implementation

2.6.1 Data-structure: Representing several cells in 3D

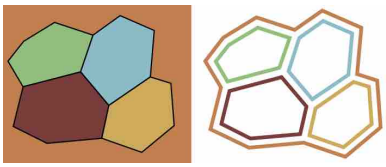


Figure 2.50: Each triangle has two labels: inside/outside materials. By convention, the faces normals (defined by the right-hand rule) are oriented towards the inside. From [82].

A bounded volume can be defined by a manifold mesh thas as a consistent orientation of triangles, where each triangle's vertex ordering and the right-hand rule define a normal direction pointing towards the interior. Representing multiple bounded volumes with a non-manifold mesh necessitates an n-ary material classification, because mesh normals are insufficient. Following earlier works by [81, 82], each material is assigned a unique integer label, and these labels are applied to both the front and back of each triangle (i.e., each half-face has a label). In a comparable 2D polygon example, colors represent labels. The left image illustrates the material regions (including the "exterior" orange region), while the right image displays the corresponding mesh with edges having two labels each. This representation necessitates consistent material labels: all half-faces enclosing a closed region must have identical labels. By maintaining and leveraging this expanded concept of mesh orientation, watertight regions can be defined.

In this context, the term "region" refers to a closed volume of space, while "material" denotes a region's type as indicated by its labels. Consequently, two regions can be made of the same material, but a region cannot consist of multiple materials. In order to handle complex scenarios, we will utilize the software developped in [68], that uses [82] as backend. This sophisticated software is written in C++ for optimal performance. Each iteration of the software consists of several intermediate steps, including:

- Computing the current energy

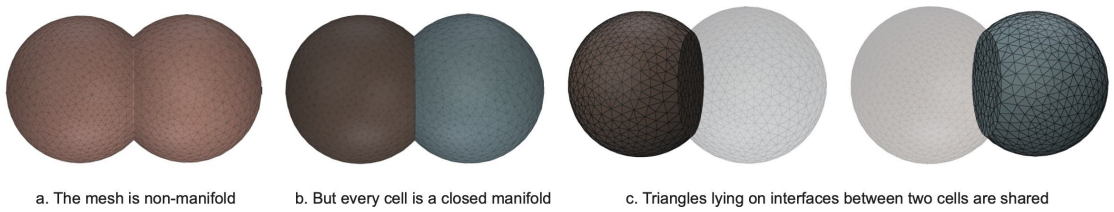


Figure 2.51: Our multimaterial meshes data-structure generalizes elegantly the manifold data structure to several materials.

- Calculating the energy gradient
- Applying volume control corrections to adjust vertex positions and gradients
- Updating vertex positions using a line-search method
- Use multitracker to manage non-manifold mesh operations: collision, detection, remeshing, topology transitions..

The idea behind Multitracker [82] is to start as an input with a correct multimaterial mesh, and vertex displacements. The software returns you a correct mesh with the given displacements executed as best as possible, along with some refinement operations. These operations encompass collision detection, mesh improvement and material merging, separation of different materials, and T1/T2 topology transitions. We will see throughout this thesis that the software is capable of efficiently and effectively managing a wide range of complex situations.

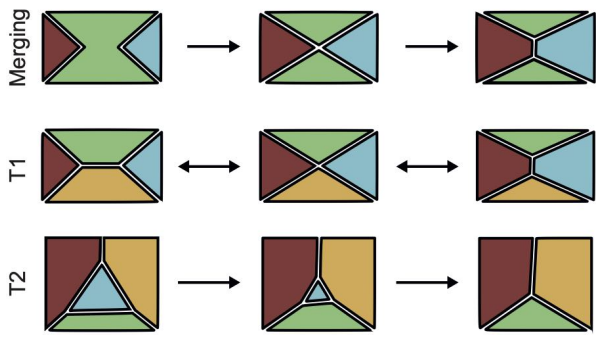


Figure 2.52: Illustrations of merging, T1 and T2 topology changes in 2D. From [82].

2.6.2 Solving topology transitions

In the field of foam physics, the study of T1 and T2 topology transitions plays a crucial role in understanding the behavior and properties of foams. These transitions, initially introduced by [91], provides a taxonomy of the possible rearrangements of foam structures and provides insights on their stability under various conditions. When considering cells or tissues modeled as heterogeneous foams, we encounter the same types of topology changes. However, in foams there is not necessary volume conservation for each closed domain and thus domains can disappear, leading to a coarsening of the foam. In biological systems, at intermediate timescales (slow enough to allow for relaxation, but fast enough to avoid any long-term drifts [92]), except when cells divide, their volumes are kept constant, as they are filled with water, an incompressible fluid. In tissues, surface tensions can vary over time, potentially leading to topological transitions.

T1 transitions occur during a neighbor exchange inside the foam. Cells rearrange their edges and vertices to adopt a new configuration. This process involves the disappearance of an interface separating two cells, followed by the appearance of a new interface connecting a different pair of cells.

T1 Topology transitions are driven by surface tensions changes

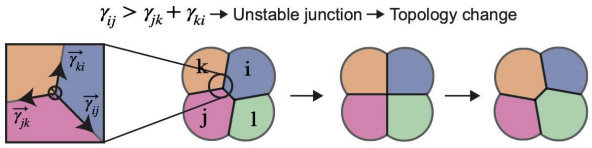


Figure 2.53: The instability of interfaces are due to Young-Dupré relations that makes a junction line shrink. Thus the associated interfaces shrink with them.

For any given junction between three materials i, j, k, we have a Young-Dupré relation that can be written as

$$\vec{\gamma}_{ij} + \vec{\gamma}_{jk} + \vec{\gamma}_{ki} = \vec{0} \tag{2.72}$$

For any surface tension vector $\vec{\gamma}$, we write $||\vec{\gamma}|| = \gamma$. This is only possible if three triangular relationships are verified:

$$\begin{aligned} \gamma_{ij} &< \gamma_{jk} + \gamma_{ki} \\ \gamma_{jk} &< \gamma_{ij} + \gamma_{ki} \\ \gamma_{ki} &< \gamma_{ij} + \gamma_{jk} \end{aligned}$$

Resolution of topology transitions

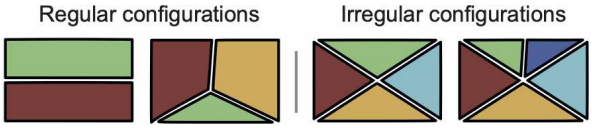


Figure 2.54: Solving topology transitions consists of finding the configurations containing irregular vertex configurations, and restoring it into a regular configurations. This is much easier in 2D than in 3D.

Addressing topology transitions involves identifying configurations that contain irregular vertex arrangements and transforming them into regular configurations (Figure 2.54). This process is considerably simpler in 2D compared to 3D (Figure 2.55). During the simulation, such as the gradient descent, the mesh is examined at each step to detect irregular vertex configurations. When an irregular configuration is discovered, it is resolved to restore a regular arrangement. Following this resolution, the simulation proceeds, ensuring that the overall mesh maintains its desired structure and properties.

T1 topology transitions: encompasses a wide variety of phenotypes

A T1 topology transition happens when at one junction the surface tensions equilibrium could not be respected. In this case, one of the interfaces shrink until new cell-cell contacts are created. However, in some cases the creation of new cell contacts is not possible, or

Figure 2.55: In 3D, even a simple T1 transition requires to solve many complicated intermediate steps. From [82].

not sufficient to restore the equilibrium. In this case, a T1 transition[2] can lead to a richer number of situations: internalization or cell detachment. On the toy-model of the doublet, we demonstrated that we could obtain such situations with certain surface tensions combinations. With our 3D implementation, we are able to reproduce the phenotypes that we found earlier with our model of the doublet [68], and that also correspond to in-vivo phenotypes (Figure 2.56).

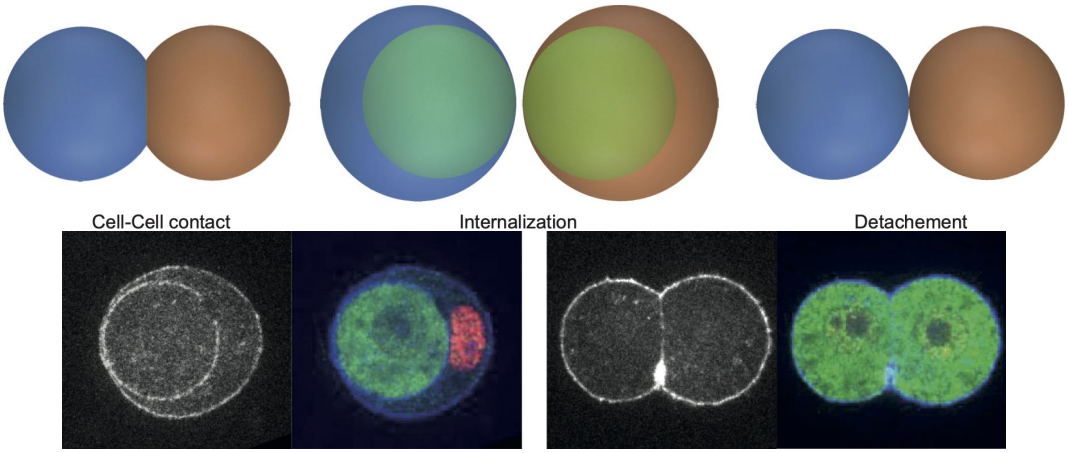


Figure 2.56: Multitracker remeshing, merging and T1 topology changes operations allows to reproduce the configurations of detachement and of cell internalization encoutered in our model the doublet. (Cells are rendered as translucid materials, to observe ingressed cell in the case of internalization). We can observe the phenotype of internalization on blastomeres from mouse early embryos [68]

[2]We argue that this is a special case of a T1 transition. T1 transition is not limited to the classical situation of Figure 2.53

2.6.3 What is a good mesh ?

We said that Multitracker [82] did refinement operations in order to maintain a good mesh-quality. But what is a good mesh ? During the previous optimization, the surface energy and gradient formulas were defined using geometrical quantities that make sense when defined on continuous surfaces: areas A_m, volumes V_l, area derivatives $\frac{\partial A_m}{\partial \vec{x}}$ and volume derivatives $\frac{\partial V_l}{\partial \vec{x}}$. In a triangle mesh, these quantities have to be defined discretely, and often several possible formulae exist for a single geometrical quantity.

A mesh is a discrete approximation of a continuous surface. When the number of vertices n_v and the number of triangles n_t goes to infinity, this approximated surface converges to a smooth surface. A good mesh and a good formula are thus a mesh and a formula with which the computation of the geometrical quantity of interest approximates well the real quantity in the surface that the mesh is trying to approximate. This is a very unpractical definition, as it provides no metric to estimate how good or how bad a mesh is.

The field of discrete differential geometry precisely aims at proposing robust formulas to compute quantities on meshes, as well as general principles to define, create and repair meshes in order to have good meshes. Indeed, there are very general rules that makes a mesh good or bad in general. We have to keep in mind that discrete differential geometry is an experimental science: people use general rules that "are logical", that "make sense", but are not always backed by theorems, and constitutes just "good practises" that one need to follow.:

- Quality of Triangles: The quality of the triangles is critical. Ideally, the triangles should be as equilateral as possible. Extremely elongated or flat triangles can cause numerical issues in calculations.
- Uniformity: A good mesh has relatively uniform triangle sizes across the surface. Uniformity ensures smooth surface representation and helps avoid numerical problems in many calculations, such as finite element analysis or geometric computations.
- Manifoldness: A manifold mesh means that the neighborhood of each vertex resembles a disc. In other words, each edge should be shared by exactly two triangles. Non-manifold meshes, where an edge is shared by more than two triangles or a vertex belongs to non-connected edges, can lead to problems in many geometry processing algorithms.
- No Self-Intersections: Self-intersecting meshes can lead to unexpected results in rendering and physics simulations.
- Smoothness: If representing a smooth surface, the mesh should approximate the surface's curvature as closely as possible. The triangle normals should vary smoothly over the surface.

In the case of the area derivative, cotangents naturally appear (see Appendix A). $\cot(\theta)$ diverges at $\theta = 0$. Thus during the optimization process, if one of the angles of the triangles goes to 0 (or equivalently is the area of one of the triangles goes to 0), we encounter singularities in the area derivatives, and thus numerical instabilities. This is why the gradient-descent diverged at some point.

This mesh-based description is quite practical, as it allows to take advantage of the huge body of work available in the litterature to measure geometrical quantities, repair and refine meshes, and do physical simulations. One of our cardinal ideas has thus been to develop

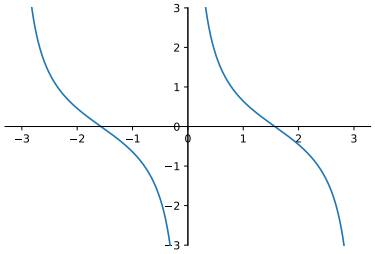


Figure 2.57: The discrete derivative of the area of a triangle cotains a cotanθ in its mathematical expression, diverges if some angles of triangles θ are equal to zero.

robust methods to extract meshes from biological data, to fill the gap between the biological data and the physical models.

In next chapter, we develop Delaunay-Watershed an algorithm to convert instance segmentation masks (easily available since the advent of deep learning) into multimaterial meshes, and we develop an algorithm to perform tension inference.

Bibliography

[1] LadyOfHats. Wikipedia user. *URL: https://commons.wikimedia.org/wiki/User:LadyofHats*.

[2] Pierre-Gilles De Gennes, Françoise Brochard-Wyart, and David Quéré. *Gouttes, bulles, perles et ondes*. Belin Paris, 2002.

[3] Jacob N Israelachvili. Surface forces. In *The Handbook of Surface Imaging and Visualization*, pages 793–816. CRC Press, 2022.

[4] Bruce Alberts. *Molecular biology of the cell*. Garland science, 2017.

[5] AJ Lomakin, CJ Cattin, GPF Nader, PJ Sáez, JM Garcia-Arcos, IY Zhitnyak, et al. The nucleus acts as a ruler tailoring cell responses to spatial constraints. *Science*, 370(6514):eaba2894, 2020.

[6] Valeria Venturini, Fabio Pezzano, Frederic Catala Castro, Hanna-Maria Häkkinen, Senda Jiménez-Delgado, Mariona Colomer-Rosell, Monica Marro, Queralt Tolosa-Ramon, Sonia Paz-López, Miguel A Valverde, et al. The nucleus measures shape changes for cellular proprioception to control dynamic cell behavior. *Science*, 370(6514):eaba2644, 2020.

[7] Paul. J. Flory and M. Volkenstein. Statistical mechanics of chain molecules. *Biopolymers*, 8(5):699–700, 1969.

[8] Pierre-Gilles De Gennes. *Scaling concepts in polymer physics*. Cornell university press, 1979.

[9] Terence R Strick, MN Dessinges, G Charvin, NH Dekker, JF Allemand, D Bensimon, and V Croquette. Stretching of macromolecules and proteins. *Reports on Progress in Physics*, 66(1):1, 2002.

[10] David Quéré. *Surfaces molles: majeure de mécanique, option dynamique des fluides: promotion 1999, majeure 1, 2e année, PHY562*. École polytechnique, 2001.

[11] Elaine Fuchs and Don W Cleveland. A structural scaffolding of intermediate filaments in health and disease. *Science*, 279(5350):514–519, 1998.

[12] MC Ledbetter and KR Porter. A" microtubule" in plant cell fine structure. *The Journal of cell biology*, 19(1):239–250, 1963.

[13] Lindsay M Biga, Sierra Dawson, Amy Harwell, Robin Hopkins, Joel Kaufmann, Mike LeMaster, Philip Matern, Katie Morrison-Graham, Devon Quick, and Jon Runyeon. *Anatomy & physiology*. OpenStax/Oregon State University, 2020.

[14] Kenneth C Holmes, David Popp, Werner Gebhard, and Wolfgang Kabsch. Atomic model of the actin filament. *Nature*, 347(6288):44–49, 1990.

[15] Thomas D Pollard and Christopher C Beltzner. Structure and function of the arp2/3 complex. *Current opinion in structural biology*, 12(6):768–774, 2002.

[16] Jacques Prost, Frank Jülicher, and Jean-François Joanny. Active gel physics. *Nature physics*, 11(2):111–117, 2015.

[17] Guillaume T Charras, Justin C Yarrow, Mike A Horton, L Mahadevan, and TJ Mitchison. Non-equilibration of hydrostatic pressure in blebbing cells. *Nature*, 435(7040):365–369, 2005.

[18] Guillaume T Charras, Margaret Coughlin, Timothy J Mitchison, and L Mahadevan. Life and times of a cellular bleb. *Biophysical journal*, 94(5):1836–1853, 2008.

[19] Guillaume Charras and Ewa Paluch. Blebs lead the way: how to migrate without lamellipodia. *Nature reviews Molecular cell biology*, 9(9):730–736, 2008.

[20] Tetsuro Hirose, Kensuke Ninomiya, Shinichi Nakagawa, and Tomohiro Yamazaki. A guide to membraneless organelles and their various roles in gene regulation. *Nature Reviews Molecular Cell Biology*, 24(4):288–304, 2023.

[21] Victoria Tianjing Yan, Arjun Narayanan, Tina Wiegand, Frank Jülicher, and Stephan W Grill. A condensate dynamic instability orchestrates actomyosin cortex activation. *Nature*, 609(7927):597–604, 2022.

[22] Karsten Kruse, Jean-Francois Joanny, Frank Jülicher, Jacques Prost, and Ken Sekimoto. Generic theory of active polar gels: a paradigm for cytoskeletal dynamics. *The European Physical Journal E*, 16:5–16, 2005.

[23] M Cristina Marchetti, Jean-François Joanny, Sriram Ramaswamy, Tanniemola B Liverpool, Jacques Prost, Madan Rao, and R Aditi Simha. Hydrodynamics of soft active matter. *Reviews of modern physics*, 85(3):1143, 2013.

[24] UraniumMonkey. *URL: https://commons.wikimedia.org/wiki/User:UraniumMonkey*.

[25] Hervé Turlier, Basile Audoly, Jacques Prost, and Jean-François Joanny. Furrow constriction in animal cell cytokinesis. *Biophysical journal*, 106(1):114–123, 2014.

[26] Hudson Borja da Rocha, Jeremy Bleyer, and Hervé Turlier. A viscous active shell theory of the cell cortex. *Journal of the Mechanics and Physics of Solids*, 164:104876, 2022.

[27] J Arthur Thomson. On growth and form, 1917.

[28] Sascha Hilgenfeldt, Sinem Erisken, and Richard W Carthew. Physical modeling of cell geometric order in an epithelial tissue. *Proceedings of the National Academy of Sciences*, 105(3):907–911, 2008.

[29] Frans Van Roy and Geert Berx. The cell-cell adhesion molecule e-cadherin. *Cellular and molecular life sciences*, 65:3756–3788, 2008.

[30] Jean-Léon Maître, Hélène Berthoumieux, Simon Frederik Gabriel Krens, Guillaume Salbreux, Frank Jülicher, Ewa Paluch, and Carl-Philipp Heisenberg. Adhesion functions in cell sorting by mechanically coupling the cortices of adhering cells. *science*, 338(6104):253–256, 2012.

[31] kds4444. *URL: https://commons.wikimedia.org/wiki/User:KDS4444*.

[32] Jasmin Imran Alsous, Nicolas Romeo, Jonathan A Jackson, Frank M Mason, Jörn Dunkel, and Adam C Martin. Dynamics of hydraulic and contractile wave-mediated fluid transport during drosophila oogenesis. *Proceedings of the National Academy of Sciences*, 118(10):e2019749118, 2021.

[33] Anniina Salonen, Cyprien Gay, Armando Maestro, Wiebke Drenckhan, and Emmanuelle Rio. Arresting bubble coarsening: A two-bubble experiment to investigate grain growth in the presence of surface elasticity. *Europhysics Letters*, 116(4):46005, 2017.

[34] Jean-Léon Maître, Ritsuya Niwayama, Hervé Turlier, François Nédélec, and Takashi Hiiragi. Pulsatile cell-autonomous contractility drives compaction in the mouse embryo. *Nature cell biology*, 17(7):849–855, 2015.

[35] E Evans and A Yeung. Apparent viscosity and cortical tension of blood granulocytes determined by micropipet aspiration. *Biophysical journal*, 56(1):151–160, 1989.

[36] Franz J Giessibl. Advances in atomic force microscopy. *Reviews of modern physics*, 75(3):949, 2003.

[37] Daniel Rugar and Paul Hansma. Atomic force microscopy. *Physics today*, 43(10):23–30, 1990.

[38] Peter Eaton and Paul West. *Atomic force microscopy*. Oxford university press, 2010.

[39] Elisabeth Fischer-Friedrich, Yusuke Toyoda, Cedric J Cattin, Daniel J Müller, Anthony A Hyman, and Frank Jülicher. Rheology of the active cell cortex in mitosis. *Biophysical journal*, 111(3):589–600, 2016.

[40] Elisabeth Fischer-Friedrich, Anthony A Hyman, Frank Jülicher, Daniel J Müller, and Jonne Helenius. Quantification of surface tension and internal pressure generated by single mitotic cells. *Scientific reports*, 4(1):6213, 2014.

[41] Cyrille Norotte, Francoise Marga, Adrian Neagu, Ioan Kosztin, and Gabor Forgacs. Experimental evaluation of apparent tissue surface tension based on the exact solution of the laplace equation. *EPL (Europhysics Letters)*, 81(4):46003, 2008.

[42] Dennis E Discher, Paul Janmey, and Yu-li Wang. Tissue cells feel and respond to the stiffness of their substrate. *Science*, 310(5751):1139–1143, 2005.

[43] Adam J Engler, Shamik Sen, H Lee Sweeney, and Dennis E Discher. Matrix elasticity directs stem cell lineage specification. *Cell*, 126(4):677–689, 2006.

[44] Kshitiz, JinSeok Park, Peter Kim, Wilda Helen, Adam J Engler, Andre Levchenko, and Deok-Ho Kim. Control of stem cell fate and function by engineering physical microenvironments. *Integrative biology*, 4(9):1008–1018, 2012.

[45] Otger Campàs, Tadanori Mammoto, Sean Hasso, Ralph A Sperling, Daniel O'connell, Ashley G Bischof, Richard Maas, David A Weitz, Lakshminarayanan Mahadevan, and Donald E Ingber. Quantifying cell-generated mechanical forces within living embryonic tissues. *Nature methods*, 11(2):183–189, 2014.

[46] Friedhelm Serwane, Alessandro Mongera, Payam Rowghanian, David A Kealhofer, Adam A Lucio, Zachary M Hockenbery, and Otger Campas. In vivo quantification of spatially varying mechanical properties in developing tissues. *Nature methods*, 14(2):181–186, 2017.

[47] Robert P Behringer and Bulbul Chakraborty. The physics of jamming for granular materials: a review. *Reports on Progress in Physics*, 82(1):012601, 2018.

[48] OJ O'loan. Jamming transition in a homogeneous one-dimensional system: The bus route model. *Physical Review E*, 58(2):1404, 1998.

[49] ME Cates, JP Wittmer, and J-P Bouchaud. Jamming, force chains, and fragile matter. *Physical review letters*, 81(9):1841, 1998.

[50] Andrea J Liu and Sidney R Nagel. Jamming is not just cool any more. *Nature*, 396(6706):21–22, 1998.

[51] Edouard Hannezo and Carl-Philipp Heisenberg. Rigidity transitions in development and disease. *Trends in Cell Biology*, 2022.

[52] Alessandro Mongera, Payam Rowghanian, Hannah J Gustafson, Elijah Shelton, David A Kealhofer, Emmet K Carn, Friedhelm Serwane, Adam A Lucio, James Giammona, and Otger Campàs. A fluid-to-solid jamming transition underlies vertebrate body axis elongation. *Nature*, 561(7723):401–405, 2018.

[53] Dapeng Bi, JH Lopez, and M Lisa Manning. A density-independent rigidity transition in biological tissues. *Nature Physics*, 11(12):1074–1079, 2015.

[54] Dapeng Bi, Xingbo Yang, M Cristina Marchetti, and M Lisa Manning. Motility-driven glass and jamming transitions in biological tissues. *Physical Review X*, 6(2):021011, 2016.

[55] Jin-Ah Park, Jae Hun Kim, Dapeng Bi, Jennifer A Mitchel, Nader Taheri Qazvini, Kelan Tantisira, Chan Young Park, Maureen McGill, Sae-Hoon Kim, Bomi Gweon, et al. Unjamming and cell shape in the asthmatic airway epithelium. *Nature materials*, 14(10):1040–1048, 2015.

[56] Nicoletta I Petridou, Bernat Corominas-Murtra, Carl-Philipp Heisenberg, and Edouard Hannezo. Rigidity percolation uncovers a structural basis for embryonic tissue phase transitions. *Cell*, 184(7):1914–1928, 2021.

[57] Ojan Khatib Damavandi, Varda F Hagh, Christian D Santangelo, and M Lisa Manning. Energetic rigidity. i. a unifying theory of mechanical stability. *Physical Review E*, 105(2):025003, 2022.

[58] Ojan Khatib Damavandi, Varda F Hagh, Christian D Santangelo, and M Lisa Manning. Energetic rigidity. ii. applications in examples of biological and underconstrained materials. *Physical Review E*, 105(2):025004, 2022.

[59] Philippe Marmottant, Abbas Mgharbel, Jos Käfer, Benjamin Audren, Jean-Paul Rieu, Jean-Claude Vial, Boudewijn Van Der Sanden, Athanasius FM Marée, François Graner, and Hélène Delanoë-Ayari. The role of fluctuations and stress on the effective viscosity of cell aggregates. *Proceedings of the National Academy of Sciences*, 106(41):17271–17275, 2009.

[60] Diana Pinheiro, Roland Kardos, Édouard Hannezo, and Carl-Philipp Heisenberg. Morphogen gradient orchestrates pattern-preserving tissue morphogenesis via motility-driven unjamming. *Nature Physics*, 18(12):1482–1493, 2022.

[61] Robert I. Saye and James A. Sethian. Multiscale modeling of membrane rearrangement, drainage, and rupture in evolving foams. *Science*, 340(6133):720–724, 2013.

[62] Alberto Stolfi, Elijah K Lowe, Claudia Racioppi, Filomena Ristoratore, C Titus Brown, Billie J Swalla, and Lionel Christiaen. Divergent mechanisms regulate conserved cardiopharyngeal development and gene expression in distantly related ascidians. *eLife*, 3:e03728, sep 2014.

[63] Reza Farhadifar, Jens-Christian Röper, Benoit Aigouy, Suzanne Eaton, and Frank Jülicher. The influence of cell mechanics, cell-cell interactions, and proliferation on epithelial packing. *Current biology*, 17(24):2095–2104, 2007.

[64] G Odell, G Oster, B Burnside, and P Alberch. A mechanical model for epithelial morphogenesis. *Journal of mathematical biology*, 9:291–295, 1980.

[65] Tatsuzo Nagai and Hisao Honda. A dynamic cell model for the formation of epithelial tissues. *Philosophical Magazine B*, 81(7):699–719, 2001.

[66] G Wayne Brodland. Computational modeling of cell sorting, tissue engulfment, and related phenomena: A review. *Appl. Mech. Rev.*, 57(1):47–76, 2004.

[67] Sangwoo Kim, Marie Pochitaloff, Georgina A Stooke-Vaughan, and Otger Campàs. Embryonic tissues as active foams. *Nature physics*, 17(7):859–866, 2021.

[68] Jean-Léon Maître, Hervé Turlier, Rukshala Illukkumbura, Björn Eismann, Ritsuya Niwayama, François Nédélec, and Takashi Hiiragi. Asymmetric division of contractile domains couples cell positioning and fate specification. *Nature*, 536(7616):344–348, 2016.

[69] J. Steiner. Einfache beweise der isoperimetrischen hauptsätze. *Journal für die reine und angewandte Mathematik*, 18:281–296, 1838.

[70] H. A. Schwarz. Beweis des satzes, dass die kugel kleinere oberfläche besitzt, als jeder andere körper gleichen volumens. *Nachrichten von der Königl. Gesellschaft der Wissenschaften und der Georg-Augusts-Universität zu Göttingen*, 1884:1–13, 1884.

[71] John M Sullivan and Frank Morgan. Open problems in soap bubble geometry. *International Journal of Mathematics*, 7(06):833–842, 1996.

[72] Emanuel Milman and Joe Neeman. The gaussian double-bubble and multi-bubble conjectures. *Annals of Mathematics*, 195(1):89–206, 2022.

[73] Keenan Crane. Discrete differential geometry: An applied introduction. *Notices of the AMS, Communication*, 1153, 2018.

[74] Leiv Magne Siqveland and Svein Magne Skjæveland. Derivations of the young-laplace equation. *Capillarity*, 4(2):23–30, 2021.

[75] Lev Davidovich Landau and Evgenii Mikhailovich Lifshitz. *Fluid Mechanics: Landau and Lifshitz: Course of Theoretical Physics, Volume 6*, volume 6. Elsevier, 2013.

[76] Krishna Vedala. *URL: https://commons.wikimedia.org/wiki/User:Krishnavedala.*

[77] John M Sullivan. Pleasing shapes for topological objects. In *Mathematics and Modern Art: Proceedings of the First ESMA Conference, held in Paris, July 19-22, 2010*, pages 153–165. Springer, 2012.

[78] Gregal A.L. *URL: https://commons.wikimedia.org/wiki/User:Greg_A_L.*

[79] James A Sethian et al. *Level set methods and fast marching methods*, volume 98. Cambridge Cambridge UP, 1999.

[80] Oleg Alexandrov. *URL: https://commons.wikimedia.org/wiki/User:Oleg_Alexandrov.*

[81] Kenneth A Brakke. The surface evolver. *Experimental mathematics*, 1(2):141–165, 1992.

[82] Fang Da, Christopher Batty, and Eitan Grinspun. Multimaterial mesh-based surface tracking. *ACM Trans. Graph.*, 33(4):112–1, 2014.

[83] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[84] Robert E Wengert. A simple automatic derivative evaluation program. *Communications of the ACM*, 7(8):463–464, 1964.

[85] Seppo Linnainmaa. The representation of the cumulative rounding error of an algorithm as a taylor expansion of the local rounding errors. Master's thesis, University of Helsinki, 1970. Master's Thesis (in Finnish).

[86] Andreas Griewank and Andrea Walther. *Evaluating derivatives: principles and techniques of algorithmic differentiation*. SIAM, 2 edition, 2008.

[87] Atilim Gunes Baydin, Barak A Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: a survey. *Journal of Machine Learning Research*, 18(153):1–43, 2018.

[88] DP Kingma. Adam: a method for stochastic optimization. In *Int Conf Learn Represent*, 2014.

[89] Baptiste Nicolet, Alec Jacobson, and Wenzel Jakob. Large steps in inverse rendering of geometry. *ACM Transactions on Graphics (TOG)*, 40(6):1–13, 2021.

[90] Selena Zihan Ling, Nicholas Sharp, and Alec Jacobson. Vectoradam for rotation equivariant geometry optimization. *Advances in Neural Information Processing Systems*, 35:4111–4122, 2022.

[91] Denis L Weaire and Stefan Hutzler. *The physics of foams*. Oxford University Press, 2001.

[92] Laure Saias, Jim Swoger, Arturo D'Angelo, Peran Hayes, Julien Colombelli, James Sharpe, Guillaume Salbreux, and Jérôme Solon. Decrease in cell volume generates contractile forces driving dorsal closure. *Developmental cell*, 33(5):611–621, 2015.

Chapter 3

Embryo mechanics cartography: inference of 3D force atlases from fluorescence microscopy

Written in collaboration with Fabrice Delbary, Alex McDougall, Rémi Dumollard and Hervé Turlier.

This work resulted in a publication, currently under revision and available on bioRχiv.

This project was the main focus of my Ph.D. thesis. Once I understood that multimaterial meshes were the right data structure to study heterogeneous foams numerically, I developed Delaunay-Watershed, a method to extract such meshes from segmented data. From there, the next steps were quite obvious: One can easily extract geometrical quantities from such meshes, such as angles and curvatures, and use them to invert equilibrium equations to infer forces. Then, I validated this procedure by generating images from simulations before applying this procedure to real data. It was quite a surprise to see that this procedure scaled so well, and that we were able to find meaningful tension patterns of surface tension on Phallusia Mammillata embryos of up to 800 cells. An important lesson of this work is that open-access data allowed us to greatly accelerate the application of our work. To conclude, it is interesting to notice that this project was not the hardest one of my PhD, but it was well executed and solves an important problem, and will thus probably be the one that will have the biggest impact. I picked a low-hanging fruit.

The morphogenesis of tissues and embryos results from a tight interplay between gene expression, biochemical signaling and mechanics. Although sequencing methods allow the generation of cell-resolved spatio-temporal maps of gene expression in developing tissues, creating similar maps of cell mechanics in 3D has remained a real challenge. Exploiting the foam-like geometry of cells in embryos, we propose a robust end-to-end computational method to infer spatiotemporal atlases of cellular forces from fluorescence microscopy images of cell membranes. Our method generates precise 3D meshes of cell geometry and successively predicts relative cell surface tensions and pressures in the tissue. We validate it with 3D foam simulations, study its noise sensitivity, and prove its biological relevance in mouse, ascidian and *C. elegans* embryos. 3D inference allows us to recover mechanical features identified previously, but also predicts new ones, unveiling potential new insights on the spatiotemporal regulation of cell mechanics

in early embryos. Our code is freely available and paves the way for unraveling the unknown mechanochemical feedbacks that control embryo and tissue morphogenesis.

Chapitre 3

Cartographie de la mécanique de l'embryon : inférence d'atlas de forces 3D à partir de la microscopie à fluorescence

Écrit en collaboration avec Fabrice Delbary, Alex McDougall, Rémi Dumollard et Hervé Turlier.

Ce travail a abouti à une publication, actuellement en révision et disponible sur bioRχiv.

Ce projet était le principal projet de ma thèse de doctorat. Après avoir compris que les maillages multimateriaux étaient la structure de données appropriée pour étudier numériquement les mousses hétérogènes, j'ai développé Delaunay-Watershed, une méthode pour extraire de tels maillages à partir de données volumétriques segmentées. Les étapes suivantes étaient évidentes : Il est facile d'extraire des quantités géométriques de ces maillages, telles que les angles et les courbures, et les utiliser pour inverser les équations d'équilibre afin de déduire les forces. J'ai ensuite validé cette procédure en générant des images à partir de simulations avant d'appliquer cette procédure à de vraies données. Ce fut une agréable surprise de constater que cette procédure se généralisait a des plus gros amas de cellules, et que nous étions capables de trouver des motifs de tension significatifs de tension de surface sur des embryons de Phallusia Mammillata contenant jusqu'à 800 cellules. Une leçon importante de ce travail est que les données en accès libre nous ont permis d'accélérer grandement l'application de notre travail. Pour conclure, il est intéressant de noter que ce projet n'était pas le plus difficile de mon doctorat, mais il a été bien exécuté et résoud un problème important, et sera donc probablement celui qui aura le plus grand impact.

La morphogenèse des tissus et des embryons résulte d'une étroite interaction entre l'expression génétique, la signalisation biochimique et la mécanique. Bien que les méthodes de séquençage permettent la génération de cartes spatio-temporelles à l'échelle de la cellule de l'expression génique dans les tissus en développement, créer des cartes similaires de la mécanique cellulaire en 3D reste un véritable défi. Exploitant la géométrie semblable à de la mousse des cellules dans les embryons, nous proposons une méthode informatique robuste de bout en bout pour inférer des atlas spatiotemporels de forces cellulaires à partir d'images de microscopie à fluorescence des membranes cellulaires. Notre méthode génère des maillages 3D précis de la géométrie cellulaire et prédit successivement les tensions relatives de la surface cellulaire et les pressions dans le tissu. Nous la validons avec des simulations de mousse en 3D, étudions sa sensibilité au bruit, et prouvons sa pertinence biologique dans les embryons de souris, d'ascidien et de *C. elegans*. L'inférence en 3D nous permet de récupérer des caractéristiques mécaniques identifiées précédemment, mais en prédit également de nouvelles, révélant de possibles mécanismes de régulation spatiotemporelle de la mécanique cellulaire dans les premiers embryons. Notre code est librement disponible et ouvre la voie à l'élucidation des rétroactions mécano-chimiques inconnues qui contrôlent la morphogenèse des embryons et des tissus.

3.1 Introduction

Understanding the mechanical regulation of embryo and tissue shape emergence is a long-standing goal in developmental biology and biological physics. Although gene expression patterning in early embryos is increasingly documented thanks to recent single cell sequencing methods [1, 2], we still know very little about how cellular forces are spatio-temporally patterned within embryos and tissues. This is due to the lack of efficient methods for extracting cell- and time-resolved mechanics in a systematic, tissue-wide, and noninvasive manner.

Most experimental methods to measure mechanics are local and time-consuming, such as micropipette aspiration, AFM measurement, or embedded droplet deformation [3–15], making the generation of spatio-temporal maps of mechanics tedious; others are invasive, such as laser ablation, perturbing normal tissue development [16, 17]; or they probe mechanics only at the tissue level [18–21], ignoring mechanical heterogeneities within the multicellular structure. Interestingly, all methods require live 3D imaging to follow the deformation of cells, tissues, or embedded objects. Advances in fluorescence microscopy allow us to record the geometry of cells during the development of an embryo *in toto* from the zygote to a few hundreds of cells with a confocal microscope [22] and up to thousands of cells with a light sheet microscope [23, 24]. Attractive new microscopy techniques have emerged to try to quantify cellular mechanics directly, such as Brillouin microscopy [25, 26], or membrane tension probes [27–29], but such methods still lack cross-validations and remain difficult to link directly to mechanical models of tissues.

An alternative idea that emerged a decade ago is to infer the forces that dictate the shape of cells directly from their geometry by solving an inverse mechanical model [30]. These mechanical inference methods (also called *force* or *stress inference*) are based only on image analysis and do not require tissue perturbation: they have therefore a lower entry barrier than many other methods, as they do not require complex experimental setups. They have been shown to be efficient in inferring tensions (and pressure) in 2D cell monolayers [31] and can be scaled to hundreds or thousands of cells. For tissues and embryos, most inference methods are based on the hypothesis that cells adopt shapes and arrangements similar to bubbles in a foam, as pointed out by D'Arcy Thompson more than a century ago [**?**]. This analogy implies that the mechanics of cells is dominated by tensile stresses on their surface, which are generated by actomyosin contractility [32]. Because actomyosin contractility may be regulated differentially in distinct cells or at different interfaces (such as the cell-medium and cell-cell interfaces [5]), embryos and tissues may be seen as heterogeneous foams, where each cellular interface may adopt a different tension. Actual inference methods also assume generally a quasistatic mechanical equilibrium, where the viscous relaxation of tensions (dozens of seconds) is much faster than typical developmental timescales (dozens of minutes to hours). This foam-like mechanical equilibrium underpins two force balances, the Young-Dupré and Young-Laplace equations (Section 3.2), relating surface tensions with contact angles and cell pressures with interface curvatures. In the next, we will therefore refer to tension and pressure inference.

First versions of tension inference methods [33, 34] neglected Laplace's law by assuming straight cell interfaces, as in traditional vertex models [35, 36]. In addition, they treated tensions and pressure as independent variables, which made the inverse problem generally underdetermined and relatively sensitive to noise. Alternatively, segmentation of cell membranes into 2D polygonal lines to explicitly measure their curvature [37, 38] allows successive determinations of tension and pressure and makes the set of equations generally overdetermined. In the particular case where the whole tissue can be imaged with its boundaries -

as this is generally the case for early embryos - the problem turns out to be systematically overdetermined. However the generalization of this approach to three dimensions has not been convincing so far, since high-quality images and a robust segmentation pipeline are required [39,40]. To avoid such issue, an elegant variational 2D approach was recently proposed in which cell junctions are fitted by circular arcs to find tensions and pressure [41], taking advantage of a mapping between a heterogeneous 2D foam and the tiling of the space into "circular arc polygons". This tiling falls actually within the class of Möbius diagrams [42,43], whose mapping to 2D foams was already pointed out mathematically [44]. In 3D however, interfaces have mean constant curvatures but are generally not portions of sphere and may adopt saddle-node shapes, as remarked for homogeneous foams already [45]. In contrary to a recent assumption [46], the generalization to 3D of the variational scheme developed by [41] with Möbius diagrams is mathematically not correct.

To fill the gap, we propose a robust end-to-end computational method for performing force inference in three dimensions, starting directly from 3D fluorescence microscopy of cell membranes. Our pipeline follows the 2D approach of [37], where we decouple tension and pressure inference. It relies particularly on a novel and efficient surface mesh reconstruction method to precisely quantify cell geometry and our inversion algorithm includes junction length and interface areas as weights to infer tensions and pressures more robustly. Importantly, we perform a comprehensive benchmarking of our pipeline using 3D foam-like simulations and a systematic sensitivity analysis on various tension and pressure inversion formulas. Our inference pipeline yields convincing results on early embryos of mice, *C. elegans* and ascidians by recovering known mechanical characteristics and predicting new ones. We provide an easy-to-install Python software and a comprehensive set of user-friendly 3D visualization tools.

3.2 Results

Delaunay-watershed algorithm for multimaterial mesh generation

An essential first step is to extract the precise geometry of cells from microscopy images. Voxel-based segmentation masks are heavy data structures that are not well adapted to measure geometrical features such as contact angles or mean curvatures. Alternatively, triangle mesh representations of cell interfaces possess several advantages: they are sparse data structures that facilitate the retrieval of geometric quantities using a discrete differential formula [48,49]. They are easy to render graphically and form basic elements for computational modeling, such as vertex models [50,51] or finite element methods [52]. The surface meshes of interest in our case are triangular, nonmanifold to account for tri-cellular junctions, and multimaterial to keep track of the identity of each enclosed cell or region ("material"), in the spirit of [53]. Although triangle meshes can be generated by discretizing voxel-based segmentation masks directly, using marching cube algorithms [54] or more recent methods [55], we found that previous algorithms introduced large errors in angle measurements in general.

Therefore, we developed a novel algorithm that robustly generates nonmanifold multimaterial surface meshes from cell segmentation masks[1]. The first step consists of computing a Euclidean distance transform map (EDT) [57] from the cell segmentation mask [2], which represents a smooth topographic map of cell (and image) boundaries (Fig. 3.2a). From the

[1] in this paper we used either the deep-learning tool *cellpose* [56] or preexisting segmentation masks

[2] This EDT map may also be predicted directly from raw fluorescent images by training a convolutional-neural network [58,59]

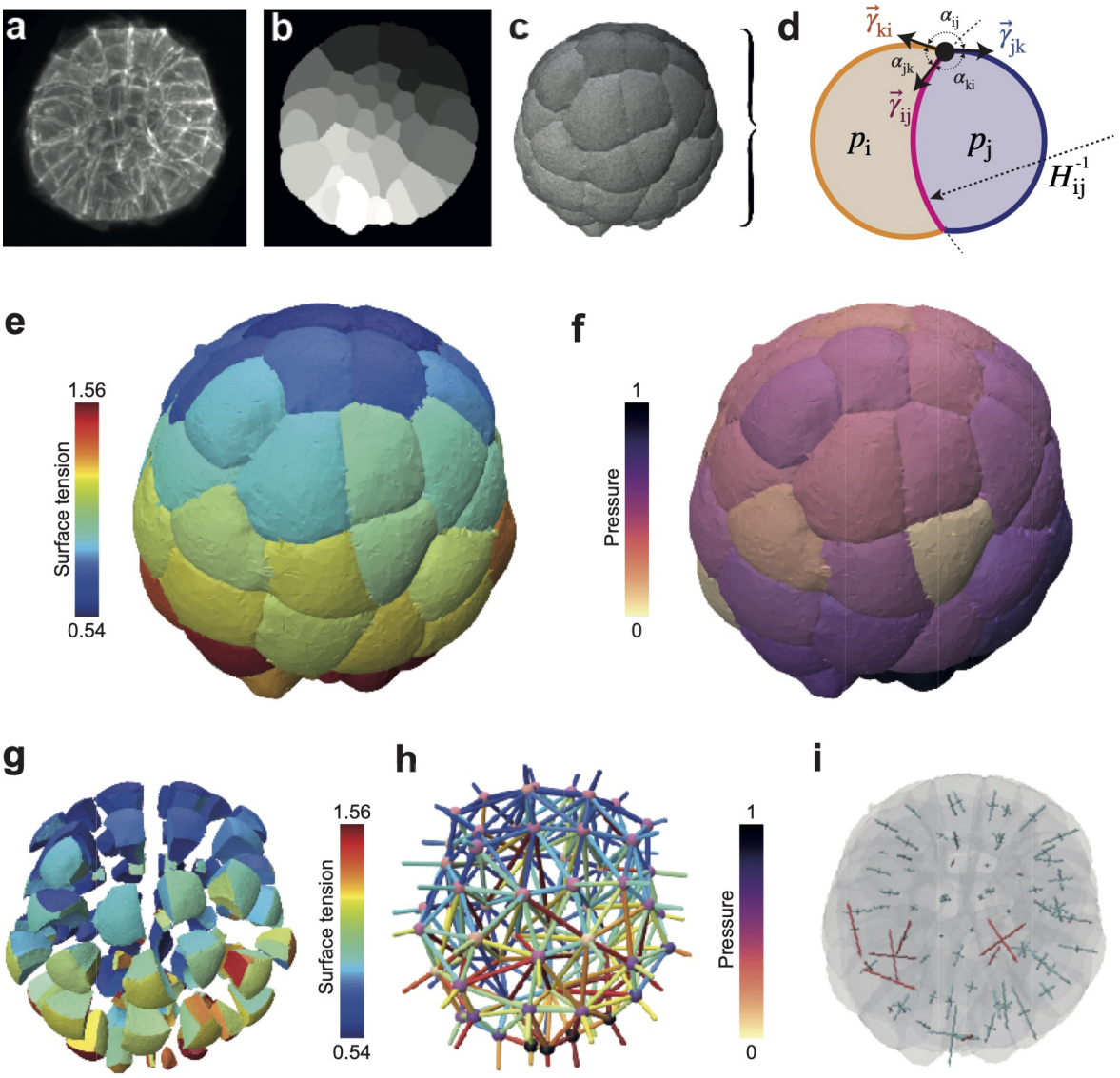


Figure 3.1: **3D force inference procedure and resulting mechanical atlas for a 64 cell ascidian embryo. a)** 3D fluorescence microscopy image (max projection) of a 64-cell *Phallusia mammillata* embryo (from [23]). **b)** Cell segmentation mask in one focal plane of the 3D image. **c)** Multicellular surface mesh of cell interfaces. **d)** Schematic cell doublet illustrating the two force balances that need to be inverted: the Young-Dupré equation that relates surface tensions γ_{ij}, γ_{ik} and γ_{jk} with contact angles α_{ij}, α_{ik} and α_{jk}, and the Young-Laplace equation that relates cell pressure difference $P_j - P_i$ with tension γ_{ij} and the radius of the interface curvature H_{ij}^{-1}. **e)** 3D map of relative surface tensions in the embryo, plotted with a color code from blue (lowest) to red (highest). **f)** Pressure map in the embryo, normalized from 0 to 1. **g)** Exploded view of the surface tension map that illustrates cell-cell contact tensions within the embryo. **h)** Force graph representation of the mechanical atlas, where each node represents a cell with its associated pressure and each edge corresponds to an interface colored by its tension value. **i)** 3D stress eigenvalue representation, corresponding to a stress tensor calculated per cell with the Batchelor formula [47]. Positive eigenvalues are plotted in blue (compressive stress) while negative are plotted in red (extensile stress).

distance map, we sample points at the extrema of the elevation value using a max-pooling operator, which serves as control points to generate a Delaunay tessellation of the space (triangulation in 2D or tetrahedralization in 3D). A dual Voronoi diagram is then generated from

the Delanaunay tessellation and is represented as an edge-weighted graph $\mathcal{G} = (\mathcal{N}, \mathcal{E}, \mathcal{W})$, where $\mathcal{N}$ is the set of nodes, representing tetrahedra in the dual space (triangles in 2D), $\mathcal{E}$ the set of edges between these nodes and $\mathcal{W}$ their associated weights. These weights are defined here according to the average value of the integrated distance map measured along the corresponding triangle (or edge in 2D) in the dual space (Extended data Fig. 2a). Seeding each region using masks, we partition this graph using a watershed algorithm [60] that separates the nodes in the graph between the different cells and the external region[3]. Mapped back on the dual Delaunay space, this partition defines a unique surface (contour in 2D) mesh that accurately follows cell boundaries.

Our *Delaunay-watershed* mesh generation algorithm works just as well in 2D as in 3D (Fig. 3.2a). Since the main purpose of this mesh generation algorithm is to extract precise geometrical features, we generated a set of 47 foam-like simulations of embryos with a number of cells varying from 2 to 11, which we translated into artificial confocal fluorescent images of size $\sim [250 \times 250 \times 250]$ [64] to compare the error generated for different geometrical measures of interest (contact angles, mean curvature, junction length, area, and volume) by our pipeline and state-of-the-art surface meshing techniques implemented in CGAL [55]. Our Delaunay-watershed algorithm [65] outperforms CGAL [55] for the retrieval of contact angles (Fig. 3.2b), and cell volumes or junction lengths (Extended data Fig. 2b), while its precision is comparable for the retrieval of interface areas and mean curvatures (Extended data Fig. 2b).

Tension and pressure balance

Once the geometry of the cells can be calculated from the cell segmentation mesh (Extended data Fig.1 and Supplementary Note), we have to formulate the inverse mechanical problem to retrieve the relative force of the cells from their geometry. A quasi-static foam-like equilibrium underpins two stress balance equations within the tissue (Fig. 3.1d). The Young-Laplace equation

$$p_i - p_j = \gamma_{ij} H_{ij} \tag{3.1}$$

relates the hydrostatic pressure difference $p_i - p_j$ between cells of indices[4] i and j with the interface tension $\gamma_{ij} = \|\vec{\gamma}_{ij}\|$ and the interface mean curvature H_{ij}, which is homogeneous along each interface. The Young-Dupré force balance

$$\vec{\gamma}_{ij} + \vec{\gamma}_{jk} + \vec{\gamma}_{ki} = \vec{0} \tag{3.2}$$

states that the sum of vectorial tensions should be zero at each tri-cellular junction line that joins the interfaces between cells i, j and k. This vectorial sum is equivalent to saying that tensions are coplanar and form a triangle, which implies the triangle inequality $\gamma_{ij} < \gamma_{jk} + \gamma_{ki}$ and equivalent relations by permutation of the indices i, j and k. Non-compliance with one of these inequalities indicates that tension balance breaks down and predicts generically a topological transition in the embryo or tissue. The Young-Dupré tension balance can generically be decomposed into a set of two independent scalar equations that combine the polar angles between the interfaces α_{ij}, α_{jk} and α_{ki} (Fig. 3.1d). In the following, we use five different variants of tension balance that involve cosines and sines of polar angles only [30], which we named *Young-Dupré*, *Young-Dupré projection*, *Lami*, *inverse Lami* and *Lami logarithm* [66] (see Methods 3.6 and Supplementary Note)

[3] Other graph partitioning methods such as multicut [61], hierarchical agglomeration [62] or Mutex watershed [63] algorithms may also be envisioned, although we have not tried them directly.

[4] by convention the index 0 will refer to the external medium

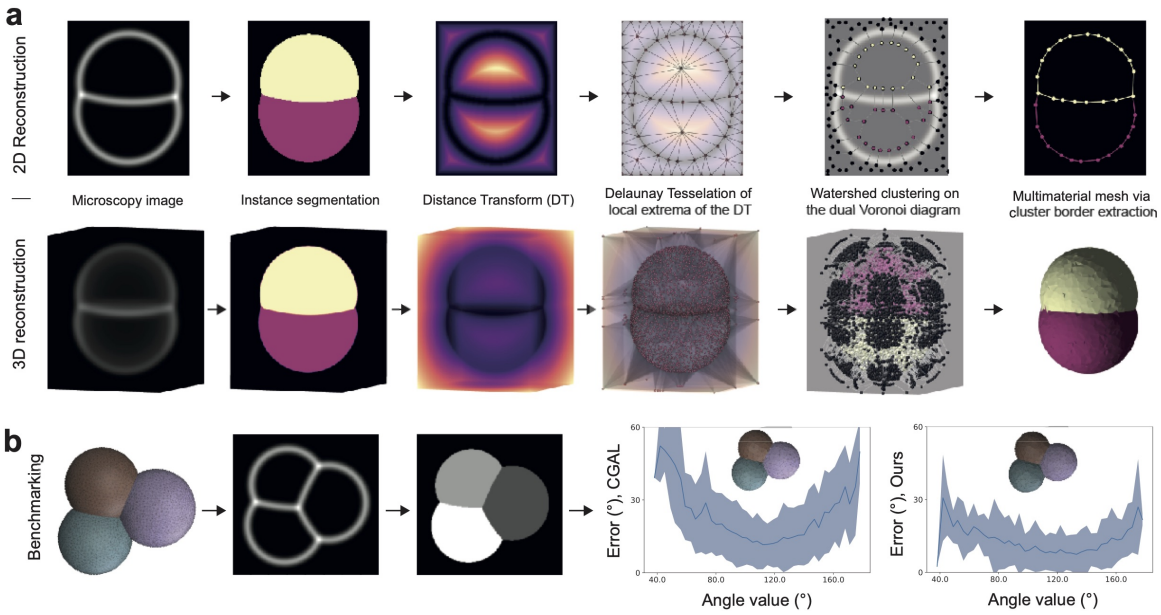


Figure 3.2: **Multimaterial mesh generation algorithm. a)** From a microscopy image (artificial here) in 2D (resp. 3D), we first generate a distance transform map, including the image boundaries; we then sample points at the extremum values of this map to generate a Delaunay triangulation (resp. tetrahedralization) of the 2D (resp. 3D) space; the average integrated elevation value along edges (resp. triangles) of this tessellation gives weight to edges in the dual Voronoi diagram; a watershed cut algorithm [60] is applied to this weighted graph to partition nodes into cell and exterior regions, resulting *in fine* in a multimaterial nonmanifold polygonal mesh segmentation (resp. triangle surface mesh) of the original cell membrane image. **b)** The geometric precision of our mesh generation algorithm is benchmarked on foam-like simulations, which are transformed into artificial images to reconstruct surface meshes. Our pipeline reconstructs cell geometry with better precision than state-of-the-art mesh generation methods, such as CGAL [55], as shown by the comparison of the error in the reconstructed angles as a function of the original angle. The shaded region measures the standard deviation.

The balance of forces in a foam-like tissue of n_c cells can also be derived from the minimization of surface energy under cell volume constraints. This formulation is particularly adapted to numerical simulations on a discrete mesh [50, 51] and is based on a Lagrangian function. Assigning an index m to each existing interface $\mathcal{A}_m$ between one cell and another or the external medium, the Lagrangian reads

$$\mathcal{L} = \sum_{m=1}^{n_m} \gamma_m \mathcal{A}_m - \sum_{k=1}^{n_c} p_k (\mathcal{V}_k - \mathcal{V}_k^0), \tag{3.3}$$

where γ_m and A_m are, respectively, the surface tension and the area of the interface between the regions a_m and b_m where $\{a_m, b_m\} \in [\![0, N]\!]^2$. $\mathcal{V}_k$ and $\mathcal{V}_k^0$ are the current and target volumes in the cell k, and p_k its pressure plays the role of a Lagrange multiplier for volume conservation. Discretized on a mesh, where areas and volumes are functions of the positions of the n_v vertices $\{\vec{x}_\alpha\}_{\alpha=1}^{n_v}$ (Extended data Fig.1 and Supplementary Note), optimality conditions [67] for this Lagrangian function produce a force balance at each vertex $\vec{x}_\alpha$:

$$\vec{0} = \frac{\partial \mathcal{L}}{\partial \vec{x}_\alpha} = \sum_{m=1}^{n_m} \gamma_m \frac{\partial \mathcal{A}_m}{\partial \vec{x}_\alpha} - \sum_{k>1}^{n_c} p_k \frac{\partial \mathcal{V}_k}{\partial \vec{x}_\alpha} \tag{3.4}$$

We define $\Gamma = (\gamma_1, \gamma_2, \ldots, \gamma_{n_m})^T$ a generalized vector of tensions of size n_m, and $\mathrm{P} = (p_1, p_2, \ldots, p_{n_c})^T$ a generalized vector of pressures of size n_c. Inspired by projection methods [50] used generically to solve constrained optimization problems, we derive from equation

(4.6) a linear system of equations whose solutions are - to an arbitrary factor - the tensions and pressures corresponding to a given heterogeneous foam geometry (see Supplementary Note). It reads, in matrix form

$$\begin{pmatrix} \mathrm{G}_\Gamma & -\mathrm{B}_P \\ \mathrm{B}_\Gamma & -\mathrm{G}_P \end{pmatrix} \times \begin{pmatrix} \Gamma \\ \mathrm{P} \end{pmatrix} = 0, \tag{3.5}$$

where $\mathrm{G}_{\Gamma,P}$ are symmetric matrices of sizes, respectively n_m^2 and n_C^2 and $B_{\Gamma,P}$ rectangular matrices of sizes, respectively $n_m \times n_C$ and $n_C \times n_m$ (see Supplementary Note). The linear system to solve for pressures at given tensions $\mathrm{G_P} \times P = B_\Gamma \Gamma$ is called *Variational Laplace* in the next. Because the square matrix G_P is full rank and, therefore, invertible (see the Supplementary Note), we can also write a closed-form linear system for the tensions alone as $\left(\mathrm{G}_\Gamma - \mathrm{B}_P \mathrm{G}_P^{-1} \mathrm{B}_\Gamma\right) \times \Gamma = 0$, which we call *variational Young-Dupré*.

3.3 Tension and pressure inference

Tensions depend only on contact angles at trijunctions and are independent of cell pressures, so here we decompose the inverse problem into two steps, in the same spirit as [37]: first, we solve the tensions and then determine the cell pressures using inferred tension values. The advantage of this two-step approach is that tensions can still be inferred in embryos or tissues under confinement or compression (such as *C. elegans*), where Laplace's force balance does not apply anymore, since the interfaces may adopt non-uniform mean curvatures. Importantly, tensions (and pressures) are known up to a multiplicative (respectively, an additive) factor. To remove this indeterminacy, we impose that the average tensions shall be equal to unity, which adds an equation to the system, and we arbitrarily fix the external pressure to zero.

The tension inference problem can be generically cast into a linear system $\mathrm{A}_\Gamma \times \Gamma = c_\Gamma$, where A_Γ is a matrix of size $(n_\Gamma + 1) \times n_m$ that collects $n_\Gamma + 1$ equations that relate the n_m unknown tensions, and $c_\Gamma = (0, \ldots, 0, n_m)^T$ implements the constraint on the average tensions. This system is overdetermined and is solved in the sense of ordinary least squares (OLS). Performing a systematic benchmark of our method, we found that better results are obtained when the n_Γ tension equations are weighted by the length of the corresponding junction (Supplementary Note), which is the choice taken further.
In Fig. 3.3, we compare the sensitivity of our inference algorithm for the different variants of the Young-Dupré formula (3.6), (3.7), (3.8), and the *variational* Young-Dupré equation. By perturbing vertex positions with random noise in mesh solutions of foam-like simulations (Fig. 3.3a), we calculate and plot the mean square error on the tensions inferred from this perturbed mesh (Fig. 3.3b). At low noise values, we find that the scalar Young-Dupré equation gives better results, but this error increases then faster for larger noise. *Variational* Yound-Dupré and the different Lami variants have an error that increases faster at low noise, but then reaches a lower relative plateau at higher noise.

For pressure inference, we follow the same approach, expressing the inverse problem as a linear system $\mathrm{A}_P \times \mathrm{P} = b_P$, which we solve with the OLS method. Here, we compare the traditional Laplace formula (3.1) and our new *variational* Laplace formula (4.10). Interestingly, we find that our mesh-based variational formula performs systematically better regardless of the level of noise (Fig. 3.3c).

Error in inference results may originate from deviations of cells shape from the solution of an heterogeneous foam or from an insufficient image resolution (Extended data Fig. 3c), but they are also the result of an inevitable intrinsic noise generated by our pipeline that comes from the

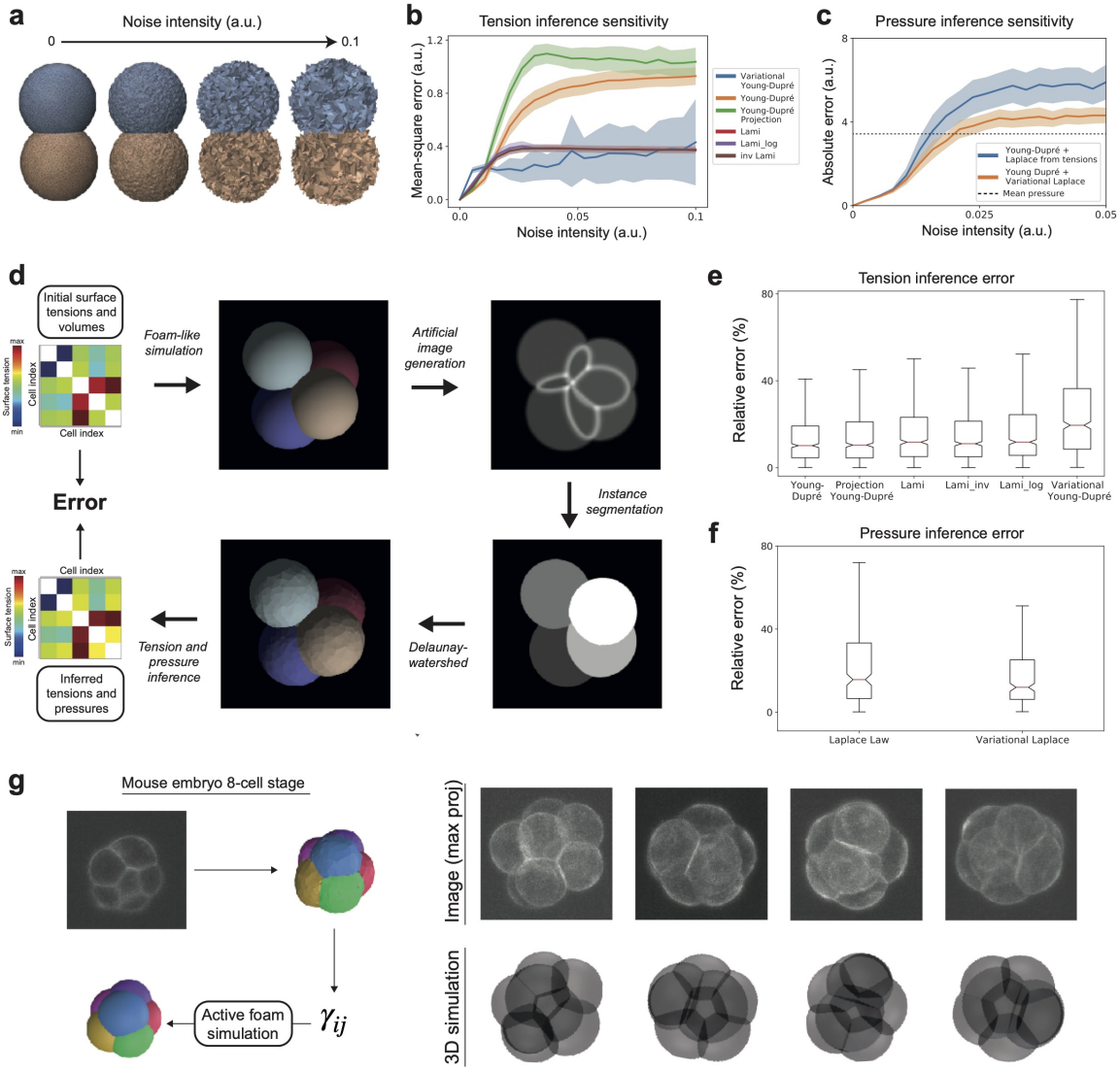


Figure 3.3: ***In silico* validation of the force inference pipeline.** **a)** Sensitivity analysis of different formulas for tension and pressure inference. Foam-like simulation meshes are perturbed by randomly displacing each vertex position following a uniform law. **b)** Plot of the mean square error on inferred tensions as a function of the intensity of the noise for the different tension formulas. **c)** Plot of the absolute error on the inferred pressure as a function of the noise intensity for the Laplace and variational Laplace formulas. **d)** Pipeline for benchmarking force inference: from random surface tensions (visualized here as a symmetrical $n_c + 1 \times n_c + 1$ matrix) and cell volume values, a dataset of foam-like embryo meshes is simulated, from which artificial microscopy images are generated; then, our end-to-end pipeline is applied to regenerate a mesh and infer tension and pressure values. **e)** Plot of the relative error in the inferred tensions for the different tension inference formulas applied to our simulated embryo dataset. **f)** Relative errors on inferred pressures on our simulated embryo dataset with Laplace and variational Laplace formulas. **g)** Self-consistent validation of the inference on the compaction of the 8-cell mouse embryo. Surface tensions are inferred with the pipeline and averaged between the cell-medium and cell-cell interfaces. Foam-like simulations are performed using these tensions and yield an *in silico* embryo morphology that is compared to the real embryo image.

segmentation and meshing operations. To evaluate which formula may be most adapted given this minimal and ineluctable level of noise, we generate ideal artificial confocal microscopy images from mesh results of foam-like simulations (see Supplementary Note). This dataset [64] is used to benchmark our method: the images are segmented using cellpose [56] and translated

into multimaterial meshes with our Delaunay-watershed algorithm to ultimately infer tensions and pressures using the various formulas introduced earlier (Fig. 3.3d). In general, we find that the systematic error induced intrinsically by our pipeline remains very low, with the best inference results obtained with the scalar Young-Dupré formula (3.6) and the variational Laplace formula (Figs. 3.3e-f). For all tension and pressure inference examples shown below, we therefore systematically use the scalar Young-Dupré and variational Laplace formula.

3.4 Force inference applied to early embryo development

To validate the biological relevance of our novel force inference pipeline, we inferred 3D mechanical atlases of mouse and ascidian embryos using fluorescent microscopy images of cell membranes. We first study the self-consistency of the heterogeneous foam model in compacting 8-cell mouse embryos. Compaction corresponds to the extension of internal cell contacts that round up the embryo and was shown by micropipette tension measurements [5] to be characterized by a decrease in the ratio $\alpha = \frac{\gamma_{cc}}{2\gamma_{cm}}$ - called *compaction parameter* - where γ_{cm} is the tension at the cell medium interface of cells and γ_{cc} the tension at cell-cell contacts. This single parameter is enough to characterize the embryo shape and is equal to the cosine of half the contact angle of the cell medium. Using confocal fluorescent images of 8-cell mouse embryos at successive levels of compaction, we segmented them into multimaterial meshes and inferred relative tensions. We then performed 3D foam-like simulations and compared them with the original microscopy images (Fig. 3.3g), and found a very good qualitative agreement. Interestingly, our automatic inference methods yields systematically lower variability in inferred γ_{cc} values than previously obtained by measuring contact angles manually [5], as illustrated on Fig. 3d. This confirms the relevance of a heterogeneous foam model hypothesis and exemplifies the capability of our inference pipeline.

To go beyond this example, where cell-medium and cell-cell tensions are uniform within the embryo, we inferred spatio-temporal mechanical atlases of the early ascidian embryo *Phallusia mammillata*. We used fluorescent images of cell membranes that were acquired with a confocal microscope from the zygote to the 44 cell stage (see Methods) or with a light sheet microscope from the 64 cell stage to the late neurula ($\lesssim$ 800 cells) [23]. We first focused on the shape of the embryo from 16 cells to the early gastrula, where divisions are reported to be asynchronous with cell divisions that alternate between the animal and vegetal hemispheres [68]. Recently, it was shown in *P. mammillata* embryos at 16, 32 and 44 cell stages, that cells at mitosis entry have lower apical tension than their interphase counterparts located in the opposite hemisphere [6]. This striking result, in notable contrast to mitotic cortical stiffening reported in most somatic cells [69, 70], is again predicted by our force inference method, which finds a ratio of apical tension between mitotic and interphase cells that is systematically lower than 1 in the 16 to 32 cell stages (Fig. 3.4e). This mitotic softening alternates between the animal and vegetal poles, as illustrated also from pressure maps (Extended data Fig. 4a) further explains the overall 3D shape of the embryo which is flatter on the side of interphase cells (16 and 32 cells). As one would expect from Laplace's law, if applied globally to the embryo approximated to a droplet, a higher apical tension at one pole leads indeed to its flattening. Inference not only confirms previous results, but also predicts an unknown switch in the 64-cell embryo, where mitotic blastomeres have higher apical tension than their interphase neighbors (Fig 4. 3.4e, Extended data Fig. 4a) suggesting that, from this stage on, cells undergo mitotic stiffening. This mitotic stiffening persists during gastrulation (stage 120 in Fig. 3.4e) and later (Extended data Fig. 4a). This illustrates the predictive power of our inference pipeline, which reveals novel mechanical features that explain the shape of cells and embryos.

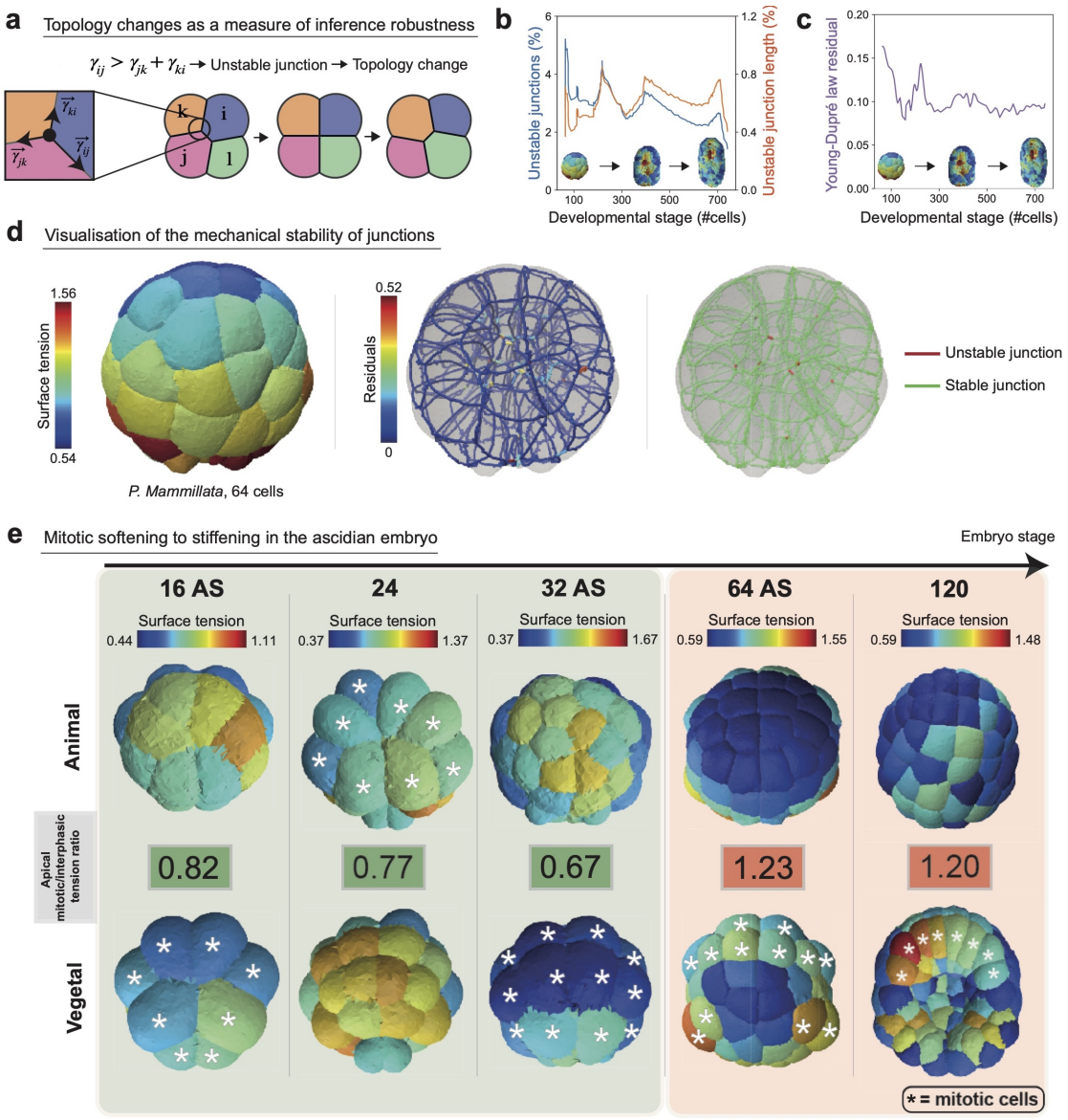


Figure 3.4: ***In vivo* validation of the 3D tension inference.** **a)** Illustration of the process of T1 topological transition when one tension at a junction becomes greater than the sum of the two others. **b)**Plot of the percentage of unstable junctions in the embryo (blue) and the ratio of unstable junction length to total junction length in the embryo as a function of its development stage, defined by its number of cells. **c)** Plot of the mean residual of Young-Dupré equations (purple) in the embryo as a function of its development stage, defined by its number of cells. **d)** *Left* Surface tension map of the 64-cell ascidian embryo. *Middle* Visualization of the residuals of Young-Dupré equations for each junction in the same embryo. *Right* Junctions for which inference predicts a T1 topological transition are mechanically unstable. Visualization of stable (green) and unstable (red) junctions in a 64-cell ascidian embryo (*P. mammillata*). **e)** Maps of apical tension at the animal and vegetal poles of the early ascidian embryo (*P. mammillata*) in the 16AS, 24, 32AS, 64 and 120 cell stages. The ratio of mitotic to interphase apical tension is colored green if it is less than 1 and red if it is greater than 1. Mitotic cells are indicated by a white star.

To further assess the validity of our inference method, we searched for locations in the embryo where the hypothesis of foam-like mechanical equilibrium may break down. An interesting idea is to look for junctions that are unstable for the predicted tensions. In fact, when

$\gamma_{ij} > \gamma_{jk} + \gamma_{ki}$, we expect the junction ij to be unstable and undergo a T1 topological transition (Fig. 3.4a). Any unstable junction is therefore the sign of mechanical equilibrium breakdown that can result either 1) from a too large error in tension inference or 2) from an inadequacy of the heterogeneous foam model to describe cell arrangement or geometry [71]. In a 64-cell *P. mammillata* embryo, we found 31 unstable junctions in a total of 569 junctions (Fig. 3.4d). Interestingly, these unstable junctions are detected exclusively close to the embryo center, where the lengths of the junctions become very small, and segmentation struggles to resolve cell geometry (Fig. 3.4d, Extended data Fig. 4b). In general, the percentage of unstable junctions predicted by our inference pipeline remains very low, around $\approx 3\%$, throughout the development of the ascidian embryo up to late neurula (Fig. 3.4b). This represents an even lower percentage of unstable junction length, below 1%, which confirms that the tension equilibrium predicted by our inference pipeline is generally satisfied. To assess the validity of the inference, it is also useful to visualize the deviation from equilibrium using the force balance at the junctions (3.2). We therefore propose a visualization of the residuals $\|\mathrm{A}_\Gamma \times \Gamma - c_\Gamma\|^2$ at each trijunction, as shown in Fig. 3.4c,d and Extended data Fig. 4b.

To further illustrate the capabilities of our inference method, we report three aspects of early ascidian embryo development brought to light by our mechanical atlases. The early development of the ascidian is characterized by its high degree of invariance [23], and a stereotypical feature of this invariance is the bilateral symmetry of the embryo. However, each embryo shows a certain degree of geometric variability between its left and right sides, which is well reflected in the mechanical asymmetry, as illustrated by the (a)symmetry of the tension and pressure maps inferred in Fig. 3.5a and Extended data Fig. 5a.

The cell fate in the ascidian embryo is also invariant, as has been described for several decades (reviewed in [23, 72]). At the 76-cell stage, the animal hemisphere is composed exclusively of ectodermal cells, while the vegetal hemisphere is segregated into neural/notochord progenitors and endoderm or mesoderm germ layers. We find that this patterning of cell fate is reflected in a remarkable manner in different regions of cell mechanics: ectoderm and endoderm cells have lower apical tension and lower pressure, while neural plate and mesoderm cells form very distinct regions of higher apical tension and pressure (Fig. 3.5b). This is probably due to the different mitotic history of each lineage, since fate specification is accompanied by an independent cell cycle timing in each specified tissue [23, 68]. In the 76-cell stage, neural/notochord and mesoderm cells have, in fact, just undergone cell division (they are in their eighth cell cycle), while endoderm cells were born more than 40 minutes ago and are in the middle of interphase, just before they undergo apical constriction [73]. In the neurula stage, apical constriction has been reported to drive neural tube closure with greater contractility on the apical side of the nerve cord and brain tissues [74, 75]. Consistent with this, our inference pipeline predicts on the vegetal side of the embryo at 395, 702 and 758-cell stage a high apical tension in cells located in the anterior neural plate that are undergoing folding (Fig. 3.5c arrow in the vegetal pole view, Extended data Fig. 5b). A sagittal section of the embryo at this stage reveals that the neural tube has more cortical tension than the overlying epidermis of the underlying endoderm and notochord (Fig. 3.5c sagittal section); this higher tension is reflected in a stronger accumulation of myosin II in the neural tube compared to other tissues (Fig. 3.5c, myosin sagittal section and see also [74]).

Finally, we performed tension inference in the early *C. elegans* embryo from 4 to 15 cells (Fig. 3.6). Unlike the ascidian and, to a certain extent, mouse embryos, an eggshell strongly constrains the shape of cells from the zygote stage. This confinement has shown to be an essential cue controlling early cell arrangement [76, 77] and makes Laplace's law no longer

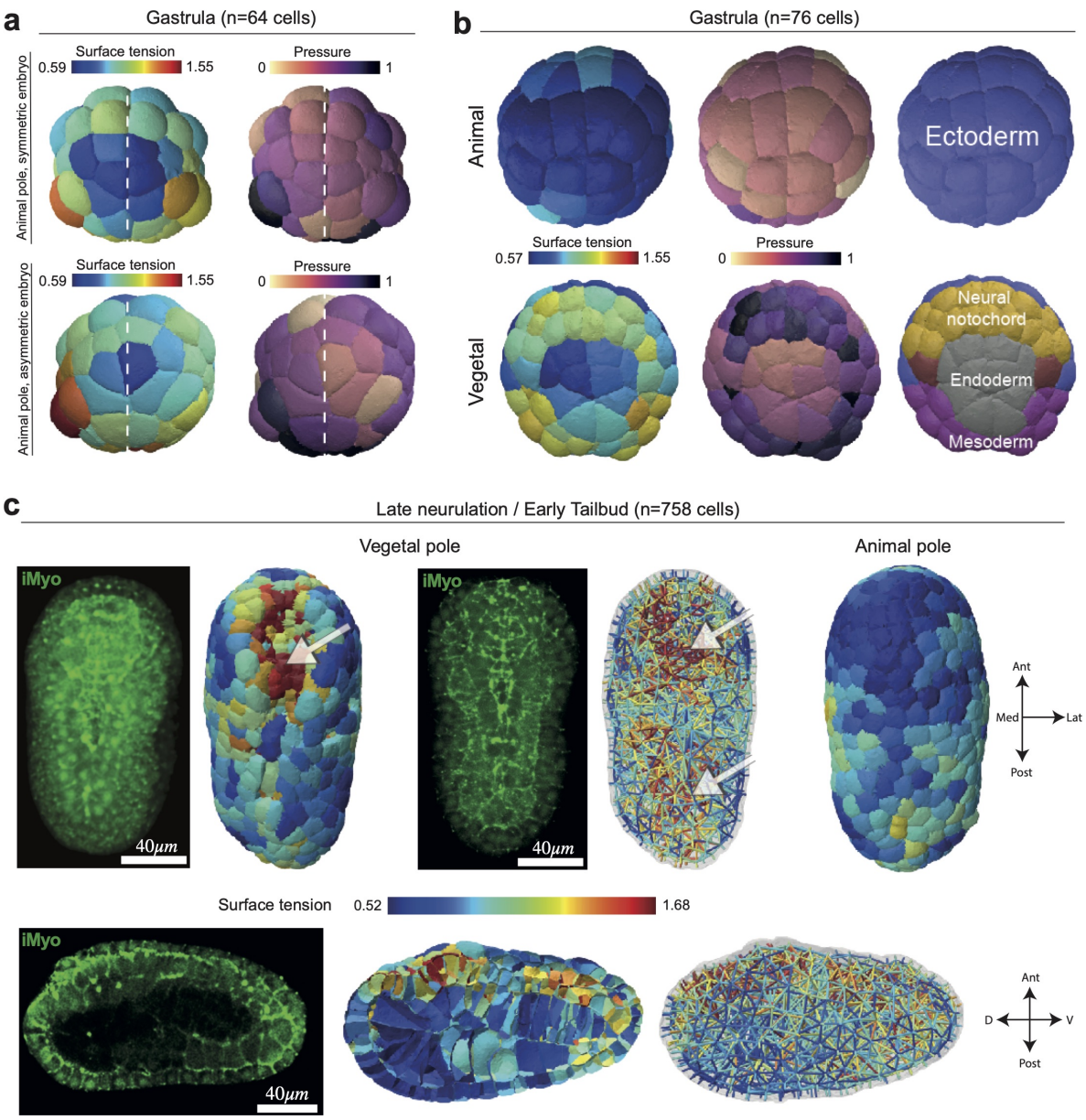


Figure 3.5: **Spatiotemporal patterning of mechanics in the ascidian embryo *P. mammillata*** **a)** Tension and pressure maps of the animal pole of two 64 cell embryos. Imperfections in the geometric bilateral symmetry of the embryo are reflected by a corresponding asymmetry in the apical tension and pressure of the cell. **b)** Tension and pressure maps at the animal and vegetal poles of a 76-cell embryo and the corresponding pattern of cell fate in the germ layers. **c)** Tension maps in late neurula (758 cells) from vegetal, animal, and sagittal views. The white arrows indicate regions of higher tension (red) within the embryo. Fluorescent microscopy images of myosin II (iMyo) at the vegetal pole and in the sagittal view: 3D reconstruction (top left) or selective plane projection of 10 confocal planes (top middle and bottom left). The orientation of the embryo is given by arrows Ant: anterior, Pos: posterior, Med: medial, Lat: lateral, D: dorsal, V: ventral.

adequate to account for cell pressures, which are directly affected by the mechanical resistance of the shell. We confirm this characteristic with 3D simulations of a 4 cell embryo confined within an ellipsoid (Fig. 3.6a), using realistic parameters that we previously measured in [77]. In this realistic simulation, we show that the mean curvature may be locally perturbed by the shell along cell-medium interfaces, especially for ABp and EMS blastomeres, which

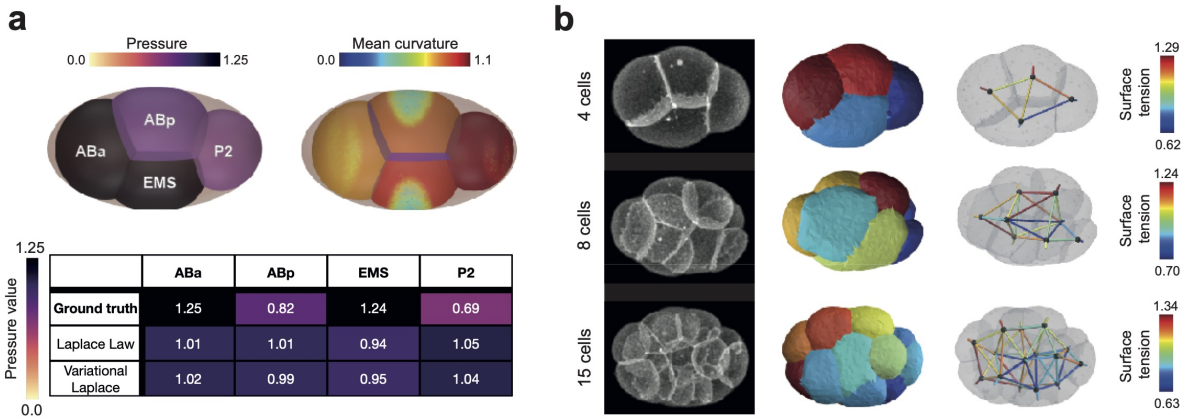


Figure 3.6: **Force inference in *C.elegans* embryo. a)** Pressure inference on a simulated embryo confined in a rigid shell. The shell induces deformations in the membrane that lead to spatial changes in curvatures compared to those of an isolated foam. Both the Laplace and the variational Laplace formulas are inadequate to infer correct pressures, as illustrated by the table of values. **b)** Surface tension can still be inferred, as equilibrium at junctions is still verified.

precludes the use of Laplace's law, which assumes constant mean curvature interfaces. Indeed, when we infer pressures with the Laplace or Laplace variational formula on this mesh, we obtain pressure predictions, which are 20% to 30% different from the actual value in the four blastomeres (Fig. 3.6a). Therefore, simultaneous tension and pressure inference may not be a good strategy in this case [40], while breaking down the inference in two successive steps still allows us to infer tensions independently of cell pressures. Interestingly, we find, in agreement with the measurements in [77], a lower cell-medium tensions in P2 and EMS cells in the 4-cell stage *C. elegans* embryo, and predict a general trend of lower cell-medium cortical tension in descendants of the P-lineage at subsequent stages of embryo development (Fig. 3.6b).

3.5 Discussion

We presented a robust end-to-end computational pipeline to infer relative surface tensions and pressures directly from three-dimensional fluorescent images of embryos or tissues. It is based, in particular, on a novel and fast method for generating surface meshes from cell segmentation masks, which allows for a more accurate extraction of geometric features than previous approaches [55]. Therefore, our algorithm is compatible with the latest segmentation methods [56, 78–80] and can scale to thousands of cells. We also introduced a novel formula for inferring pressures from a triangle surface mesh, which outperforms the direct inversion of Laplace's law. By performing a systematic sensitivity analysis on simulated embryos, we showed that the classic Young-Dupré formula gives the best tension inference results for moderate noise in the image or in the cell shape. Our pipeline intrinsically achieves maximum relative force errors of $\approx 10\%$ from images of simulated embryos (Fig. 3.3c and Supplementary Note). Additionally, we provide several visualization tools to display multicellular morphology and forces in multiple ways, including a force graph representation of the cell aggregate and a 3D map of cellular stress tensors (Figs. 3.1h-i). The residues and predicted topological changes of inference for each junction in the aggregate can also be directly plotted to enable local evaluation of the method and/or the active foam hypothesis (Fig. 3.4b,c). Subsequently, we demonstrated the biological relevance of our approach by generating mechanical atlases of the early ascidian embryo: our inference method can recover characteristic patterns of apical tension previously observed [74], including a lower apical tension measured in mitotic cells

before 64-cell stage [6]. Interestingly, it can also make new predictions and reveal mirroring patterns of cell mechanics and cell fate in germ layers. Finally, we demonstrate the utility of decoupling pressure and tension inference by applying our methodology to the early *C.elegans* embryo, which develops within a shell.

One forthcoming challenge will be to generate spatio-temporal mechanical atlases of various embryos. Indeed, a temporal reference is so far missing to calibrate the successive spatial maps in time. As demonstrated in 2D [77], combining static inference with the temporal measurement of absolute forces in a single location, or imaging phosphomyosin fluorescence intensity as a proxy for tension, could become a generic approach to construct temporal atlases of absolute mechanical forces, but this needs to be repeated in 3D.

A second challenge will involve the inclusion of junctional mechanics in the form of additional line tension contributions at the apical surface of cells. Indeed, blastomeres with a contact to the cell medium acquire generally apico-basal polarity short before the blastula stage in early embryos. This emergence of apical polarity is generally associated with the formation of tight junctions and a contractile ring of actomyosin delimiting each apical surface [81, 82], that is expected to create additional line tensions at tricellular junctions. The question of the uniqueness of the inverse solution will furthermore arise, since several stable discontinuous bifurcation states can exist in the presence of line and surface tensions [83,84], which will first require a in-depth theoretical effort.

A third challenge will consist of generalizing force inference methods to more complex mechanical models, such as recent active viscous surface models [52, 83, 85–88], which naturally generate inhomogeneous and anisotropic surface tensions, as well as possible torques, leading to more complex shapes and force balance equations. This will be particularly important for precisely characterizing the mechanics of dividing cells and faster growing organisms, such as *C. elegans*, for which the time scales of visco-active relaxation and development may no longer be well separated. A possible generic avenue to solve these problems may lie in a fully variational approach, where a mathematical loss between the microscopy images and the meshes could be constrained by an arbitrary mechanical model to allow direct gradient-based optimization of its spatio-temporal parameters. Our recent effort to design such an efficient loss for comparing a mesh and an image may begin to fill this gap [89]. Importantly, the current force inference method we introduced will remain a fundamental building block to this research field, providing already accurate geometric and mechanical maps, which will form an ideal initial guess to refined but more computationally expensive iterative methods.

With a documented and user-friendly implementation in Python [90], our 3D force inference method can be easily applied to 3D images of embryos or small tissues undergoing a sufficiently slow development, and can be combined with spatial "omic" data generated in early embryos to uncover possible mechanochemical couplings. 3D force inference complements the growing range of tools available for studying the mechanical properties of tissues in space and time [25, 91, 92], and we anticipate that this approach will help elucidate the mechanical underpinnings of large-scale morphogenetic movements at the cellular level and illuminate the intricate interplay between chemical signaling and mechanics during development [93–95]. By revealing the developmental forces shaping organisms, our method may open new evo-devo studies, such as the investigation of the mechanical differences between closely related phylogenetic neighbors or the understanding of the mechanical aspects contributing to the divergence of developmental pathways in evolution.

Data availability

Images and segmentation masks are already available publicly for *P. Mammillata* embryos on figshare ($\geq$ 64 cells) [23] and for *C. elegans* embryos on figshare [22]. The simulated dataset (artificial images, segmentation masks and tensions/pressures) used to benchmark the method is available publicly on Zenodo [64]. Additional experimental images of ascidian embryos ($<$ 64 cells) and their segmentation masks used generated this study will be made available on Zenodo upon publication.

Code availability

The separate source code for mesh reconstruction with Delaunay-watershed will be available on GitHub [65] upon publication. All other scripts and main code used in this paper will be available upon publication on GitHub [90] and the version released at publication will be archived on Zenodo.

3.6 Methods

Variants of Young-Dupré formulas

Starting from the vectorial expression of the Young-Dupré law (3.2) we call its decomposition simply by *Young-Dupré* its decomposition with cosines of polar angles:

$$\begin{aligned} \gamma_{ij} + \gamma_{jk}\cos\alpha_{ki} + \gamma_{ki}\cos\alpha_{jk} &= 0 \\ \gamma_{ij}\cos\alpha_{ki} + \gamma_{jk} + \gamma_{ki}\cos\alpha_{ij} &= 0 \\ \gamma_{ij}\cos\alpha_{jk} + \gamma_{jk}\cos\alpha_{ij} + \gamma_{ki} &= 0 \end{aligned} \tag{3.6}$$

Another set involves both cosines and sines of angles made by vectorial tensions with one direction chosen arbitrarily choose along a tension vector, and we call it *Young-Dupré projection*:

$$\begin{aligned} \gamma_{ij} + \gamma_{jk}\cos\alpha_{ki} + \gamma_{ki}\cos\alpha_{jk} &= 0 \\ \gamma_{jk}\sin\alpha_{ki} - \gamma_{ki}\sin\alpha_{jk} &= 0 \end{aligned} \tag{3.7}$$

Many other mathematically equivalent formulas may in fact be derived from trigonometric laws applied to the triangle formed by vectorial tensions (see Supplementary Note). Here, we will also use Lami's theorem, which derives directly from the law of sines and was proposed as an alternative formula for tension inference in 2D [41, 66]:

$$\frac{\gamma_{ij}}{\sin\alpha_{ij}} = \frac{\gamma_{jk}}{\sin\alpha_{jk}} = \frac{\gamma_{ki}}{\sin\alpha_{ki}} \tag{3.8}$$

To avoid divergence at small polar angles, it was proposed to consider the same equations written as $\gamma_{ij}\sin\alpha_{jk} = \gamma_{jk}\sin\alpha_{ij}$, $\gamma_{jk}\sin\alpha_{ki} = \gamma_{ki}\sin\alpha_{jk}$, which we call *inverse Lami*, or to consider the logarithm of the equation (3.8), that we call *Lami logarithm*.

Biological material

The eggs of the ascidian *Phallusia mammillata* were harvested from animals obtained in Sète and kept in the laboratory in a tank of natural seawater at 16°C. Egg preparation and microinjection have been previously described (see detailed protocols in [96], [97]). Eggs and sperm were collected by dissection. Sperm was activated in pH 9.0 seawater prior to fertilization (see the detailed protocol in [97]). All imaging experiments were performed at 20°C.

Plasma membrane and myosin-II fluorescent labeling

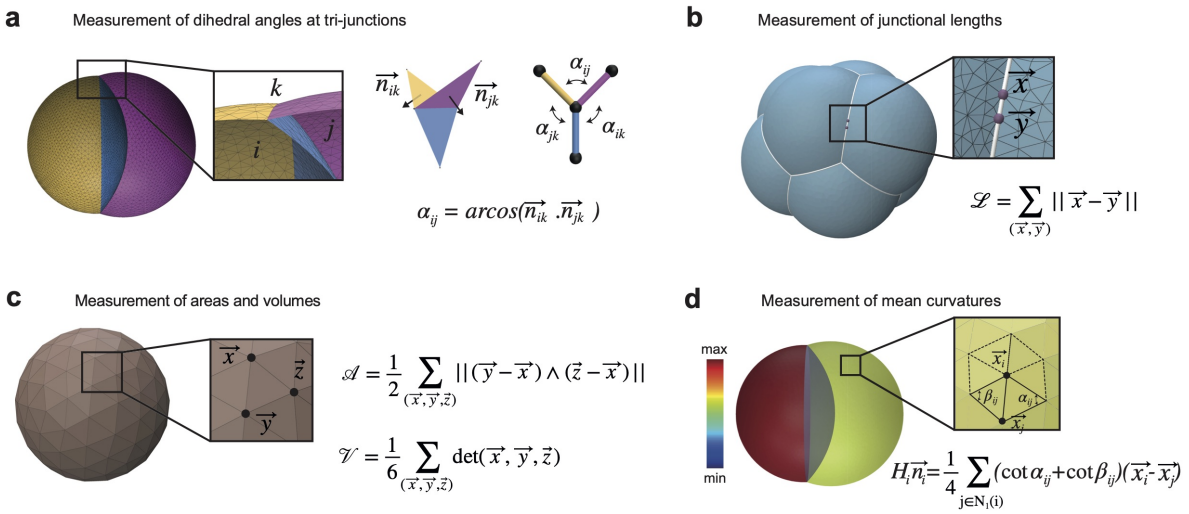
The plasma membrane was imaged using our characterized construct PH::Tomato [97] whereas Myosin II was imaged using Myosin II intrabody iMyo (called SF9::GFP in Chaigne et al., 2016, the plasmid pRN3-SF9-GFP is a kind gift from the M.H. Verlhac laboratory). RNAs coding for PH::Tomato (1 µg.µL-1) and SF9/iMyo::GFP (4 µg.µL-1) were injected in unfertilized Phallusia oocytes that were then fertilized between 2 and 12 hours after injection.

Confocal imaging of *Phallusia mammillata* embryos

4D confocal imaging was performed at 20 ° C using a Leica TCS SP8 inverted microscope equipped with hybrid detectors and a 20×/0.8NA water objective lens. A 3D stack was taken every minute with a pixel size of 1µm x 1µm and a z-step of 1 µm (in order to obtain cubic voxels). The Phallusia embryos shown in Fig. 4d (and Extended data Fig. 4) from 16 cells to 32 cells were imaged in the Team ABC laboratory, while embryos from stage 64 cells and later stages (shown in Fig. 1, Fig. 4, Fig. 5, Ext Fig. 4, Ext Fig. 5) were obtained from a public dataset of segmented *P. mammillata* embryos published in [23].

Statistical Analysis

The boxplots (shown in Fig. 3e, Fig. 3f, Ext Fig. 3b) are realized with the default parameters of the boxplot function of the matplotlib python library. The box center is located at the median, and its extremities represents the first and third quartiles. The whiskers are located at Q1 - 1.5*(Q3-Q1) and Q3 + 1.5*(Q3-Q1).
The shaded regions in plots displays the standard deviation (in Fig. 3b, Fig. 3c, Fig. 2b, Ext Fig. 2b).



Extended data Fig. 1: **Measurement of geometrical quantities on nonmanifold multimaterial triangle surface meshes. a)** Contact angles are calculated at each junction as the mean of dihedral angles in each triplet of triangles that constitutes the junction. A dihedral angle is computed from the unit normals to the two adjacent triangles. **b)** Junctions are lines that separate three different materials or regions (three cells or 2 cells and the cell medium). Their length can be easily defined and measured with our nonmanifold mesh data structure. **c)** Each cell is represented by a bounded volume (a discrete manifold). We can compute their volumes and areas from our multimaterial mesh data structure with formulas derived in the Supplementary Note. **d)** Mean discrete curvatures can be computed using the cotangent formula (see Supplementary Note).

Bibliography

[1] Charles Gawad, Winston Koh, and Stephen R Quake. Single-cell genome sequencing: current state of the science. *Nature Reviews Genetics*, 17(3):175–188, 2016.

[2] Byungjin Hwang, Ji Hyun Lee, and Duhee Bang. Single-cell rna sequencing technologies and bioinformatics pipelines. *Experimental & molecular medicine*, 50(8):1–14, 2018.

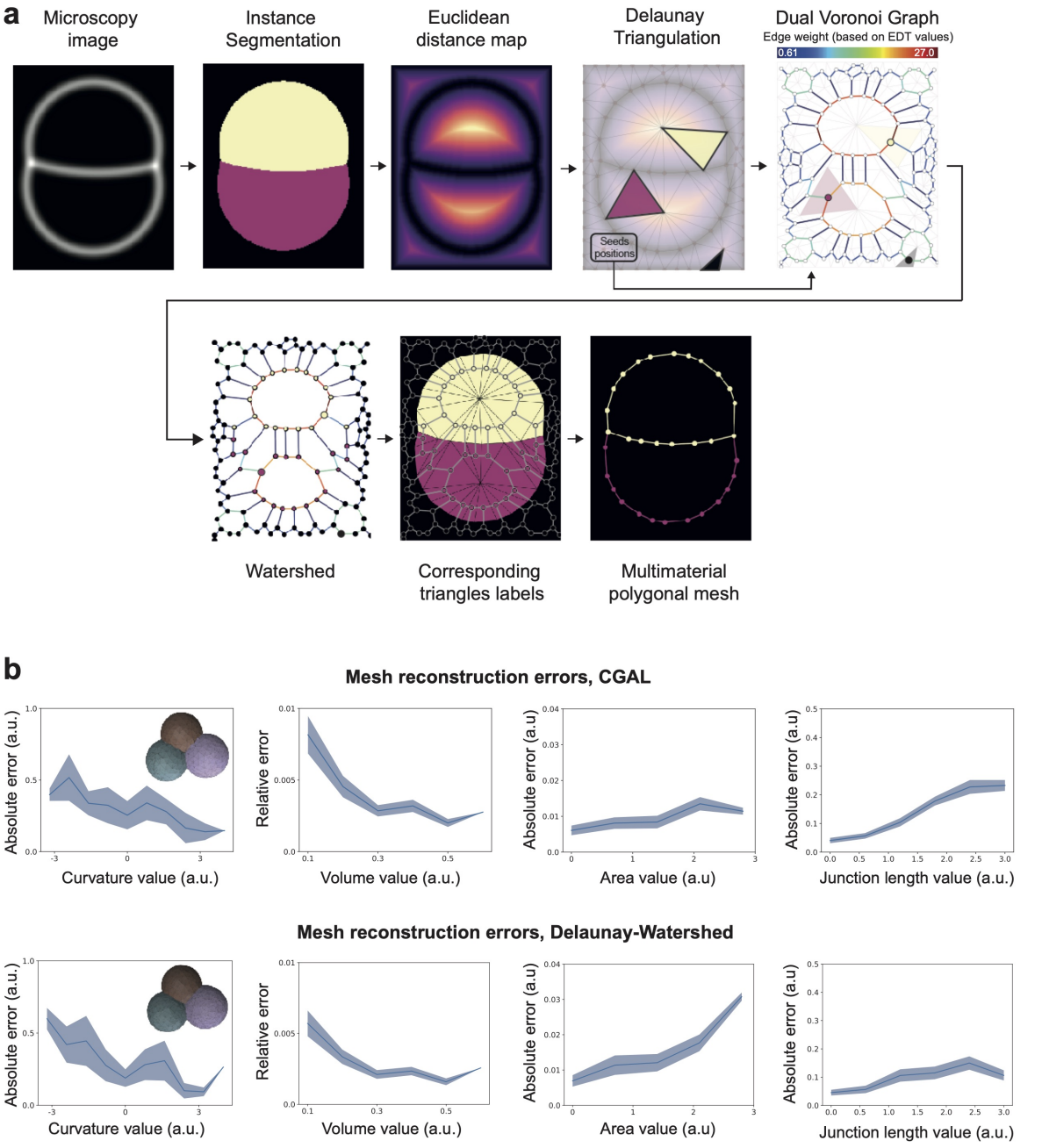
[3] JMt Mitchison and MM Swann. The mechanical properties of the cell surface. *J. exp. Biol*, 31(3):443–460, 1954.

[4] Karine Guevorkian, Marie-Josée Colbert, Mélanie Durth, Sylvie Dufour, and Françoise Brochard-Wyart. Aspiration of biological viscoelastic drops. *Physical review letters*, 104(21):218101, 2010.

[5] Jean-Léon Maître, Ritsuya Niwayama, Hervé Turlier, François Nédélec, and Takashi Hiiragi. Pulsatile cell-autonomous contractility drives compaction in the mouse embryo. *Nature cell biology*, 17(7):849–855, 2015.

[6] Benoit G Godard, Rémi Dumollard, Edwin Munro, Janet Chenevert, Céline Hebras, Alex McDougall, and Carl-Philipp Heisenberg. Apical relaxation during mitotic rounding promotes tension-oriented cell division. *Developmental Cell*, 55(6):695–706, 2020.

[7] Karel Svoboda and Steven M Block. Biological applications of optical forces. *Annual review of biophysics and biomolecular structure*, 23(1):247–285, 1994.



Extended data Fig. 2: **Detailed procedure and benchmarking of the Delaunay-watershed mesh generation algorithm. a)** Pipeline for mesh generation from a microscopy image (here in 2D for graphical purposes). From the Delaunay triangulation of the image domain, we construct a graph of the dual Voronoi diagram . The edge weights of this graph are computed by integrating the value of the Euclidean distance map along corresponding edges that separates two triangles in the primary domain. The watershed is performed on the dual graph, and the seeds are chosen by taking the triangles containing the pixel with the highest EDT value in the primary domain. **b)** Comparison of the geometric error obtained on interface curvatures, cell volumes, interface areas, and junctional lengths with CGAL and our Delaunay-watershed algorithms for mesh reconstruction.

[8] Monica Tanase, Nicolas Biais, and Michael Sheetz. Magnetic tweezers in cell biology. *Methods in cell biology*, 83:473–493, 2007.

[9] Kapil Bambardekar, Raphaël Clément, Olivier Blanc, Claire Chardès, and Pierre-François Lenne. Direct laser manipulation reveals the mechanics of cell contacts in vivo. *Proceedings of the National Academy of Sciences*, 112(5):1416–1421, 2015.

[10] Johannes Rheinlaender, Andrea Dimitracopoulos, Bernhard Wallmeyer, Nils M Kronenberg, Kevin J Chalut, Malte C Gather, Timo Betz, Guillaume Charras, and Kristian Franze. Cortical cell stiffness is independent of substrate mechanics. *Nature materials*, 19(9):1019–1025, 2020.

[11] Yuki Fujii, Wataru C Koizumi, Taichi Imai, Megumi Yokobori, Tomohiro Matsuo, Kotaro Oka, Kohji Hotta, and Takaharu Okajima. Spatiotemporal dynamics of single cell stiffness in the early developing ascidian chordate embryo. *Communications biology*, 4(1):1–12, 2021.

[12] Otger Campàs, Tadanori Mammoto, Sean Hasso, Ralph A Sperling, Daniel O'connell, Ashley G Bischof, Richard Maas, David A Weitz, Lakshminarayanan Mahadevan, and Donald E Ingber. Quantifying cell-generated mechanical forces within living embryonic tissues. *Nature methods*, 11(2):183–189, 2014.

[13] Friedhelm Serwane, Alessandro Mongera, Payam Rowghanian, David A Kealhofer, Adam A Lucio, Zachary M Hockenbery, and Otger Campas. In vivo quantification of spatially varying mechanical properties in developing tissues. *Nature methods*, 14(2):181–186, 2017.

[14] Alexandre Souchaud, Arthur Boutillon, Gaëlle Charron, Atef Asnacios, Camille Noûs, Nicolas B David, François Graner, and François Gallet. Live 3d imaging and mapping of shear stresses within tissues using incompressible elastic beads. *Development*, 149(4):dev199765, 2022.

[15] Ariadna Marín-Llauradó, Sohan Kale, Adam Ouzeri, Raimon Sunyer, Alejandro Torres-Sánchez, Ernest Latorre, Manuel Gómez-González, Pere Roca-Cusachs, Marino Arroyo, and Xavier Trepat. Mapping mechanical stress in curved epithelia of designed size and shape. *bioRxiv*, pages 2022–05, 2022.

[16] L V Beloussov, J G Dorfman, and V G Cherdantzev. Mechanical stresses and morphological patterns in amphibian embryos. *Journal of embryology and experimental morphology*, 34(3):559–574, December 1975.

[17] Matteo Rauzi, Pascale Verant, Thomas Lecuit, and Pierre-François Lenne. Nature and anisotropy of cortical forces orienting drosophila tissue morphogenesis. *Nature cell biology*, 10(12):1401–1410, 2008.

[18] Cyrille Norotte, Francoise Marga, Adrian Neagu, Ioan Kosztin, and Gabor Forgacs. Experimental evaluation of apparent tissue surface tension based on the exact solution of the laplace equation. *EPL (Europhysics Letters)*, 81(4):46003, 2008.

[19] Gabor Forgacs, Ramsey A Foty, Yinon Shafrir, and Malcolm S Steinberg. Viscoelastic properties of living embryonic tissues: a quantitative study. *Biophysical journal*, 74(5):2227–2234, 1998.

[20] Francois Mazuel, Myriam Reffay, Vicard Du, Jean-Claude Bacri, Jean-Paul Rieu, and Claire Wilhelm. Magnetic flattening of stem-cell spheroids indicates a size-dependent elastocapillary transition. *Physical review letters*, 114(9):098105, 2015.

[21] Gaëtan Mary, François Mazuel, Vincent Nier, Florian Fage, Irène Nagle, Louisiane Devaud, Jean-Claude Bacri, Sophie Asnacios, Atef Asnacios, Cyprien Gay, et al. All-in-one rheometry and nonlinear rheology of multicellular aggregates. *Physical Review E*, 105(5):054407, 2022.

[22] Jianfeng Cao, Guoye Guan, Vincy Wing Sze Ho, Ming-Kin Wong, Lu-Yan Chan, Chao Tang, Zhongying Zhao, and Hong Yan. Establishment of a morphological atlas of the caenorhabditis elegans embryo using deep-learning-based 4d segmentation. *Nature communications*, 11(1):1–14, 2020.

[23] Léo Guignard, Ulla-Maj Fiúza, Bruno Leggio, Julien Laussu, Emmanuel Faure, Gaël Michelin, Kilian Biasuz, Lars Hufnagel, Grégoire Malandain, Christophe Godin, et al. Contact area–dependent cell communication and the morphological invariance of ascidian embryogenesis. *Science*, 369(6500):eaar5663, 2020.

[24] Katie McDole, Léo Guignard, Fernando Amat, Andrew Berger, Grégoire Malandain, Loïc A Royer, Srinivas C Turaga, Kristin Branson, and Philipp J Keller. In toto imaging and reconstruction of post-implantation mouse development at the single-cell level. *Cell*, 175(3):859–876, 2018.

[25] Robert Prevedel, Alba Diz-Muñoz, Giancarlo Ruocco, and Giuseppe Antonacci. Brillouin microscopy: an emerging tool for mechanobiology. *Nature methods*, 16(10):969–977, 2019.

[26] Carlo Bevilacqua, Juan Manuel Gomez, Ulla-Maj Fiuza, Chii Jou Chan, Ling Wang, Sebastian Hambura, Manuel Eguren, Jan Ellenberg, Alba Diz-Muñoz, Maria Leptin, and Robert Prevedel. High-resolution line-scan Brillouin microscopy for live imaging of mechanical properties during embryo development. *Nature Methods*, 2023.

[27] Paul R Selvin. The renaissance of fluorescence resonance energy transfer. *Nature structural biology*, 7(9):730–734, 2000.

[28] Charlène Gayrard and Nicolas Borghi. Fret-based molecular tension microscopy. *Methods*, 94:33–42, 2016.

[29] Adai Colom, Emmanuel Derivery, Saeideh Soleimanpour, Caterina Tomba, Marta Dal Molin, Naomi Sakai, Marcos González-Gaitán, Stefan Matile, and Aurélien Roux. A fluorescent membrane tension probe. *Nature chemistry*, 10(11):1118–1125, 2018.

[30] Chloé Roffay, Chii J Chan, Boris Guirao, Takashi Hiiragi, and François Graner. Inferring cell junction tension and pressure from cell geometry. *Development*, 148(18):dev192773, 2021.

[31] S Ishihara, K Sugimura, SJ Cox, Isabelle Bonnet, Y Bellaïche, and F Graner. Comparative study of non-invasive force and stress inference methods in tissue. *The European Physical Journal E*, 36(4):1–13, 2013.

[32] Thomas Lecuit and Pierre-Francois Lenne. Cell surface mechanics and the control of cell shape, tissue patterns and morphogenesis. *Nature reviews Molecular cell biology*, 8(8):633–644, 2007.

[33] Shuji Ishihara and Kaoru Sugimura. Bayesian inference of force dynamics during morphogenesis. *Journal of theoretical biology*, 313:201–211, 2012.

[34] Kevin K Chiou, Lars Hufnagel, and Boris I Shraiman. Mechanical stress inference for two dimensional cell arrays. *PLoS computational biology*, 8(5):e1002512, 2012.

[35] Tatsuzo Nagai and Hisao Honda. A dynamic cell model for the formation of epithelial tissues. *Philosophical Magazine B*, 81(7):699–719, 2001.

[36] Reza Farhadifar, Jens-Christian Röper, Benoit Aigouy, Suzanne Eaton, and Frank Jülicher. The influence of cell mechanics, cell-cell interactions, and proliferation on epithelial packing. *Current biology*, 17(24):2095–2104, 2007.

[37] G Wayne Brodland, Jim H Veldhuis, Steven Kim, Matthew Perrone, David Mashburn, and M Shane Hutson. Cellfit: a cellular force-inference toolkit using curvilinear cell boundaries. *PloS one*, 9(6):e99116, 2014.

[38] Weiyuan Kong, Olivier Loison, Pruthvi Chavadimane Shivakumar, Eunice HoYee Chan, Mehdi Saadaoui, Claudio Collinet, Pierre-François Lenne, and Raphaël Clément. Experimental validation of force inference in epithelia from cell to tissue scale. *Scientific reports*, 9(1):1–12, 2019.

[39] Jim H Veldhuis, Ahmad Ehsandar, Jean-Léon Maître, Takashi Hiiragi, Simon Cox, and G Wayne Brodland. Inferring cellular forces from image stacks. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 372(1720):20160261, 2017.

[40] Muzhi Xu, Yicong Wu, Hari Shroff, Min Wu, and Madhav Mani. A scheme for 3-dimensional morphological reconstruction and force inference in the early c. elegans embryo. *PLoS One*, 13(7):e0199151, 2018.

[41] Nicholas Noll, Sebastian J Streichan, and Boris I Shraiman. Variational method for image-based inference of internal stress in epithelial tissues. *Physical Review X*, 10(1):011072, 2020.

[42] Jean-Daniel Boissonnat and Menelaos I Karavelas. On the combinatorial complexity of euclidean voronoi cells and convex hulls of d-dimensional spheres. In *SODA*, volume 3, pages 305–312, 2003.

[43] Jean-Daniel Boissonnat, Camille Wormser, and Mariette Yvinec. Curved voronoi diagrams. In *Effective Computational Geometry for Curves and Surfaces*, pages 67–116. Springer, 2006.

[44] David Eppstein. A möbius-invariant power diagram and its applications to soap bubbles and planar lombardi drawing. *Discrete & Computational Geometry*, 52(3):515–550, 2014.

[45] John Sullivan. Nonspherical bubble clusters. In *Proceedings of Bridges 2014: Mathematics, Music, Art, Architecture, Culture*, pages 453–456, 2014.

[46] Siqi Liu, Patrick Lemaire, Edwin Munro, and Madhav Mani. A mathematical theory for the mechanics of three-dimensional cellular aggregates reveals the mechanical atlas for ascidian embryogenesis. *bioRxiv*, pages 2022–11, 2022.

[47] GK Batchelor. The stress system in a suspension of force-free particles. *Journal of fluid mechanics*, 41(3):545–570, 1970.

[48] Mark Meyer, Mathieu Desbrun, Peter Schröder, and Alan H Barr. Discrete differential-geometry operators for triangulated 2-manifolds. In *Visualization and mathematics III*, pages 35–57. Springer, 2003.

[49] Keenan Crane. Discrete differential geometry: An applied introduction. *Notices of the AMS, Communication*, 1153, 2018.

[50] Kenneth A Brakke. The surface evolver. *Experimental mathematics*, 1(2):141–165, 1992.

[51] Jean-Léon Maître, Hervé Turlier, Rukshala Illukkumbura, Björn Eismann, Ritsuya Niwayama, François Nédélec, and Takashi Hiiragi. Asymmetric division of contractile domains couples cell positioning and fate specification. *Nature*, 536(7616):344–348, 2016.

[52] Hudson Borja da Rocha, Jeremy Bleyer, and Hervé Turlier. A viscous active shell theory of the cell cortex. *Journal of the Mechanics and Physics of Solids*, 164:104876, 2022.

[53] Fang Da, Christopher Batty, and Eitan Grinspun. Multimaterial mesh-based surface tracking. *ACM Trans. Graph.*, 33(4):112–1, 2014.

[54] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 21(4):163–169, 1987.

[55] Pierre Alliez, Clément Jamin, Laurent Rineau, Stéphane Tayeb, Jane Tournois, and Mariette Yvinec. 3D mesh generation. In *CGAL User and Reference Manual*. CGAL Editorial Board, 5.5.1 edition, 2022.

[56] Carsen Stringer, Tim Wang, Michalis Michaelos, and Marius Pachitariu. Cellpose: a generalist algorithm for cellular segmentation. *Nature methods*, 18(1):100–106, 2021.

[57] Per-Erik Danielsson. Euclidean distance mapping. *Computer Graphics and image processing*, 14(3):227–248, 1980.

[58] Min Bai and Raquel Urtasun. Deep watershed transform for instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5221–5229, 2017.

[59] Weikang Wang, David A Taft, Yi-Jiun Chen, Jingyu Zhang, Callen T Wallace, Min Xu, Simon C Watkins, and Jianhua Xing. Learn to segment single cells with deep distance estimator and deep cell detector. *Computers in biology and medicine*, 108:133–141, 2019.

[60] Jean Cousty, Gilles Bertrand, Laurent Najman, and Michel Couprie. Watershed cuts: Minimum spanning forests and the drop of water principle. *IEEE transactions on pattern analysis and machine intelligence*, 31(8):1362–1374, 2008.

[61] Jörg Hendrik Kappes, Markus Speth, Björn Andres, Gerhard Reinelt, and Christoph Schn. Globally optimal image partitioning by multicuts. In *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 31–44. Springer, 2011.

[62] Alberto Bailoni, Constantin Pape, Nathan Hütsch, Steffen Wolf, Thorsten Beier, Anna Kreshuk, and Fred A Hamprecht. Gasp, a generalized framework for agglomerative clustering of signed graphs and its application to instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11645–11655, 2022.

[63] Steffen Wolf, Constantin Pape, Alberto Bailoni, Nasim Rahaman, Anna Kreshuk, Ullrich Kothe, and FredA Hamprecht. The mutex watershed: efficient, parameter-free image partitioning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 546–562, 2018.

[64] Sacha Ichbiah and Hervé Turlier. Simulation dataset to benchmark 3D force inference methods, April 2023.

[65] Ichbiah, Sacha and Turlier, Hervé. Delaunay-watershed: multimaterial surfacee mesh reconstruction from segmentation masks.

[66] Nicolas Harmand. *Pertinence et limites des tensions de surface et de ligne pour rendre compte des formes des cellules épithéliales*. PhD thesis, Université Paris Cité, 2019.

[67] Jorge Nocedal and Stephen J Wright. *Numerical optimization*. Springer, 1999.

[68] Rémi Dumollard, Celine Hebras, Lydia Besnardeau, and Alex McDougall. Beta-catenin patterns the cell cycle during maternal-to-zygotic transition in urochordate embryos. *Dev Biol*, 384(2):331–342, 2013.

[69] Martin P Stewart, Jonne Helenius, Yusuke Toyoda, Subramanian P Ramanathan, Daniel J Muller, and Anthony A Hyman. Hydrostatic pressure and the actomyosin cortex drive mitotic cell rounding. *Nature*, 469(7329):226–230, 2011.

[70] Anna V Taubenberger, Buzz Baum, and Helen K Matthews. The mechanics of mitotic cell rounding. *Frontiers in cell and developmental biology*, page 687, 2020.

[71] François Graner and Daniel Riveline. 'the forms of tissues, or cell-aggregates': D'arcy thompson's influence and its limits. *Development*, 144(23):4226–4237, 2017.

[72] Patrick Lemaire. Unfolding a chordate developmental program, one cell at a time: invariant cell lineages, short-range inductions and evolutionary plasticity in ascidians. *Dev Biol*, 332(1):48–60, Aug 2009.

[73] Kristin Sherrard, François Robin, Patrick Lemaire, and Edwin Munro. Sequential activation of apical and basolateral contractility drives ascidian endoderm invagination. *Current Biology*, 20(17):1499–1510, 2010.

[74] Hidehiko Hashimoto, Francois B Robin, Kristin M Sherrard, and Edwin M Munro. Sequential contraction and exchange of apical junctions drives zippering and neural tube closure in a simple chordate. *Developmental cell*, 32(2):241–255, 2015.

[75] Hidehiko Hashimoto and Edwin Munro. Differential expression of a classic cadherin directs tissue-level contractile asymmetry during neural tube closure. *Developmental cell*, 51(2):158–172, 2019.

[76] Kazunori Yamamoto and Akatsuki Kimura. An asymmetric attraction model for the diversity and robustness of cell arrangement in nematodes. *Development*, 144(23):4437–4449, 2017.

[77] Kazunori Yamamoto, Sacha Ichbiah, Joana Pinto, Fabrice Delbary, Nathan Goehring, Hervé Turlier, and Guillaume Charras. Dissecting the subcellular forces sculpting early c. elegans embryos. *bioRxiv*, pages 2023–03, 2023.

[78] Romain Fernandez, Pradeep Das, Vincent Mirabet, Eric Moscardi, Jan Traas, Jean-Luc Verdeil, Grégoire Malandain, and Christophe Godin. Imaging plant growth in 4d: robust tissue reconstruction and lineaging at cell resolution. *Nature methods*, 7(7):547–553, 2010.

[79] Adrian Wolny, Lorenzo Cerrone, Athul Vijayan, Rachele Tofanelli, Amaya Vilches Barro, Marion Louveaux, Christian Wenzl, Sören Strauss, David Wilson-Sánchez, Rena Lymbouridou, et al. Accurate and versatile 3d segmentation of plant tissues at cellular resolution. *Elife*, 9:e57613, 2020.

[80] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023.

[81] Seham Ebrahim, Tomoki Fujita, Bryan A Millis, Elliott Kozin, Xuefei Ma, Sachiyo Kawamoto, Michelle A Baird, Michael Davidson, Shigenobu Yonemura, Yasuo Hisa, et al. Nmii forms a contractile transcellular sarcomeric network to regulate apical cell junctions and tissue geometry. *Current biology*, 23(8):731–736, 2013.

[82] Meng Zhu, Jake Cornwall-Scoones, Peizhe Wang, Charlotte E Handford, Jie Na, Matt Thomson, and Magdalena Zernicka-Goetz. Developmental clock and mechanism of de novo polarization of the mouse embryo. *Science*, 370(6522):eabd2703, 2020.

[83] Hervé Turlier, Basile Audoly, Jacques Prost, and Jean-François Joanny. Furrow constriction in animal cell cytokinesis. *Biophysical journal*, 106(1):114–123, 2014.

[84] Edouard Hannezo, Jacques Prost, and Jean-Francois Joanny. Theory of epithelial sheet morphology in three dimensions. *Proceedings of the National Academy of Sciences*, 111(1):27–32, 2014.

[85] Mirjam Mayer, Martin Depken, Justin S Bois, Frank Jülicher, and Stephan W Grill. Anisotropies in cortical tension reveal the physical basis of polarizing cortical flows. *Nature*, 467(7315):617–621, 2010.

[86] Sundar Ram Naganathan, Sebastian Fürthauer, Masatoshi Nishikawa, Frank Jülicher, and Stephan W Grill. Active torque generation by the actomyosin cell cortex drives left–right symmetry breaking. *elife*, 3:e04165, 2014.

[87] Guillaume Salbreux and Frank Jülicher. Mechanics of active surfaces. *Physical Review E*, 96(3):032404, 2017.

[88] Alejandro Torres-Sánchez, Daniel Millán, and Marino Arroyo. Modelling fluid deformable surfaces with an emphasis on biological interfaces. *Journal of fluid mechanics*, 872:218–271, 2019.

[89] Sacha Ichbiah, Fabrice Delbary, and Hervé Turlier. Differentiable rendering for 3d fluorescence microscopy. *arXiv preprint arXiv:2303.10440*, 2023.

[90] Ichbiah, Sacha and Delbary, Fabrice and Turlier, Hervé. Foambryo: 3d inference from microscopy images.

[91] Kaoru Sugimura, Pierre-François Lenne, and François Graner. Measuring forces and stresses in situ in living tissues. *Development*, 143(2):186–196, 2016.

[92] Pere Roca-Cusachs, Vito Conte, and Xavier Trepat. Quantifying forces in cell biology. *Nature cell biology*, 19(7):742–751, 2017.

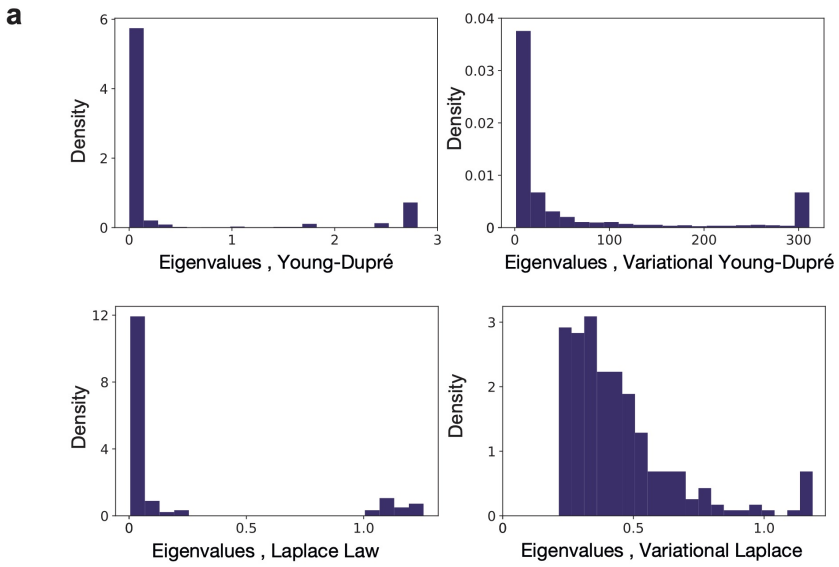
[93] Akankshi Munjal, Jean-Marc Philippe, Edwin Munro, and Thomas Lecuit. A self-organized biomechanical network drives shape changes during tissue morphogenesis. *Nature*, 524(7565):351–355, 2015.

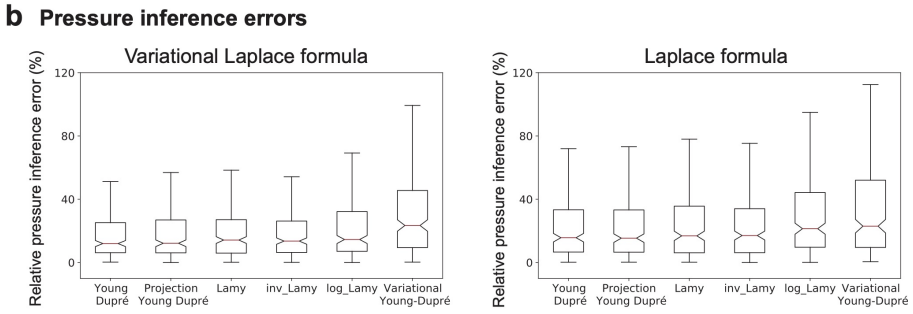
[94] Edouard Hannezo and Carl-Philipp Heisenberg. Mechanochemical feedback loops in development and disease. *Cell*, 178(1):12–25, 2019.

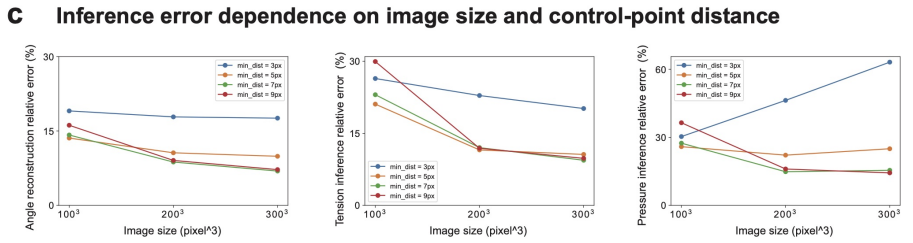
[95] Anaïs Bailles, Emily W Gehrels, and Thomas Lecuit. Mechanochemical principles of spatial and temporal patterns in cells and tissues. *Annual review of cell and developmental biology*, 38:321–347, 2022.

[96] Alex McDougall, Karen W Lee, and Rémi Dumollard. Microinjection and 4d fluorescence imaging in the eggs and embryos of the ascidian phallusia mammillata. *Methods Mol Biol*, 1128:175–185, 2014.

[97] Alex McDougall, Janet Chenevert, Gaelle Pruliere, Vlad Costache, Celine Hebras, Guillaume Salez, and Remi Dumollard. Centrosomes and spindles in ascidian embryos and eggs. In *Methods in Cell Biology*, volume 129, pages 317–339. Elsevier, 2015.







d **Comparison of manually and automatically inferred tensions variability in the 8-cell mouse embryo**

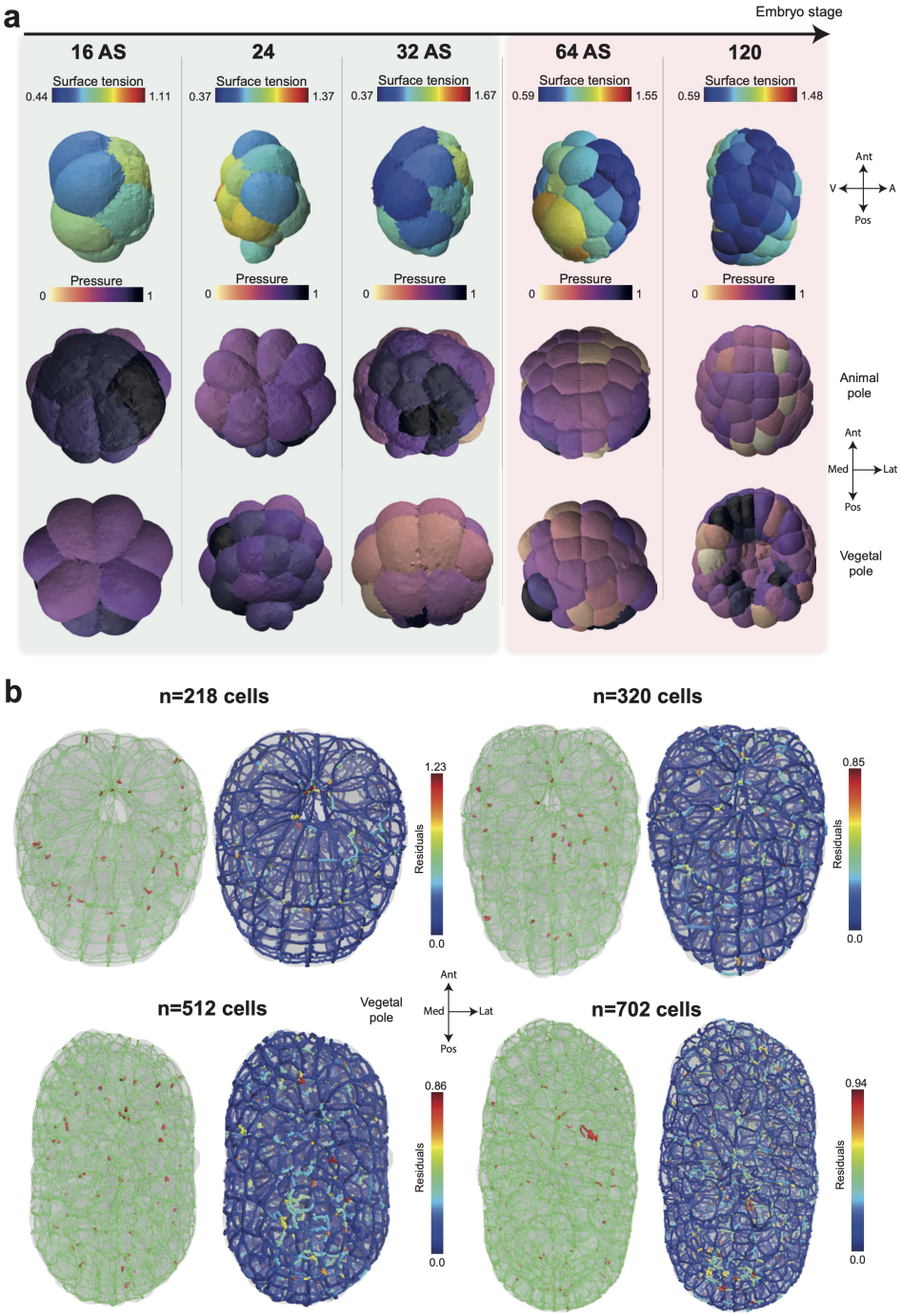
Variability in tensions inferred automatically with our pipeline

	Embryo 1	Embryo 2	Embryo 3	Embryo 4
$\text{std}(\gamma_{cm})/\text{mean}(\gamma_{cm})$	0.26	0.27	0.18	0.24
$\text{std}(\gamma_{cc})/\text{mean}(\gamma_{cc})$	0.10	0.05	0.08	0.07

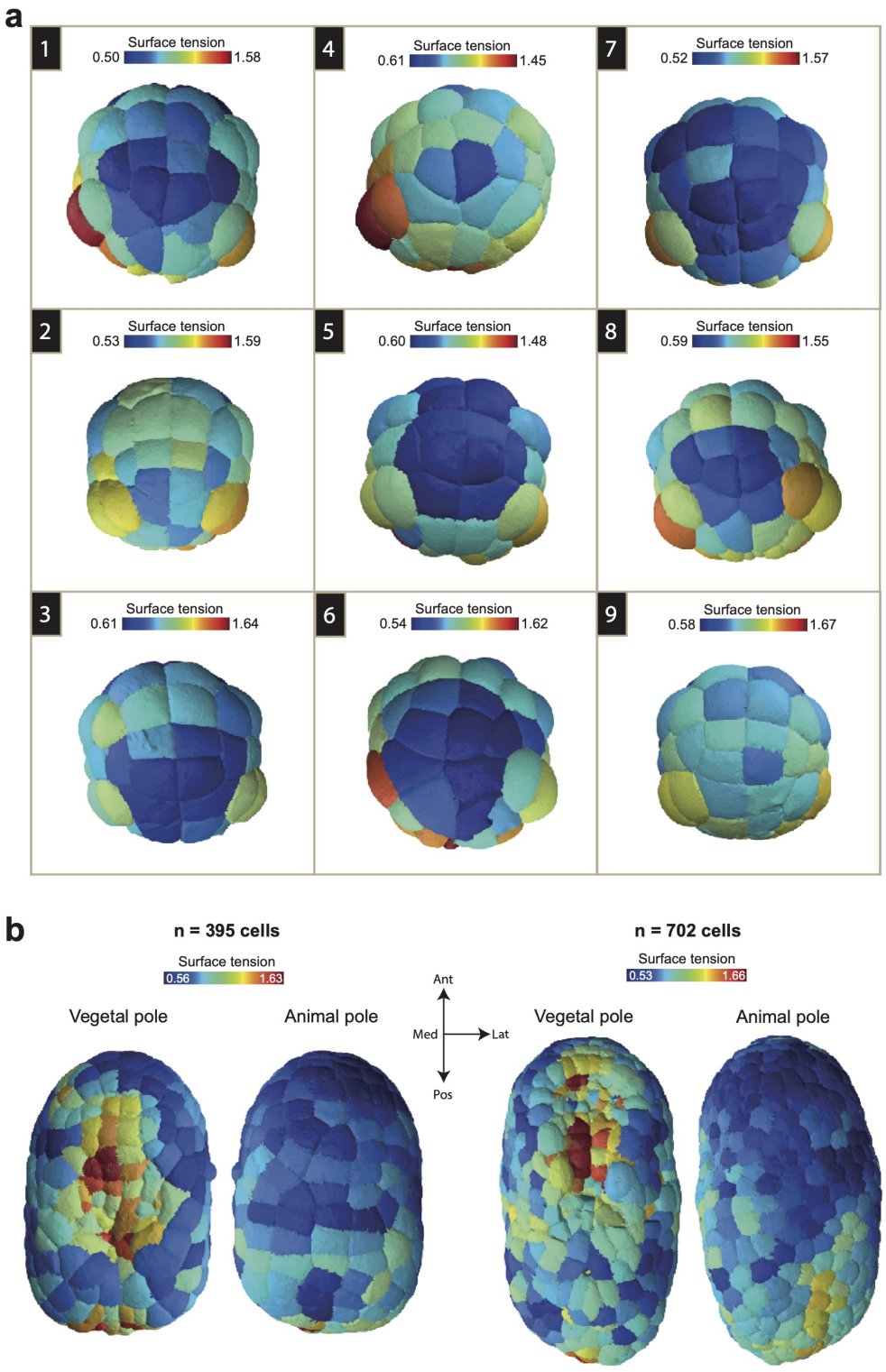
Variability in tensions inferred manually (data from ref. [5])

Time post-division (hh:mm)	0:00	3:00	6:00	9:00	12:00
$\text{std}(\gamma_{cm})/\text{mean}(\gamma_{cm})$	0.27	0.27	0.29	0.24	0.31
$\text{std}(\gamma_{cc})/\text{mean}(\gamma_{cc})$	0.25	0.26	0.48	0.41	0.43

Extended data Fig. 3: **Inference sensitivity and influence of tensions formulas for pressure inference.** **a)** Histogram of eigenvalues of the pseudo-inverse matrices used to infer tensions and pressures for the Young-Dupré, variational Young-Dupré, Laplace and Variational Laplace formulas, on our simulated embryo dataset. The spread of the histogram is a measure of the conditioning of the matrix. **b)** Comparison of the relative error on inferred pressures obtained on our simulated embryo dataset between Laplace and variational Laplace formulas. **c)** Mean relative error on angles reconstruction (left), tension inference (middle) and pressure inference (right), depending on the refinement of the mesh (in pixels) and the image size.



Extended data Fig. 4: **Additional validation data of the 3D tension inference. a)** Mitotic softening and stiffening in the 16AS, 24, 32AS, 64 and 120 cell stages of the early ascidian embryo (*P. mammillata*). Upper row: sagittal view of inferred apical tension. Middle and bottow rows: animal and vegetal views of the inferred cell pressures. The ratio of mitotic to interphase apical tension is colored green if it is less than 1 and red if it is greater than 1. The orientation of the embryo is given by arrows Ant: anterior, Pos: posterior, Med: medial, Lat: lateral, V: vegetal, A: animal. **b)** Vegetal view of stable (green) and unstable (red) junctions (Left) and tension inference residues (Right) in ascidian embryos (*P. mammillata*) at 218, 320, 512 and 702 cell stages. The orientation of the embryo is given by arrows Ant: anterior, Pos: posterior, Med: medial, Lat: lateral.



Extended data Fig. 5: **Additional tension maps of ascidian *P. mammillata* gastrula and neurula. a)** Nine examples of apical tension maps of 64 cell gastrula (animal pole). **b)** Maps of apical tensions at the animal and vegetal poles of early (Left) and late neurula ascidian embryos. The orientation of the embryo is given by arrows Ant: anterior, Pos: posterior, Med: medial, Lat: lateral.

Chapter 4

An Adjoint-State method for Tension Inference on Heterogeneous Foams

Written in collaboration with Hervé Turlier.

This work has just been achieved lately and has not been submitted yet.

This project began as my master's thesis project. The initial idea was to infer the surface tensions of the embryo from images using machine learning, more precisely graph neural networks. However, I rapidly realized that this was not a good approach. I privileged more classical methods to solve this inverse problem and wrote in my master's thesis the equations to solve the adjoint-state method for the doublet. It took me a lot of time to generalize it to any cell clusters, and it finally worked only at the very end of my Ph.D. Many things were crucial. First, the simulation engine was greatly enhanced by Nicolas Ecker, and it allows for faster and less error-prone simulations. Then, the state equations were essential. We took the variational Young-Dupré equations that came from our previous work. Eventually, a good SE(3)-invariant norm to compare meshes with each other was important to obtain a smooth gradient descent. This work is the ultimate frontier of tension inference in foams and provides a fitting conclusion to my efforts. This method is less straightforward than a simple mean-square minimization of Young-Dupré equations, but it has greater potential, as it can be generalized to other physical models more easily.

In this study, we propose a novel approach to solve the inverse problem of surface tension inference in heterogeneous foams, using an iterative gradient-based nonlinear optimization method. Our methodology is based on the utilization of the adjoint-state method to compute gradients of the final shape with respect to the initial surface tensions. We delve into the design and implementation principles of this method, detailing the development of the state equations, the cost function, and the optimization procedure. We study in depth their role in enabling efficient and accurate surface tension inference, and provide results from simple examples that underscore its potential in pushing forward the state of the art of tension inference. Eventually, we propose an interesting generalization to line tension inference, offering a way to measure biological quantities that remain difficult to measure or infer.

Chapitre 4

Une méthode d'état adjoint pour l'inférence de tension sur les mousses hétérogènes

Écrit en collaboration avec Hervé Turlier.

Ce travail vient tout juste d'être achevé récemment et n'a pas encore été soumis.

Ce projet a débuté comme mon projet de thèse de master. L'idée initiale était de déduire les tensions de surface de l'embryon à partir d'images en utilisant des réseaux de neurones sur graphes. Après avoir réalisé que ce n'était pas la bonne approche, j'ai privilégié des méthodes plus classiques pour résoudre ce problème inverse et j'ai écrit dans ma thèse de master les équations pour résoudre la méthode de l'état adjoint pour le doublet. Il m'a fallu beaucoup de temps pour la généraliser à n'importe quel amas de cellules, et cela n'a finalement fonctionné qu'à la toute fin de mon doctorat. Plusieurs éléments étaient cruciaux. D'abord, le moteur de simulation en 3D a été grandement amélioré par Nicolas Ecker, ce qui permet des simulations plus rapides et moins sujettes aux erreurs. Ensuite, les équations d'état étaient essentielles. Nous avons pris les équations variationnelles de Young-Dupré issues de nos travaux précédents. Finalement, une norme invariante par transformations du groupe Euclidien SE(3) pour comparer les maillages entre eux était importante pour regulariser la descente de gradient. Ce travail est le stade ultime de l'inférence de tension dans les mousses et fournit une conclusion adéquate à mes efforts. Cette méthode est moins directe qu'une simple minimisation des carrés moyens des équations de Young-Dupré, mais elle a un plus grand potentiel, car elle peut être généralisée plus facilement à d'autres modèles physiques.

Dans cette étude, nous proposons une nouvelle approche pour résoudre le problème inverse de l'inférence de tension de surface dans les mousses hétérogènes, en utilisant une méthode d'optimisation non linéaire itérative basée une descende de gradient. Notre méthodologie repose sur l'utilisation de la méthode de l'état adjoint pour calculer les gradients de la forme finale par rapport aux tensions de surface initiales. Nous approfondissons les principes de conception et de mise en œuvre de cette méthode, en détaillant le développement des équations d'état, de la fonction de coût et de la procédure d'optimisation. Nous étudions en profondeur leur rôle pour permettre une inférence de tension de surface efficace et précise, et fournissons des résultats à partir d'exemples simples qui soulignent son potentiel pour faire avancer l'état de l'art de l'inférence de tension. Finalement, nous proposons une généralisation intéressante à l'inférence de tension linéaire, offrant un moyen de mesurer des quantités biologiques qui restent difficiles à mesurer ou à déduire.

4.1 Introduction

The field of foam physics plays a crucial role in the understanding and modeling of various biological phenomena, specifically to study early embryos and epithelial tissues, in the absence of an extracellular matrix [1–4]. A key task within this area is tension inference, which aims at solving an inverse problem to reconstruct the forces that govern the shape of the tissue.

In this domain, a wealth of progress has been made through image-based inference techniques, both in 2D and 3D [1,4–6]. The typical strategy involves reconstructing the foam using explicit or dual representation, then inverting mechanical equilibrium equations to extract the surface tensions of the embryonic interfaces from geometry. Several sets of equilibrium equations exist to link physical parameters (such as surface tensions γ) with geometrical counterparts [1, 7]. Most of them correspond to variants of the expression of the force balance equilibrium at junctions, named *Young-Dupré* equations [8,9], that make intervene the angles between junctions. We recently proposed a new set of equations, called *Variational Young-Dupré* equations, that relies on vertices positions [4]. Despite theoretical equivalence, the final result obtained using these equations differ in practice, because of their sensitivity to the discrete representation of the geometry. Overall, all these formula depends on the accuracy of the numerical determination of the geometrical quantities that they rely on. Unfortunately, both angles and overall vertices positions are hard to determine from data, in spite of recent progresses [10,11].

In this study, we reframe the tension inference problem as an optimization procedure of the distance between a mesh resulting from foam simulation, and the target mesh derived from data. We utilize the adjoint-state method [12–14] for computing the gradient of a cost function with respect to input parameters, i.e surface tensions. The cost function's definition can be adapted to suit the accuracy of geometrical quantity reconstruction and does not require direct representation of equilibrium. Our proposed approach affords numerous advantages: by solving the forward problem, we ensure that the parameters are compatible with real foam shapes. We can also calculate the corresponding pressures and the stress tensor accurately. We delve into the inference problem in-depth, demonstrating how a cost function closely formulated to energy can aid in minimizing overall uncertainty in surface tension determination.

Contributions The key contributions of this paper are :

- We propose a novel formulation of state equations for heterogeneous foams compatible with the adjoint-state method.
- We demonstrate that gradient-based optimization surpasses the previous state of the art in tension inference, allowing for topology changes if necessary.
- We offer a simple implementation of the Adjoint Method leveraging automatic differentiation.
- We design a principled distance between foam meshes that is SE(3) invariant.

4.2 Adjoint-State Method for Tension Inference

The field of foam physics plays a crucial role in the understanding and modeling of various biological phenomena, specifically to study epithelial tissues, in the absence of an extracellular matrix. An heterogeneous foam is a foam with heterogeneous surface tensions γ_m at each of its

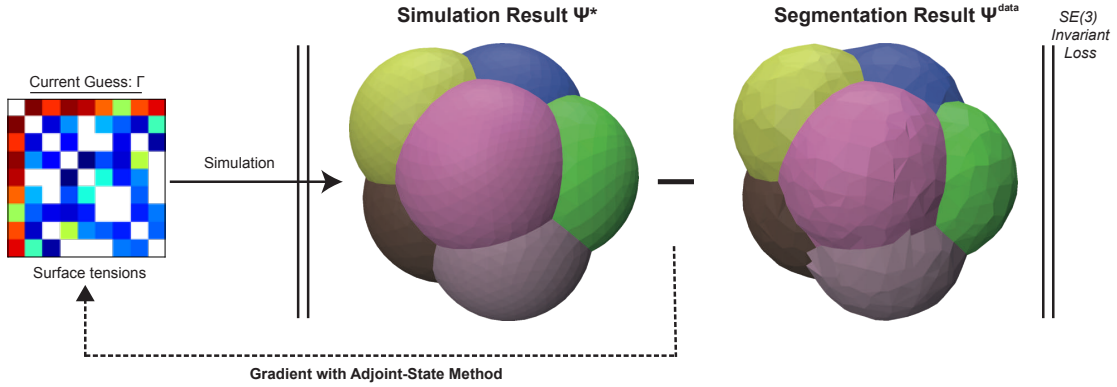


Figure 4.1: We retrieve the surface tensions Γ from the an input geometry Ψ^{data} with an iterative optimization, where at each step, we do a simulation from the current tensions Γ, and compare the simulation result Ψ^* to the input geometry Ψ^{data}, with a rigid-body motion invariant loss (SE(3) invariance). The gradient with respect to Γ is computed using the adjoint-state method.

interfaces, of area $\mathcal{A}_m$. It minimizes its total area energy, under the constraint of conservation of the volume $\mathcal{V}_k$ of every individual cells, and is described by the following Lagragian:

$$\mathcal{L} = \sum_{m=1}^{n_m} \gamma_m \mathcal{A}_m - \sum_{k=1}^{n_c} p_k(\mathcal{V}_k - \mathcal{V}_k^0), \tag{4.1}$$

Determining the geometry of such an heterogeneous foam requires to take as input the surface tensions of its interfaces, represented in a vector Γ,, the volumes of its cells V, an input description of its geometry Ψ, and to returns an output geometry $\Psi^*(\Gamma)$ than minimizes the surface energy, according to the Lagrangian. We call this operation **solving the forward problem**. The output geometry $\Psi^*(\Gamma)$ verifies the following state equations:

$$\mathcal{S}(\Psi^*(\Gamma), \Gamma) = 0 \tag{4.2}$$

The classical approach consists of reconstructing the geometry from data Ψ^{data}, and to use these state equations to recover the physical parameters:

$$\Gamma^{\text{data}} = \min_{\Gamma} \|(\mathcal{S}(\Psi^{\text{data}}, \Gamma)\|^2 \tag{4.3}$$

We consider another approach (Fig 4.1), which is to formulate a cost function that depends on the resulting geometry $C(\Gamma) = l(\Psi^*(\Gamma)) = D(\Psi^*(\Gamma), \Psi^{\text{data}})$, and find the set of parameters Γ^{data} that minimizes this cost function, using an iterative gradient-based optimization of $\Gamma : \Gamma \mapsto \Gamma - \eta \dfrac{\partial C}{\partial \Gamma}$. When the forward problem can be solved in a differentiable manner, the gradients can be obtained using automatic differentiation. However, in our case the forward problem is more complicated, as it requires itself an optimization, that includes non differentiable operations.

The adjoint methods aims at computing the gradient of the functional $C(\Gamma)$ in such difficult cases, using an intermediate variable Λ called the adjoint-state variable, by computing the adjoint state equation:

$$\textbf{Adjoint-State equation}: \boxed{\left(\frac{\partial \mathcal{S}}{\partial \Psi}\right)^T (\Psi^*, \Gamma) \times \Lambda^* = \left(\frac{\partial l}{\partial \Psi}(\Psi^*, \Gamma)\right)^T} \tag{4.4}$$

$$\textbf{Cost function gradient}: \frac{\mathrm{d}C}{\mathrm{d}\Gamma}(\Gamma) = -\Lambda^{*T}\left(\frac{\partial \mathcal{S}}{\partial \Gamma}\right)(\Psi^*, \Gamma) \tag{4.5}$$

We will use this gradient to up to perform tension inference on active foams. Solving an inverse problem through an iterative optimization procedure requires several key ingredients: A simulation engine to find the energy-minimizing geometry $\Psi^*(\Gamma)$, a set of state equations $\mathcal{S}$ that defines uniquely the equilibrium, and a good formulation of the distance D that allow for a smooth optimization.

4.3 Mechanical Equilibrium of an Heterogeneous Foam

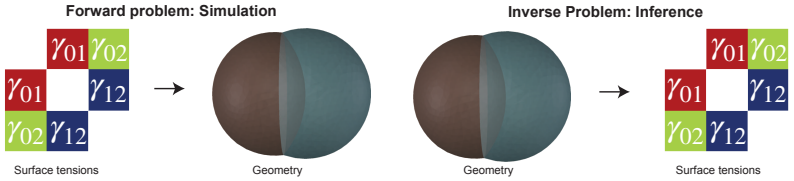


Figure 4.2: The simulation is the forward problem, and the inference is the inverse problem.

The mechanical equilibrium of an active foam can be characterized using the previously introduced Lagrangian:

$$\vec{0} = \frac{\partial \mathcal{L}}{\partial \vec{x}_\alpha} = \sum_{m=1}^{n_m} \gamma_m \frac{\partial \mathcal{A}_m}{\partial \vec{x}_\alpha} - \sum_{k>1}^{n_c} p_k \frac{\partial \mathcal{V}_k}{\partial \vec{x}_\alpha} \tag{4.6}$$

We solve the forward problem (Fig 4.2) by doing a gradient-based optimization where at each step, we compute the gradient of the surface energy, project the displacements to make the total volume variation orthogonal to the vertex displacements [15]. When converged, we end up with a surface that minimizes the total area energy under a constraint of conservation of volumes. We implemented it in Multitracker, [16], a solver that takes into account T1 and T2 topology transitions of foam dynamics [17], and manages collision detection and various mesh refinement rountines.

Foam State Equations If we consider the initial mechanical balance $\sum_{m=1}^{n_m} \gamma_m \frac{\partial \mathcal{A}_m}{\partial \vec{x}_\alpha} - \sum_{k>1}^{n_c} p_k \frac{\partial \mathcal{V}_k}{\partial \vec{x}_\alpha} = 0$, this expression makes the pressures intervene, that are themselves a function of the surface tension and the geometry of the foam. We will eliminate the pressures from the force-balance, to obtain the set of state equations $\mathcal{S}(\{\vec{x}_\alpha(\Gamma)\}, \Gamma)$ that depends only on the surface tension.

To do that, we first rewrite the equilibrium as a product between a huge tensor $Q \in \mathbb{R}^{n_v \times (n_m + n_c) \times 3}$ that contains the derivatives of the areas and volumes with respect to each vertex, and the matrix $\Pi \in \mathbb{R}^{n_m + n_c}$ that contains the surface tensions and pressures. The tensor Q can be conveniently expressed as a matrix of $\mathbb{R}^3$ vectors:

$$Q \times \Pi = \begin{pmatrix} \nabla_{\vec{x_1}} \mathcal{A}_1 & \dots & \nabla_{\vec{x_1}} \mathcal{A}_{n_m} & \nabla_{\vec{x_1}} \mathcal{V}_1 & \dots & \nabla_{\vec{x_1}} \mathcal{V}_{n_c} \\ \cdot & & \cdot & \cdot & & \cdot \\ \cdot & & \cdot & \cdot & & \cdot \\ \cdot & & \cdot & \cdot & & \cdot \\ \cdot & & \cdot & \cdot & & \cdot \\ \nabla_{\vec{x_{n_v}}} \mathcal{A}_1 & \dots & \nabla_{\vec{x_{n_v}}} \mathcal{A}_{n_m} & \nabla_{\vec{x_{n_v}}} \mathcal{V}_1 & \dots & \nabla_{\vec{x_{n_v}}} \mathcal{V}_{n_c} \end{pmatrix} \begin{pmatrix} \gamma_1 \\ \dots \\ \gamma_{n_m} \\ -p_1 \\ \dots \\ -p_{n_c} \end{pmatrix} = 0 \tag{4.7}$$

Then we define a matrix $R = Q^T Q$. The equality $R \times \Gamma = 0$, leads to two sets of equations:

$$\forall l \in [\![1 \dots n_m]\!], \quad 0 = \sum_{m=1}^{n_m} \gamma_m \left(\sum_{\vec{x}_\alpha} \frac{\partial \mathcal{A}_m}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{A}_l}{\partial \vec{x}_\alpha} \right) - \sum_{k=1}^{n_c} p_n \left(\sum_{\vec{x}_\alpha} \frac{\partial \mathcal{V}_k}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{A}_l}{\partial \vec{x}_\alpha} \right) \tag{4.8}$$

$$\forall n \in [\![1 \dots n_c]\!], \quad 0 = \sum_{m=1}^{n_m} \gamma_m \left(\sum_{\vec{x}_\alpha} \frac{\partial \mathcal{A}_m}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{V}_n}{\partial \vec{x}_\alpha} \right) - \sum_{k=1}^{n_c} p_n \left(\sum_{\vec{x}_\alpha} \frac{\partial \mathcal{V}_k}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{V}_n}{\partial \vec{x}_\alpha} \right) \tag{4.9}$$

Matrix Variational Equations We define four matrices, $\mathrm{G}_\Gamma^{lm} = \sum_{\vec{x}_\alpha} \frac{\partial \mathcal{A}_m}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{A}_l}{\partial \vec{x}_\alpha}, \mathrm{G}_P^{lk} = \sum_{\vec{x}_\alpha} \frac{\partial \mathcal{V}_k}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{A}_l}{\partial \vec{x}_\alpha}, \mathrm{B}_\Gamma^{nm} = \sum_{\vec{x}_\alpha} \frac{\partial \mathcal{A}_m}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{V}_n}{\partial \vec{x}_\alpha}, \mathrm{B}_P^{nk} = \sum_{\vec{x}_\alpha} \frac{\partial \mathcal{V}_k}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{V}_n}{\partial \vec{x}_\alpha}$, and uses them to reexpress our equations using the following block matrix:

$$\begin{pmatrix} \mathrm{G}_\Gamma & -\mathrm{G}_P \\ \mathrm{B}_\Gamma & -\mathrm{B}_P \end{pmatrix} \times \begin{pmatrix} \Gamma \\ \mathrm{P} \end{pmatrix} = 0, \tag{4.10}$$

$\mathrm{G}_{\Gamma,P}$ are matrices of sizes respectively n_m^2 and $n_m \times n_c$ and $B_{\Gamma,P}$ rectangular matrices of sizes respectively $n_c \times n_m$ and n_c^2. The linear system to solve for pressures at given tensions is called *Variational Laplace*:

$$\mathrm{B}_P \times \mathrm{P} = \mathrm{B}_\Gamma \times \Gamma \tag{4.11}$$

This equation will allow to infer the pressures with a very good precision, once surface tensions are infered. In the appendix, we demonstrate that the matrix G_P is invertible. We can thus write a closed-form linear system for the tensions alone, that constitutes our state equations, that we call the Variational Young-Dupré Equations:

$$\boxed{\mathcal{S}(\{\vec{x}_\alpha(\Gamma)\}, \Gamma) = \left(\mathrm{G}_\Gamma - \mathrm{G}_P \mathrm{B}_P^{-1} \mathrm{B}_\Gamma \right) \times \Gamma = 0} \tag{4.12}$$

4.4 Implementation of the adjoint method

The implementation of an optimization procedure using the adjoint-state method can be greatly simplified by using the derivatives provided by automatic differentiation engines used in deep-learning [18, 19]. This way to implement adjoint-state method has already been done previously [20], and we believe that it constitutes an underestimated approach to compute gradients that would be impossible to access otherwise.

4.4.1 Gradient-based optimization implementation

First Guess The first guess can be an homogeneous foam, with a surface tension $\gamma = 1$ everywhere. A better first guess can be obtained by doing tension inference using a 3D tension inference pipeline that we previously designed [4]. The algorithm takes as an input an instance voxel segmentation, that can be easily obtained using convolutional neural networks (i.e each voxel is assigned a label according to which cell it belongs to). It converts this segmentation into a multimaterial mesh, uses this mesh to measure angles, and uses these

angle measurements to invert Young-Dupré equations to infer surface tension. These surface tensions constitues the first guess of our algorithm, that will be optimized through the iteration procedure.

4.4.2 Energy landscape and Objective function design

The choice of objective that is minimized can be made using several criterias. It has to be convex, i.e converge to the real shape, and nondegenerate: If several physical parameters lead to a very little norm, in practice it will be hard to determine which parameters are concerned, leading to an uncertainty in the determination of the tension. The distance between two meshes will consist of the difference of measured geometrical quantities between the current simulated mesh $\Psi = \{x_1, x_2, ..., x_n\}$, and the target mesh $\Psi^{data} = \{y_1, y_2, ..., y_m\}$, with n and m vertices respectively. The most immediate way to compare two meshes is to take a set of points sampled from the surface of these meshes, and compute a distance based on these points.

Point-based metrics and Registration The Hausdorff loss [21,22] (also known as Chamfer loss) is a distance metric often used for comparing point sets or triangle meshes, particularly in the field of 3D computer vision [23]. The Hausdorff distance is defined as follows:

$$D_{CD}(\Psi, \Psi^{\text{data}}) = \frac{1}{n} \sum_{x \in X} \min_{y \in Y} ||x - y||_2^2 + \frac{1}{m} \sum_{y \in Y} \min_{x \in X} ||x - y||_2^2 \tag{4.13}$$

This is an asymmetric bidirectional distance measure where for each point in the first set, it finds the closest point in the second set and vice versa. The two sums are then averaged to produce the final distance measure. However, a key issue in comparing meshes is the lack of Special Euclidean Group SE(3) invariance (consisting of 3D rotations and translations, Fig 4.3) [24, 25], which means that the position and orientation of the meshes can affect the Chamfer loss. Hence, to accurately compare the meshes, it is necessary to align (register) them with respect to each other.

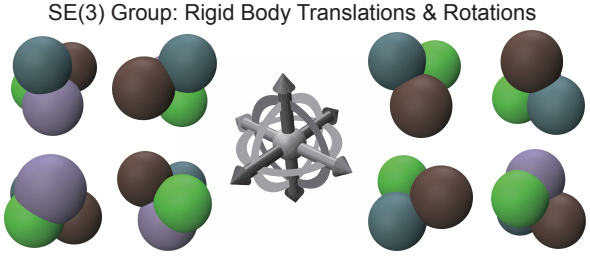


Figure 4.3: The Lie SE(3) Group is composed of rigid-body translations and rotations.

The registration process seeks to find the best SE(3) transformation that minimizes the distance between the point sets. For instance, the registration can be modeled as an optimization problem:

$$T^* = \arg\min_T D_{CD}(T \cdot \Psi, \Psi^{\text{data}}) \tag{4.14}$$

where T is the transformation matrix and T^* is the optimal transformation. The dot denotes application of the transformation to all points in the set Ψ. Registration is a non-trivial problem due to its high computational complexity, especially for large triangle meshes, and

is often be the bottleneck in comparing triangle meshes. We will avoid this cumbersome procedure by constructing a SE(3) invariant distance.

SE(3) Invariant Losses A good way to construct a SE(3) invariant distance is to measure geometrical quantities extracted from the two meshes Ψ, Ψ^{data}. It can also consist of the areas of the cells, the angles between junctions, the length of the junctions. The junctions are a feature that is very hard to reconstruct from data, so comparing their length will lead to poor results. The angles can be reconstructed fairly well, but the simulation starts from Young-Dupré relationship, i.e from the measurement of the angles, thus optimizing the distance between junctional angles leads to a kind of circular reasoning. As areas are well reconstructed, a loss based on the difference between the area of regions can lead to good results. For any type of loss, it is important to be careful about the behavior of this distance close to the equilibrium. We will show that a loss based on the comparison between areas reproduces the behavior of our model, and thus his landscape share similar properties.

The energy of our model is $E = \sum_{m=1}^{n_m} \gamma_m A_m$. We can write it using vectorial notations $E(\Gamma, A) = \Gamma^T \cdot A$. E is optimized non linearly under the constraint that the volumes of each cell is conserved. We represent this constraint using a function $\mathcal{G}(V) = 0$ where V is a vector containing the volumes of every cell. Both A and V depends on the final geometry of the foam, that we call Ψ, and that we represent explicitely with a triangle mesh. Ψ is a minimizer of the constrained equation:

$$\Psi^* = \underset{\mathcal{G}(V(\Psi))=0}{\operatorname{argmin}} \Gamma^T \cdot A(\Psi) \tag{4.15}$$

If the energy landscape is very flat around a local minimum, it means that $\frac{\partial E}{\partial \Psi} \approx 0$ for $\Psi \approx \Psi^*$. If so, there is going to be many geometrical configurations, potentially largely different, that will lead to to very close total energies. In this case, optimizing the system through the forward simulation will be harder.

More importantly, the real system always has an internal level of noise σ that can be due to perturbations of various nature (thermal, chemical, mechanical). If the real system does indeed follow our physical model and have a very flat energy landscape, the shape observed Ψ^{data} can deviate significantly from the energy-minimizing steady-state shape Ψ^*, leading to a fundamental uncertainty about the forces governing shapes, independently from the precision of the measurements or inference.

A good design principle can be to choose a loss function $D(\Psi, \Psi^{\text{data}})$ such that if $D(\Psi, \Psi^{\text{data}}) \approx 0$, then the behavior of the loss landscape of the function is the same as the one of the energy. If so, it will not add more sources of uncertainty than those already present. If the landscape is flat, then there is an uncertainty on the real physical system, with the presence of so-called soft modes where changing configurations is very easy, and then tension inference is doomed from the beginning. If not, then it will no be flat neither for the distance D, so inference will be possible with a good precision.

We write $\sigma = |E(\Psi^{\text{data}}) - E(\Psi^*)|$, the level of noise in the system. The configuration observed Ψ^{data} will not correspond exactly to the configuration of minimal energy Ψ^*, because of the observation noise sigma. We note $E' = \frac{\partial E}{\partial \Psi}$ and thus have:

$$E(\Psi^{\text{data}}) = E(\Psi^*) + E'(\Psi^{\text{data}} - \Psi^*) \Rightarrow \Psi^{\text{data}} = \Psi^* + \frac{\sigma}{E'} \tag{4.16}$$

Then, if we stop the optimization when the distance reaches a certain error level $D(\Psi, \Psi^{\text{data}}) \approx \epsilon$, we can write

$$D(\Psi, \Psi^{\text{data}}) = 0 + D'(\Psi - \Psi^{\text{data}}) \Rightarrow \Psi = \Psi^{\text{data}} + \frac{\epsilon}{D'} \tag{4.17}$$

Combining these two results, we thus end up with a relation between the recovered geometry Ψ and the optimal one Ψ^*:

$$\Psi = \Psi^* + \frac{\sigma}{E'} + \frac{\epsilon}{D'} \tag{4.18}$$

Thus, to avoid having errors with different behaviours, a good design choice would be to choose D such that $\frac{\partial E}{\partial \Psi}$ and $\frac{\partial D}{\partial \Psi}$ have a similar form. We choose

$$D_{IA}(\Psi, \Psi^{\text{data}}) = |A(\Psi) - A(\Psi^{\text{data}})| \tag{4.19}$$

Which has a derivative $D'_{IA}(\Psi, \Psi^{\text{data}}) = \frac{\partial A}{\partial \Psi}$ that is very close from the derivative of the energy $E' = \Gamma^T \cdot \frac{\partial A}{\partial \Psi}$, and will thus have a similar behaviour during optimization.

Computation of the derivatives We implement the functions $\mathcal{S}$ and C in a fully differentiable manner using Pytorch [18]. The partial derivatives involved in the adjoint-method, $\frac{\partial \mathcal{S}}{\partial \Psi}, \frac{\partial \mathcal{S}}{\partial \Gamma}$ and $\frac{\partial l}{\partial \Gamma}$ can then be obtained with a simple call to the function *torch.autograd.functional.jacobian*. Implementing our functions in pytorch also allow for a straightforward GPU parallelization, which can result in significant benefits in speed when dealing with large meshes.

Parameter update rule We need to choose an update rule of the surface tensions $\{\gamma_m\}_{m=1}^{n_m}$, based on the computation of the gradients. We use Adam [26], that is implemented directly in Pytorch. A simple gradient descent would work, but the adaptative step of Adam is more convenient close to the solution, and does not need a taylored schedule of reduction of the learning rate.

4.5 Applications

4.5.1 Test on a foam simulation

The metric that we will use to evaluate the tension inference will be the mean relative tension error $\epsilon = \frac{1}{n_m} \sum_{m=1}^{n_m} \frac{|\gamma_m^{guess} - \gamma_m^{gt}|}{\gamma_m^{gt}}$. In a previous study [4], we designed a benchmark to assess the results of our inference pipeline. We were able to obtain a mean relative tension error of 10.1% over the example of the benchmark. The goal is our method is to start from our tension inference, and improve the results using our slow but principled optimization procedure. Due to the length of the adjoint method, we did not perform this analysis over all the elements of the benchmark, and just took one example to illustrate the idea. The initial mesh has $n_v = 1570$ vertices and $n_f = 3270$ faces. We start from the infered tensions using our previous method, which had a mean relative tension error of $\epsilon = 11.9\%$. We did 400 iterations, during 8 hours 29 minutes, with a mean duration of 76.46s per iteration. We end up with an error of 6.5%.

This shows that our method is indeed capable of reducing substantially the errors of tension inference, compared to our previous method, which was already quite precise. However, the method is quite long, and its use over our previous method, which worked almost on real time, is only justified on cases where a high amount of precision is key. We believe that for tension inference, our previous method will remain a good standard.

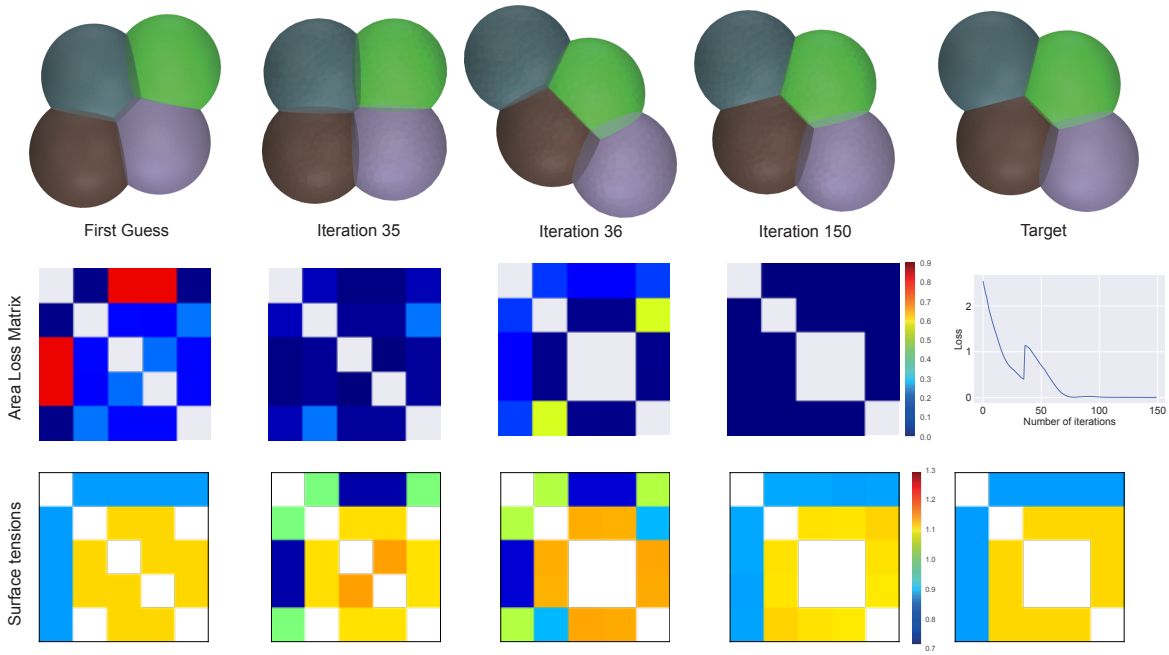


Figure 4.4: Our distance D_{IA} allow us to find back the original shapes when T1 topology transitions are needed. Here the T1 transition is performed during the iteration 36, corresponding to the bump in the optimization of D_{IA}.

4.5.2 Quadruplet with T1 transition

Now we will demonstrate the polyvalence of our loss D_{IA} by showing that we can recover the right tensions even when starting from the wrong topology (Fig 4.4). We want to recover a uniform target foam, starting from a uniform foam. However, the initial configuration of the cells is such that a T1 transition is needed. The initial mesh has $n_v = 1640$ vertices and $n_f = 3360$ faces. We do 150 iterations, with a mean duration of 39.2s per iterations, for a total length of optimization of 1h and 39 minutes.
We see that a topology change happens at the iteration 35, leading to the right cell-cell contacts. The optimization is able to change the surface tensions in order to do this T1 transition, and then find the right surface tensions by matching the interfacial areas of the target shape. We end up with a relative error on the tensions $\epsilon = \frac{1}{n_m}\sum_{m=1}^{n_m}\frac{|\gamma_m^{guess}-\gamma_m^{gt}|}{\gamma_m^{gt}} = 2.51\%$.

4.6 Conclusion

This paper presents a novel approach to surface tension inference through the application of the adjoint method. While our methodology may appear complex in comparison to existing tension inference techniques, it offers a remarkable accuracy and a greater potential of generalization to other physical models than previous methods. We noticed that we could take advantage of automatic differentiation engine to make implementation simpler, and GPU parallelization straightforward. Eventually, we propose a principled SE(3) invariant loss that allow to recover shapes even when topology changes are needed.

Implementing the adjoint method for surface tension inference requires a higher amount of computation, the optimization procedure being less direct than other force inference methods [4, 5]. Nevertheless, this approach ensures an accurate match between the simulation

and the biologically observed shape - a guarantee that direct force inference cannot provide. By solving the forward problem, we ensure that the inferred parameters will result in foam shapes that closely mirror the observed biological forms when utilized in a simulation. In this direction, the application of the adjoint method to line tensions (see Appendix B) represents a crucial breakthrough. For the first time, it will allow for the inference of line tensions in embryos, with two promising candidates for such analysis: the 16/24 cell stage ascidians and the 16 cell stage mouse embryos. It is worth noting, however, that the mouse embryos must be removed from the zona pellucida to prevent potential inaccuracies in force measurements due to the impact of this surrounding layer on their surface areas.

Despite the adjoint method's complexity, its unique advantages of accuracy and potential to measure line tensions creates exciting possibilities. Now, with this powerful tool at their disposal, it is up to the experimentalists to utilize it, pushing forward our understanding of the complex mechanics that drive embryogenesis.

Bibliography

[1] Chloé Roffay, Chii J Chan, Boris Guirao, Takashi Hiiragi, and François Graner. Inferring cell junction tension and pressure from cell geometry. *Development*, 148(18):dev192773, 2021.

[2] Jean-Léon Maître, Ritsuya Niwayama, Hervé Turlier, François Nédélec, and Takashi Hiiragi. Pulsatile cell-autonomous contractility drives compaction in the mouse embryo. *Nature cell biology*, 17(7):849–855, 2015.

[3] Jean-Léon Maître, Hervé Turlier, Rukshala Illukkumbura, Björn Eismann, Ritsuya Niwayama, François Nédélec, and Takashi Hiiragi. Asymmetric division of contractile domains couples cell positioning and fate specification. *Nature*, 536(7616):344–348, 2016.

[4] Sacha Ichbiah, Fabrice Delbary, Alex Mcdougall, Rémi Dumollard, and Hervé Turlier. Embryo mechanics cartography: inference of 3d force atlases from fluorescence microscopy. *BioRxiv*, pages 2023–04, 2023.

[5] Weiyuan Kong, Olivier Loison, Pruthvi Chavadimane Shivakumar, Eunice HoYee Chan, Mehdi Saadaoui, Claudio Collinet, Pierre-François Lenne, and Raphaël Clément. Experimental validation of force inference in epithelia from cell to tissue scale. *Scientific reports*, 9(1):1–12, 2019.

[6] Nicholas Noll, Sebastian J Streichan, and Boris I Shraiman. Variational method for image-based inference of internal stress in epithelial tissues. *Physical Review X*, 10(1):011072, 2020.

[7] Nicolas Harmand. *Pertinence et limites des tensions de surface et de ligne pour rendre compte des formes des cellules épithéliales*. PhD thesis, Université Paris Cité, 2019.

[8] Thomas Young. Iii. an essay on the cohesion of fluids. *Philosophical transactions of the royal society of London*, (95):65–87, 1805.

[9] Athanase Dupré and Paul Dupré. *Théorie mécanique de la chaleur*. Gauthier-Villars, 1869.

[10] Sacha Ichbiah, Fabrice Delbary, and Hervé Turlier. Differentiable rendering for 3d fluorescence microscopy. *arXiv preprint arXiv:2303.10440*, 2023.

[11] Matthew B Smith, Hugh Sparks, Jorge Almagro, Agathe Chaigne, Axel Behrens, Chris Dunsby, and Guillaume Salbreux. Active mesh and neural network pipeline for cell aggregate segmentation. *Biophysical Journal*, 122(9):1586–1599, 2023.

[12] Anna M Michalak and Peter K Kitanidis. Estimation of historical groundwater contaminant distribution using the adjoint state method applied to geostatistical inverse modeling. *Water resources research*, 40(8), 2004.

[13] Shingyu Leung and Jianliang Qian. An adjoint state method for three-dimensional transmission traveltime tomography using first-arrivals. 2006.

[14] R-E Plessix. A review of the adjoint-state method for computing the gradient of a functional with geophysical applications. *Geophysical Journal International*, 167(2):495–503, 2006.

[15] Kenneth A Brakke. The surface evolver. *Experimental mathematics*, 1(2):141–165, 1992.

[16] Fang Da, Christopher Batty, and Eitan Grinspun. Multimaterial mesh-based surface tracking. *ACM Trans. Graph.*, 33(4):112–1, 2014.

[17] Denis L Weaire and Stefan Hutzler. *The physics of foams*. Oxford University Press, 2001.

[18] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[19] Roy Frostig, Matthew Johnson, and Chris Leary. Compiling machine learning programs via high-level tracing. 2018.

[20] Antoine McNamara, Adrien Treuille, Zoran Popović, and Jos Stam. Fluid control using the adjoint method. *ACM Transactions On Graphics (TOG)*, 23(3):449–456, 2004.

[21] Herbert Federer. *Geometric measure theory*. Springer, 2014.

[22] Frank Morgan. *Geometric measure theory: a beginner's guide*. Academic press, 2016.

[23] Nikhila Ravi, Jeremy Reizenstein, David Novotný, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. Accelerating 3d deep learning with pytorch3d. *CoRR*, abs/2007.08501, 2020.

[24] William P HG Thurston. *Three-Dimensional Geometry and Topology, Volume 1: Volume 1*. Princeton university press, 1997.

[25] Judith Cederberg. *A course in modern geometries*. Springer Science & Business Media, 2004.

[26] DP Kingma. Adam: a method for stochastic optimization. In *Int Conf Learn Represent*, 2014.

Chapter 5

An artificial microscope to merge models and data

Written in collaboration with Fabrice Delbary, Matthieu Perez and Hervé Turlier.

This work resulted in a publication available on arRxiv. It is currently extended with the input of other authors before submission to a journal.

This project started a long time before I began my Ph.D. Fabrice Delbary was developing a segmentation algorithm based on a Sobolev norm, for which he developed a mathematical formulation of the Fourier transform of a mesh. An unexpected application of his work was the possibility to generate artificial microscopy data from meshes. I immediately understood the potential of this project to train neural networks, as this was what people were doing in other industries, such as autonomous driving or robotics. There were many aspects that were required to apply that at scale. First, the algorithm was quite slow, so I worked on the parallelization on GPU using CUDA, which allowed raising the speed by a factor of x1000. Then, an important thing was to work on the photorealism of the data, which we achieved by using realistic physical models of PSFs, and neural style transfer. Eventually, a user-friendly interface was quite important, and was developed by Matthieu Perez. Another unexpected application of this work is to use this image renderer to perform differentiable rendering, i.e., to fit the geometry and the PSF parameters from real data. In order to do that, I had to parallelize the computation of the gradient of the Fourier transform on GPU, and to develop other compute-efficient strategies to lower the computation time of each iteration as much as possible. This project has thus been developed over at least 5 years and necessitated a lot of work from 3 people. However, it will allow significant progresses in the field of 3D microscopy image analysis.

Fluorescence microscopy has become a pivotal technique for imaging biological objects, but extracting relevant quantitative information from the resulting 3D fluorescence images remains a complex task. To be able to compare numerical simulations and real microscopy images, we created alphaMic, an artificial microscope. This artificial image creator provides a principled methodology to generate synthetic images, enabling the generation of large, high-quality datasets, critical for validating computer vision algorithms. Our renderer simulates the 3D image formation process of fluorescence microscopy by convolving a 3D mesh, representing the geometry of the biological structure, with a parameterized point spread function (PSF). The framework uses the Fourier transform (FT) of the 3D

mesh for efficient computation, while overcoming challenges related to memory efficiency and numerical precision by adopting a fully differentiable GPU implementation. We further explore the impact of various PSF models on image quality and super-resolution methods. Eventually, we develop a differentiable version of our renderer, called deltaMic. Differentiable rendering permits a mesh-based reconstruction of complex shapes from artificial and real confocal microscopy images, a paradigm shift in the analysis of 3D fluorescence microscopy data.

Chapitre 5

Un microscope artificiel pour fusionner modèles et données

Écrit en collaboration avec Fabrice Delbary, Matthieu Perez et Hervé Turlier.

Ce travail a abouti à une publication disponible sur arXiv. Il est actuellement enrichi par les contributions d'autres auteurs avant soumission à une revue.

Ce projet a commencé bien avant que je ne commence mon doctorat. Fabrice Delbary développait un algorithme de segmentation basé sur une norme de Sobolev, pour lequel il avait élaboré une formulation mathématique de la transformée de Fourier d'un maillage. Une application inattendue de son travail était la possibilité de générer des données de microscopie artificielle à partir de maillages provenant de simulations. J'ai immédiatement compris le potentiel de ce projet pour entraîner des réseaux neuronaux, par analogie avec ce qui se faisait dans d'autres industries, telles que la conduite autonome ou la robotique. De nombreux aspects étaient nécessaires pour appliquer cela à grande échelle. Tout d'abord, l'algorithme était assez lent, donc j'ai travaillé sur la parallélisation sur GPU en utilisant CUDA, ce qui a permis d'augmenter la vitesse d'un facteur de 1000. Ensuite, un élément important était de travailler sur le photoréalisme des données, ce que nous avons réalisé en utilisant des modèles physiques réalistes de PSF, et des algorithmes de transfert de style. Finalement, une interface simple d'utilisation était assez importante et a été développée par Matthieu Perez. Une autre application inattendue de ce travail est d'utiliser ce générateur d'images pour effectuer un rendu différentiable, c'est-à-dire pour ajuster la géométrie et les paramètres du PSF à partir de données réelles. Pour ce faire, j'ai dû paralléliser le calcul du gradient de la transformée de Fourier sur GPU, et développer d'autres stratégies pour réduire autant que possible le temps de calcul de chaque itération. Ce projet a donc été développé sur au moins 5 ans et a nécessité beaucoup de travail de la part de 3 personnes. Cependant, il permettra des progrès significatifs dans le domaine de l'analyse d'images de microscopie 3D.

La microscopie à fluorescence est devenue une technique essentielle pour l'imagerie d'objets biologiques, mais l'extraction d'informations quantitatives pertinentes à partir des images de fluorescence 3D qui en résultent reste une tâche complexe. Pour pouvoir comparer les simulations numériques et les images de microscopie réelles, nous avons créé alphaMic, un microscope artificiel. Ce créateur d'images artificielles fournit une méthodologie fondée sur des principes physiques pour générer des images synthétiques, permettant la génération de jeux de données volumineux et de haute qualité, essentiels pour valider les algorithmes de vision par ordinateur. Notre moteur simule le processus de formation d'image 3D en microscopie à fluorescence en convoluant un maillage 3D, représentant la géométrie de la structure biologique, avec une réponse impulsionnelle spatiale (PSF) paramétrisée. Nous utilisons la transformée de Fourier (FT) du maillage 3D pour un calcul efficace, tout en surmontant les défis liés à l'efficacité de la mémoire et à la précision numérique en adoptant une mise en œuvre GPU entièrement différentiable. Nous explorons en outre l'impact de divers modèles de PSF sur la qualité de l'image et les méthodes de super-résolution. Finalement,

nous développons une version différentiable de notre moteur, appelée deltaMic. Le rendu différentiable permet une reconstruction basée sur un maillage de formes complexes à partir d'images de microscopie confocale artificielles et réelles, ouvrant des possibilités nouvelles pour l'analyse des données de microscopie à fluorescence 3D.

5.1 Introduction

Fluorescence microscopy [1] has become the most prevalent technique for imaging biological objects. In this method, biological samples are made visible by attaching a fluorescent dye, or fluorophore, to the structure of interest. A laser excites the dye, which subsequently emits fluorescent light that travels through the microscope's optics and is detected by a photosensitive sensor to produce a 2-dimensional image. In confocal microscopy, the focal plane is adjusted to generate a 3D volumetric image composed of optical sections of the biological sample.

Despite the widespread use of fluorescence microscopy in biology, extracting relevant quantitative information from 3D fluorescence images remains a significant challenge, prompting the development of new methods. In recent years, researchers have drawn inspiration from the field of computer vision, building on the success of 2D convolutional neural networks (CNN) and adapting their architectures to the peculiarities of biological images [2, 3]. These deep learning (DL)-based tools for fluorescence imaging [4] have enabled the performance and automation of various image analysis tasks, such as image restoration [5], instance segmentation [6, 7], and feature encoding with self-supervised learning [8].

To address the challenge of extracting relevant quantitative information from 3D fluorescence images, we have developed an artificial image creator that offers a principled approach to generating artificial images and proposes routines for generating corresponding annotations. The benefits of a principled method for creating artificial images are numerous, as it enables the generation of large, high-quality datasets with a wide variety of biological structures and imaging conditions, which are essential for validating computer vision algorithms with ground-truth data. Furthermore, in the context of deep learning, generating datasets of artificial images with corresponding labels simplifies the training process by eliminating the need for manual annotation, which can be time-consuming and error-prone. This allows researchers to focus on developing and refining deep learning algorithms tailored to specific biological applications, rather than expending resources on data collection and annotation.

The capability to generate artificial images paves the way for a more comprehensive understanding of the impact of PSFs on image quality and the effectiveness of super-resolution methods. We review various point spread function (PSF) models, generate artificial images using these PSF models, and directly compare and evaluate their impact on the generated images. This comparative approach not only illuminates the strengths and limitations of each PSF model but also identifies areas for potential improvement.

Our renderer is built on an efficient and flexible framework that approximates the image formation process by taking a mesh and a parameterized Point Spread Function (PSF) as inputs and generating a volumetric image from them. To create this 3D image, the mesh describing the geometry is convolved with the PSF through an element-wise product in the Fourier domain. We employ distribution theory to express the Fourier Transform (FT) of a 2D shape embedded in 3D [9,10] and derive the expression of the FT and its gradient for a triangle mesh.

We also demonstrate the application of our renderer for the inverse procedure: Differentiable rendering, wherein we use our image formation model to learn both the geometry and the PSF from the images.

Main contributions:

- We propose a simplified model of the fluorescence microscopy imaging process, com-

bining a mesh description of object geometry and a simple parameterized point spread function.

- We introduce a method to generate photorealistic artificial datasets of early embryos.
- We utilize our renderer to illustrate the effects of different PSFs, following classical models.
- We present a differentiable formulation of the Fourier transform of a triangle surface mesh, along with its highly-optimized GPU implementation.
- We demonstrate the capacity of our differentiable renderer to reconstruct shapes from both artificial and real 3D microscopy images without the need for additional shape-regularization terms.

5.1.1 Related works

Point spread function models

The resolution of images seen with a fluorescence microscope are fundamentally limited by the diffraction of light [11]. If this response is invariant by translation, we can describe the imaging system by determining its PSF, that describes the response of the imaging system to a point source. Numerous physics-based models of PSFs have been developed [12–16], mostly for deconvolution applications [17].

Given an image u_α, deconvoluting a signal aims at inverting the effects of this diffraction, by finding the solution u_ρ of the equation $u_\rho \star h + \xi = u_\alpha$, where ξ is a noise term, which usually follows poisson statistics [18]. A good deconvolution thus requires an accurate determination of the PSF.

The most straightforward way to determine a PSF is to directly generate microscopy images of tiny fluorescent beads that approximates a point source, and then fit the parameters of a given model to match this experimental PSF. In our approach, we decide to approximate the PSF by a Gaussian kernel, that is fully characterized by its covariance matrix. We note that our approach may be extended to learn parameters of more realistic PSF models, especially when vectorized implementations are available [19].

Gibson-Lanni model [12] is based on an optical theory of the diffraction of light [13], and has been used successfully to fit experimental data [20]. Recent works [19] provide efficient implementations based on a decomposition of integrals with Bessel functions that can be integrated directly in a differentiable pipeline. Another approach, applied in [14–16, 21], is based on the fact that the Fourier transform of the PSF can be described by the pupil function, a 2D function defined in the 2D unit disk. An orthogonal decomposition of the pupil function thus provides a flexible way to describe the PSF. In our approach, we decide to express the PSF as a Gaussian blur.

Artificial microscopy images generation

Creating artificial microscopy images has been the object of several studies, in order to evaluate image-analysis algorithms [22] and to create automatically annotated datasets for neural network training [23]. These works can be grouped into two families, with models that aims at reproducing the real image-formation process [24], and other approaches that uses texture

synthesis [25, 26] or generative DL [27, 28] to make photorealistic images.

In [24], fluorophore distribution is defined via a boolean mask that indicates the presence or absence of fluorophores. This mask is subsequently convolved with a PSF provided by the user before adding camera noise. We follow and extend this approach, by giving a more accurate mathematical definition of the fluorophore density, that we define as a $\mathbb{R}^3$-valued density.

Instance segmentation of fluorescent images

Biological image analysis is a subdomain of the broader field of computer vision, and state of the art segmentation methods in both fields followed the same historical evolution [29]: thresholding, watershed-transform [30–32] or optimization with graph-cuts [33] were the dominant segmentation techniques before the advent of deep-learning based pipelines [2, 3].

In 2D, the best results for the segmentation of biological objects have been obtained by using a large dataset of annotated images from varied sources to train a CNN to predict instance masks robustly and generically, being not bound to any particular biological samples nor imaging modalities [**?**, 7, 34]. Such approach could not yet be successfully reproduced in 3D, showing that DL-based volumetric data analysis is not just about extending 2D neural networks to 3D. 3D images, or Z-stacks, are obtained by stacking together 2D images by changing the position of the focal plane along the optical axis z, leading to large anisotropy along this spatial direction. This anisotropy is different for each imaging condition, leading to heterogeneous 3D datasets and making both the training and generalization of CNNs a daunting task. Limited GPU memory also poses large technical difficulties with current 3D biology images of typical size from 512^3 with a classical confocal microscope to 2048^3 with latest light-sheet technologies.

Energy based segmentation

A large portion of the classical literature on image segmentation relies on the minimization of an energy functional $\mathcal{E}(\Lambda, m)$, that aims at modeling a distance between a desired shape Λ and a distinctive feature in the image m [35]. These features may be a sharp intensity gradient (using edge detectors [36]) or regions of largely different intensities [37]. To model the shape of objects, one can generally rely either on an implicit level-set or explicit mesh representation. The final shape Λ^* is obtained iteratively by gradient-based optimization of the functional. The opposite of the gradient $\dfrac{\partial \mathcal{E}(\Lambda, m)}{\partial \Lambda}$ may be interpreted as a force that moves the contour to the desired target shape, which has motivated the generic names of *snakes*, *active contours* [38] or *active meshes* [39–41] for such methods. One large drawbacks, it that these techniques most often require user-defined shape regularization terms (mechanically equivalent to global tension or bending energies) to penalize sharp features and obtain smooth shapes.

The peculiarity of fluorescent images of slender biological structures, such as membranes or filaments, is that the shape is defined by a thin region of high intensity immersed in the dark, which is not well-adapted for region-based or edge-based energies evoked earlier. Alternative approaches have directly defined a force based on the distance to the local maximum intensity in the image [40, 42]. Others have defined a proper energy functional as a distance between a microscopy image and an artificial image created from a mesh, as in [43, 44]. The artificial image is generated by creating a thin boolean mask from the mesh, that is then convolved with a PSF, that is assumed to be known. Our approach builds on these ideas but proposes a

more rigorous, versatile and three-dimensional image rendering procedure, where we optimize both the mesh representing the shape and the PSF parameters together, without needing any explicit regularization terms to smooth the mesh.

Differentiable rendering

Rendering geometrical 2D or 3D shapes into raster (discrete) 2D images is an ubiquitous topic in computer graphics, and an increasing number of rendering frameworks are now made differentiable. This allows to solve inverse-rendering problems, where one can learn directly the parameters of a scene (shapes, textures, material properties) from single or multiple views of raster images [45–47]. In most cases, rasterization pipelines are not differentiable natively and discontinuities or occlusions can lead to incorrect gradients. A simple solution to soften sharp changes is to smoothen the image formation process with different strategies [48–51].

Our fluorescence microscope renderer is based on mathematical operations that are differentiable by design, and does not need special adaptations to obtain meaningful gradients. Indeed, convoluting the mesh FT with a PSF naturally smoothes the image, in a similar manner as what is done for differentiable rasterization [49]. As for other frameworks, the number of parameters to optimize can be very high: a triangle mesh contains typically thousands of vertices, and a PSF may be parametrized by hundreds of parameters. Assuming all the building blocks of the models are composed of differentiable operations, reverse-mode differentiation (also called backpropagation), based on the repeated use of the chain-rule, allows to compute efficiently partial derivatives with respect to any parameter. The efficiency and versatility of this method lead to the development of several libraries offering both GPU acceleration of classical automatic differentiation operations [52, 53] and a high level and memory efficient implementation of backpropagation. We built our pipeline in PyTorch [52] and provide our mesh FT as a differentiable function.

5.2 Rendering of microscopy images

5.2.1 Rendering fluorescence microscopy images

Images are defined as intensity maps from $[0,1]^3$ to $\mathbb{R}$ without any loss of generality, as any non-cubic images can be linearly fitted in the cube $[0,1]^3$. In the following, we will offer approximated descriptions of both the imaging system and the geometry of the biological samples, that will be combined to build our differentiable renderer.

Translation-invariant rendering model

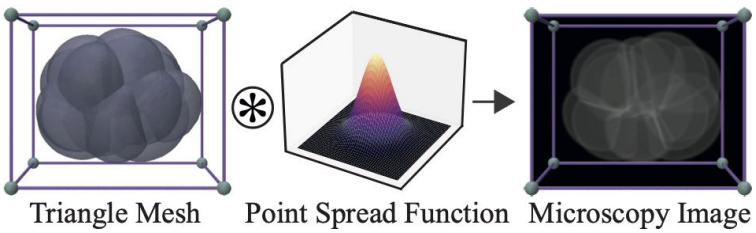


Figure 5.1: Principle of the rendering process. In our model, the fluorophores are distributed regularly on a surface Λ and the imaging system has a translation-invariant response function defined by the PSF. The microscopy image is created by convoluting the PSF with this uniform distribution of fluorophores.

We assume translation-invariance of the response function h of the fluorescence microscope. If so, a smooth image $u_\alpha : [0,1]^3 \mapsto \mathbb{R}$ results from the convolution between a density of fluorophores $u_\Lambda : [0,1]^3 \mapsto \mathbb{R}$ and a point spread function kernel $h : [0,1]^3 \mapsto \mathbb{R}$:

$$u_\alpha(\mathbf{p}) = (u_\rho * h)(\mathbf{p}) = \int_{[0,1]^3} u_\Lambda(\mathbf{x}) h(\mathbf{p}-\mathbf{x}) \mathrm{d}^3\mathbf{x}. \tag{5.1}$$

To avoid the computation of these integrals, we perform the convolution in the Fourier space by doing an element-wise multiplication of the elements of the Fourier transforms $\hat{u}_\rho$ and $\hat{h}$:

$$\hat{u}_\alpha = \hat{u}_\Lambda \cdot \hat{h}. \tag{5.2}$$

The biological structure of interest is made visible by fluorophores. The shape will thus be represented by u_Λ, whereas h should describe the response of the optical system to a point source (the PSF). Later for differentiable rendering applications, we will see that with this image formation model, one can start from a first guess (u^0_Λ, h^0), and minimize the distance between the rendered image u^0_α and a real microscopy image m. Doing so allows to learn both the geometry of the biological sample observed and the PSF of the imaging system, (u^*_Λ, h^*). However, to learn meaningful representations, both of these elements have to be constrained by parametrized models that will implement our prior knowledge of the system.

5.2.2 Geometrical models of biological objects

Extracting geometry from volumetric images consists of giving approximate representations of biological objects in terms of ND (N = 0,1,2,3) discrete objects, embedded in a 3D space. They can be pretty diverse, both in terms of size and of topology. Here, we decide to focus on the lipid membranes that delimits cells and nuclei and to approximate them as 2D surfaces, that can be modeled efficiently with triangle meshes. Choosing the topology in such manner also implements prior knowledge: as we see in section 5.6.1, early embryos are composed of cells forming bounded regions that can be represented as a single non-manifold multimaterial mesh [54].

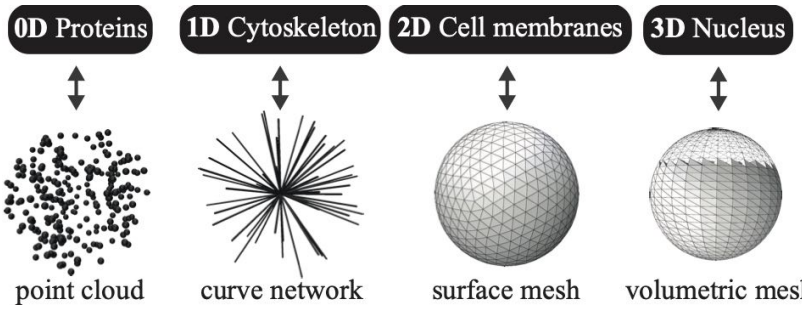


Figure 5.2: We can interpret each biological object as a geometrical object.

They are pretty diverse, both in terms of size and of topology. In the scales studied, one could consider doing the following approximations:

- **Proteins** are seen as 0D points, as the resolution is not sufficient to see individual molecules.
- **Cytoskeletons** of cells, composed of actin filaments and microtubules, can be approximated by networks of 1D polygonal curves.
- **Lipid membranes** that delimits cells and nuclei are 2D, and can be modeled efficiently with triangle meshes.

- **Bulk 3D organelles** such as nuclei or phase-separated liquid droplets are described as 3D volumetric meshes.

5.2.3 Fourier Transform of Surfaces

In this section, we will give an explicit expression of the FT for any 2D surfaces embedded in $\mathbb{R}^3$, and give formulas of the FT and its gradient with respect to vertices position in the particular case of a triangle mesh.

Surfaces as spatial Dirac distributions

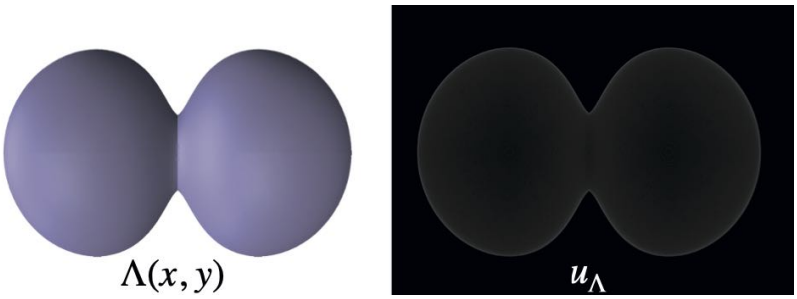


Figure 5.3: Surface Λ describing a dividing cell and its associated density u_Λ in $\mathbb{R}^3$.

The density of fluorophores u_Λ will be equal to 1 only on an infinitely thin surface Λ, and 0 everywhere else.
To describe such irregular functions, one has to define them using spatial Dirac distributions. If we consider a surface given by a parameterization $\Lambda : [0,1]^2 \to [0,1]^3$, its spatial density is given by:

$$u_\Lambda = \frac{1}{|\Lambda|} \int_0^1 \int_0^1 \delta_\Lambda(x,y) a_\Lambda(x,y) \mathrm{d}x\mathrm{d}y, \tag{5.3}$$

where a_Λ is the surface element given by $a_\Lambda(x,y) = \|\partial_x \Lambda \times \partial_y \Lambda\|$, and $|\Lambda|$ is the total area of the surface Λ. The normalization by the total area ensures a density 1. When dealing with several surfaces of different densities, one has only to use a weigthed sum of such u_Λ.
For all $\mathbf{z} \in \mathbb{R}^3$, the FT of the Dirac distribution at z is given by:

$$\hat{\delta}_{\mathbf{z}}(\boldsymbol{\xi}) = e^{-i\mathbf{z}\cdot\boldsymbol{\xi}}, \quad \boldsymbol{\xi} \in \mathbb{R}^3. \tag{5.4}$$

Hence, the linearity of the FT gives:

$$\hat{u}_\Lambda(\boldsymbol{\xi}) = \frac{1}{|\Lambda|} \int_0^1 \int_0^1 a_\Lambda(x,y) e^{-i\boldsymbol{\Lambda}(x,y)\cdot\boldsymbol{\xi}} \mathrm{d}x\mathrm{d}y. \tag{5.5}$$

The case of a triangulated surface

To describe surfaces Λ in practice, we will use triangle meshes. To compute the FT in this special case, as the FT is linear we just need to compute the FT of each triangle of the surface and sum their contributions.

Fourier transform and gradients for a triangle

We consider one triangle $\mathcal{T}$ given by its vertices $(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$ and compute u_Λ and its spatial derivatives. We denote by $\mathbf{v}_4 = \mathbf{v}_1$ and similarly by $\mathbf{v}_0 = \mathbf{v}_3$. For $p = 1 \dots 3$, we define

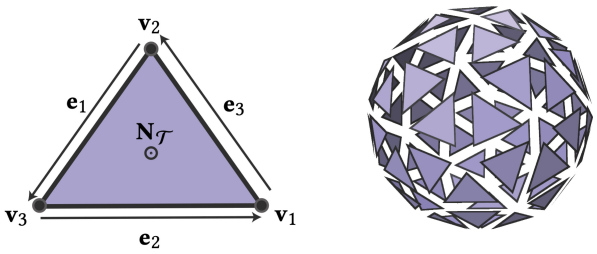


Figure 5.4: Notations for a triangle mesh. As the FT is linear, the FT of a single triangle will allow us to obtain the general expression of the FT for any triangle mesh.

$p^- = p-1$ and $p^+ = p+1$ and denote by $\mathbf{e}_p = \mathbf{v}_{p^-} - \mathbf{v}_{p^+}$ the opposite edge to $\mathbf{v}_p$ and by $l_p = |\mathbf{e}_p|$ its length. $A_\mathcal{T} = \dfrac{|\mathbf{e}_3 \times \mathbf{e}_1|}{2}$ denotes the area of the triangle, and $\mathbf{N}_\mathcal{T}$ denotes its unit normal given by: $\mathbf{N}_\mathcal{T} = \dfrac{\mathbf{e}_3 \times \mathbf{e}_1}{2A_\mathcal{T}}$ At last, for $p = 1 \dots 3$, we define $\mathbf{w}_p = \mathbf{e}_p \times \mathbf{N}_\mathcal{T}$, the non-normalized outward normal to $\mathcal{T}$ on the edge $\mathbf{e}_p$. For any p, we have: $\dfrac{\partial A_\mathcal{T}}{\partial \mathbf{v}_p} = -\dfrac{\mathbf{w}_p}{2}$

If we define $\hat{u}_\mathcal{T}(\boldsymbol{\xi}) = \int_\mathcal{T} \mathrm{e}^{-i\mathbf{z}\cdot\boldsymbol{\xi}} \mathrm{d}s(\mathbf{z})$, then for all $\boldsymbol{\xi} \in \mathbb{R}^3$, we have:

$$\hat{u}_\mathcal{T}(\boldsymbol{\xi}) = 2A_\mathcal{T} f_\mathcal{T}(\boldsymbol{\xi}), \tag{5.6}$$

with $f_\mathcal{T}(\boldsymbol{\xi})$ defined by:

$$f_\mathcal{T}(\boldsymbol{\xi}) = \sum_{p=1}^{3} \frac{\mathrm{e}^{-i\mathbf{z}_p\cdot\boldsymbol{\xi}}}{(\mathbf{e}_{p^-}\cdot\boldsymbol{\xi})(\mathbf{e}_{p^+}\cdot\boldsymbol{\xi})}. \tag{5.7}$$

For all $p = 1 \dots 3$, we have:

$$\frac{\partial \hat{u}_\mathcal{T}}{\partial \mathbf{v}_p}(\boldsymbol{\xi}) = -f_\mathcal{T}(\boldsymbol{\xi})\mathbf{w}_p + 2A_\mathcal{T}\frac{\partial f_\mathcal{T}(\boldsymbol{\xi})}{\partial \mathbf{v}_p}, \tag{5.8}$$

with:

$$\begin{aligned}\frac{\partial f_\mathcal{T}(\boldsymbol{\xi})}{\partial \mathbf{v}_p} =& \boldsymbol{\xi}\left[\frac{\mathrm{e}^{-\mathrm{i}\mathbf{z}_{p^+}\cdot\boldsymbol{\xi}}}{(\mathbf{v}_{p^-}\cdot\boldsymbol{\xi})^2(\mathbf{v}_p\cdot\boldsymbol{\xi})} - \frac{\mathrm{e}^{-\mathrm{i}\mathbf{z}_{p^-}\cdot\boldsymbol{\xi}}}{(\mathbf{v}_p\cdot\boldsymbol{\xi})(\mathbf{v}_{p^+}\cdot\boldsymbol{\xi})^2}\right. \\ & -\frac{\mathrm{i}\mathrm{e}^{-\mathrm{i}\mathbf{z}_p\cdot\boldsymbol{\xi}}}{(\mathbf{v}_{p^-}\cdot\boldsymbol{\xi})(\mathbf{v}_{p^+}\cdot\boldsymbol{\xi})} + \frac{\mathrm{e}^{-\mathrm{i}\mathbf{z}_p\cdot\boldsymbol{\xi}}}{(\mathbf{v}_{p^-}\cdot\boldsymbol{\xi})^2(\mathbf{v}_{p^+}\cdot\boldsymbol{\xi})} \\ & \left.-\frac{\mathrm{e}^{-\mathrm{i}\mathbf{z}_p\cdot\boldsymbol{\xi}}}{(\mathbf{v}_{p^-}\cdot\boldsymbol{\xi})(\mathbf{v}_{p^+}\cdot\boldsymbol{\xi})^2}\right].\end{aligned} \tag{5.9}$$

$$\boxed{\begin{aligned}\frac{\partial f_\mathcal{T}(\xi)}{\partial \mathbf{v}_p} =& \left[\left(-\frac{\mathrm{i}}{(v_{p^-}\cdot\xi)(v_{p^+}\cdot\xi)} + \frac{1}{(v_{p^-}\cdot\xi)^2(v_{p^+}\cdot\xi)}\right.\right. \\ & \left.-\frac{1}{(v_{p^-}\cdot\xi)(v_{p^+}\cdot\xi)^2}\mathrm{e}^{-\mathrm{i}z_p\cdot\xi}\right) \\ & \left.+\frac{\mathrm{e}^{-\mathrm{i}z_{p^+}\cdot\xi}}{(v_{p^-}\cdot\xi)^2(\mathbf{v}_p\cdot\xi)} - \frac{\mathrm{e}^{-\mathrm{i}z_{p^-}\cdot\xi}}{(v_p\cdot\xi)(v_{p^+}\cdot\xi)^2}\right]\boldsymbol{\xi}.\end{aligned}} \tag{5.10}$$

$$\boxed{\begin{aligned}\frac{\partial f_\mathcal{T}(\xi)}{\partial \mathbf{v}_p} =& \boldsymbol{\xi}\left[\frac{\mathrm{e}^{-\mathrm{i}z_{p^+}\cdot\xi}}{(v_{p^-}\cdot\xi)^2(v_p\cdot\xi)} - \frac{\mathrm{e}^{-\mathrm{i}z_{p^-}\cdot\xi}}{(v_p\cdot\xi)(v_{p^+}\cdot\xi)^2}\right. \\ & \left.-\frac{\mathrm{i}\mathrm{e}^{-\mathrm{i}z_p\cdot\xi}}{(v_{p^-}\cdot\xi)(v_{p^+}\cdot\xi)} + \frac{\mathrm{e}^{-\mathrm{i}z_p\cdot\xi}}{(v_{p^-}\cdot\xi)^2(v_{p^+}\cdot\xi)} - \frac{\mathrm{e}^{-\mathrm{i}z_p\cdot\xi}}{(v_{p^-}\cdot\xi)(v_{p^+}\cdot\xi)^2}\right]\end{aligned}} \tag{5.11}$$

Density for a triangle mesh

A triangle mesh is a surface $\Lambda = \{\mathcal{T}\}$ defined by a set of triangles. For $\boldsymbol{\xi} \in \mathbb{R}^3$, by linearity, its FT is defined by:

$$\hat{u}_\Lambda(\boldsymbol{\xi}) = \frac{\sum\limits_{\mathcal{T}\in\Lambda} \hat{u}_\mathcal{T}(\boldsymbol{\xi})}{|\Lambda|} = \frac{\sum\limits_{\mathcal{T}\in\Lambda} \hat{u}_\mathcal{T}(\boldsymbol{\xi})}{\sum\limits_{\mathcal{T}\in\Lambda} A_\mathcal{T}}. \tag{5.12}$$

The gradient of the FT with respect to a vertex $\mathbf{v}$ of the mesh is thus:

$$\frac{\partial \hat{u}_\Lambda(\boldsymbol{\xi})}{\partial \mathbf{v}} = \frac{1}{|\Lambda|}\left(\sum_{\mathcal{T}\in\mathbf{v}\star} \frac{\partial \hat{u}_\mathcal{T}(\boldsymbol{\xi})}{\partial \mathbf{v}} - \hat{u}_\Lambda(\boldsymbol{\xi}) \sum_{\mathcal{T}\in\mathbf{v}\star} \frac{\partial A_\mathcal{T}}{\partial \mathbf{v}}\right), \tag{5.13}$$

where $\mathbf{v}\star$ denotes the set of the triangles of Λ that contains the vertex $\mathbf{v}$.

Numerical approximations to avoid divergence

The FT of a Dirac distribution on a triangle is C^∞. However, large computational errors may arise when one denominator in the eq. (5.6) get close to zero. Error in rounding floating-point arithmetic leads to a limited numerical precision ϵ. Values smaller than this threshold ϵ cannot, in practice, be distinguished from 0. Therefore, when one term in denominator gets close to 0 in the expression (5.6) we replace it by an approximation that we describe in the following.
We write $f_\mathcal{T}(\boldsymbol{\xi}) = g(\mathbf{e}_1 \cdot \boldsymbol{\xi}, \mathbf{e}_2 \cdot \boldsymbol{\xi}, \mathbf{e}_3 \cdot \boldsymbol{\xi})$, with the function g defined for $(s,t,u) \in \mathbb{R}^3$ by

$$g(s,t,u) = \frac{-\mathrm{e}^{is}}{(s-t)(s-u)} - \frac{\mathrm{e}^{it}}{(t-u)(t-s)} - \frac{\mathrm{e}^{iu}}{(u-s)(u-t)}. \tag{5.14}$$

When two values (a,b) among (s,t,u) are such that $|a-b| < \epsilon$, we obtain a value equal to 0 at the denominator and divergence. We solve this problem by deriving exact expressions of g(s,t,u) in the case where t=u or t=s or u=s or u=s=t, that replaces the original expression (5.14) when any (a,b) among (s,t,u) is such that $|a-b| < \epsilon$.
When only two values are too close from each other:

$$\begin{aligned} g(t,t,u) &= g(u,t,t) = g(t,u,t) \\ &= i\frac{\mathrm{e}^{-it}}{t-u} + \frac{\mathrm{e}^{-it}}{(t-u)^2} - \frac{\mathrm{e}^{-iu}}{(t-u)^2}. \end{aligned} \tag{5.15}$$

When all three values are too close to each other, we replace the expression by

$$g(u,u,u) = \frac{\mathrm{e}^{-iu}}{2}. \tag{5.16}$$

5.3 Point Spread-Functions Models

We provide a short introduction of three PSF models: the gaussian PSF, and two photorealistic models: Gibson-Lanni and the Pupil-function based PSF. When formulated appropriately, they all lead to efficient differentiable expressions of the psf, allowing to learn the physical quantities that defines them.

5.3.1 Gaussian PSF

The simplest model is a gaussian PSF. In this case the PSF is fully described by its covariance matrix $\Sigma \in^{n*n}$:

$$h(z) = \frac{\mathrm{e}^{-\frac{1}{2}z^T\Sigma^{-1}z}}{\sqrt{(2\pi)^n \det\Sigma}}\ ,\ z \in^n \tag{5.17}$$

The Fourier transform $\hat{h}$ of h is given by:

$$\hat{h}(\xi) = \mathrm{e}^{-\frac{1}{2}\xi^T\Sigma^{-1}\xi}, \xi \in^n \tag{5.18}$$

.

In this framework, learning the PSF consists of learning all the coefficients of the covariance matrix Σ. This is the PSF with the most previsible behavior, that will not give rise to any unexpected artifacts, and will be the one favored in a first approach.

5.3.2 Gibson and Lanni model

The optical model

The Gibson Lanni model [12] is based on the hypothesis that the objective lens is completely free of aberrations. Thus, all the aberrations of the imaging system are generated by the other elements of the microscope: the specimen, the coverslip, and the immersion medium. In the optimal design conditions, the specimen is supposed to be in contact with the coverslip, which is not the case in practice. Moreover, the width and refraction indices of the coverslip and the immersion medium may differ from their original design values. Because of these differences, the optical path of a light ray in a design system will differ from one in the experimental condition. If we note ρ the normalized radius of the focal plane, z the axial coordinate of the focal plane, and $\mathbf{p} = (NA, n, t)$ with $n = (n_i, n_i^*, n_g, n_g^*, n_s)$ and $t = (t_i, t_i^*, t_g, t_g^*, z_p)$, the optical path difference OPD is given by:

$$\begin{aligned} OPD(\rho, z, \mathbf{p}) = \quad & (z + t_i)\sqrt{(n_i)^2 - (NA\rho)^2} + \\ & z_p\sqrt{n_s^2 - (NA\rho)^2} - t_i^*\sqrt{n_i^{*2} - (NA\rho)^2} + \\ & t_g\sqrt{n_g^2 - (NA\rho)^2} - t_g^*\sqrt{n_g^{*2} - (NA\rho)^2} \end{aligned} \tag{5.19}$$

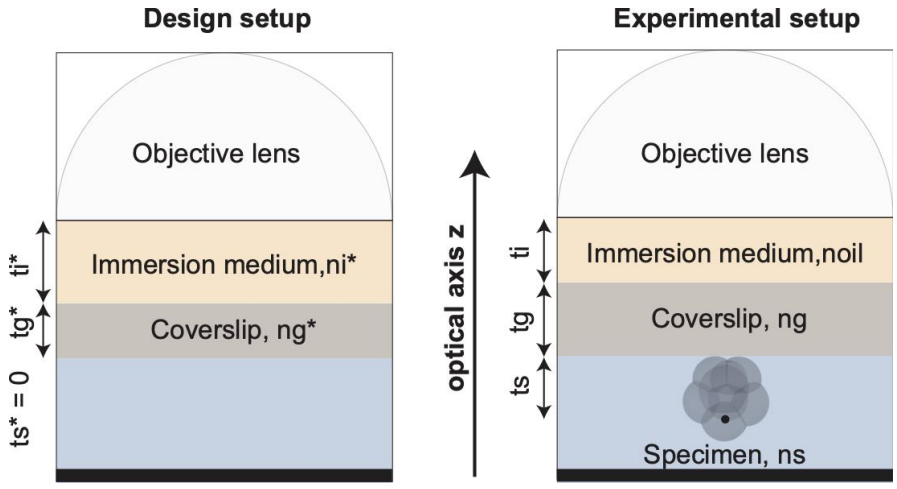


Figure 5.5: Even with an aberration-free objective lens, differences between design and experiment parameters can lead to optical aberrations, that induces modifications of the PSF

Expression of the PSF

Under the previous hypotheses, the phase aberration can be expressed as $W(\rho, z, \mathbf{p}) = kOPD(\rho, z, \mathbf{p})$ with $k = 2\pi/\lambda$ the wave number of the emitted light.
The PSF is radially symmetric in a plane orthogonal to the optical axis. Given the phase aberration, we can obtain its expression using Kirchhoff's diffraction integral formula [12, 55]

$$PSF(r, \rho, z, \mathbf{p}) = \left| \int_0^1 \mathrm{e}^{iW(\rho,z,\mathbf{p})} J_0(krNA\rho)\rho d\rho \right|^2 \tag{5.20}$$

Where r is the distance to the optical axis, and J_0 is the Bessel function of the first kind of order 0.

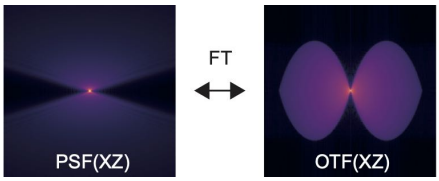


Figure 5.6: XZ values of the Gibson-Lanni PSF

We will see that this model gives a PSF that is very similar to the one computed from the pupil function, with standard parameters. However, in both models, modifying the optical parameters can change considerably the PSF.

Fast vectorial computation of the PSF

The previous integral is tedious to compute, and has been the object of many studies. We follow the approach of [19], that allows for a fast computation of the PSF using a Bessel-series approximation. The vectorial nature of the computation is easily expressed in PyTorch, and its derivatives obtained straightforwardly using reverse-mode differentiation.

5.3.3 Pupil-function based model

The OTF and the pupil function

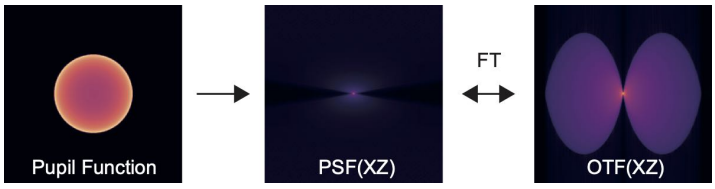


Figure 5.7: The 2D-valued pupil function can be used to generate the PSF, and its fourier transform, the OTF

The PSF specifies the response of the imaging system to a point source object in the physical space $\mathbb{R}^3$. It always depends on the optical properties of the system such as the aperture angle α, the wavelength of the observation light λ or the refractive index of the immersion medium n.
The PSF can be described equivalently in the space of spatial frequencies $(\mathbf{k}_x, \mathbf{k}_y, \mathbf{k}_z)$ by its Fourier transform, called the Optical Transfer Function (OTF). On a fluorescence microscope, the OTF has non-zero values only on a spherical cap of radius $\frac{n}{\lambda}$, with an angular span fixed by the aperture angle α. This 2D shell can be projected on the plane $(\mathbf{k}_x, \mathbf{k}_y)$, by parametrizing

$\mathbf{k}_z = \sqrt{(\frac{n}{\lambda})^2 - (\mathbf{k}_x^2 + \mathbf{k}_y^2)}$, leading to a complex $\mathbb{R}^2$ valued function non-zero only on the unit disc, called the pupil function $P(k_x, k_y)$. Eventually, the PSF can be described as:

$$PSF(x, y, z) = \left| \int \int_{pupil} P(k_x, k_y) e^{2\pi i (k_x x + k_y y + k_z z)} \mathrm{d}_{k_x} \mathrm{d}_{k_y} \right| \tag{5.21}$$

The PSF can thus be described via a 2D pupil function defined on a disk. An efficient parametrization of this pupil function can be done by decomposing it into its expansion in the orthogonal basis of Zernike polynomials.

Zernike Polynomials

Zernike Polynomials are a sequence of polynomials orthogonals on the unit disk, that are used to represent optical aberrations [55]. There are odd and even Zernike polynomials. With φ the azimutal angle and ρ the radial distance, the even polynomials have for expression:

$$Z_n^m(\rho, \varphi) = R_n^m(\rho) \cos(m\,\varphi) \tag{5.22}$$

and the odd ones have for expression:

$$Z_n^{-m}(\rho, \varphi) = R_n^m(\rho) \sin(m\,\varphi), \tag{5.23}$$

where m and n are nonnegative integers with $0 \leqslant m \leqslant n$ ($m = 0$ for even Zernike polynomials), and R_n^m are the radial polynomials defined by:

$$R_n^m(\rho) = \sum_{k=0}^{\frac{n-m}{2}} (-1)^k \binom{n-k}{k} \binom{n-2k}{\frac{n-m}{2} - k} \rho^{n-2k}. \tag{5.24}$$

Each coefficient of the expansion can be learned efficiently. A strengh of this formulation is that Zernike polynomials represent classical optical aberrations (Fig. 5.8), giving insights into the real optical properties of the system.

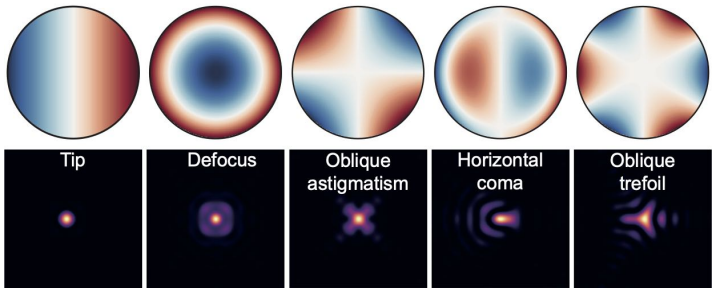


Figure 5.8: Zernike Polynomials are othogonal polynomials in the unit disc that can describe optical aberrations by modifying the pupil function

5.3.4 Confocal Microscopy PSFs

To model the point spread function (PSF) of a confocal microscope, we reproduce all the steps involved in the image-formation process.

- **Excitation PSF**: The excitation PSF (PSF_{exc}) represents the intensity distribution of the focused light used to illuminate the sample. It is typically modeled as an Airy pattern, which is the diffraction pattern created by a circular aperture. We can use the previous PSF model based on a pupil function.

- **Detection PSF**: The detection PSF (PSF_{det}) represents the intensity distribution of the emitted fluorescence light that is focused onto the detector. It is also typically modeled as an Airy pattern, similar to the excitation PSF, we will thus also use the PSF model based on a pupil function.

- **Total PSF**: The total PSF (PSF_{total}) is obtained by multiplying the excitation and detection PSFs:

$$PSF_{total} = PSF_{exc} \times PSF_{det}.$$

- **Pinhole function**: The pinhole in a confocal microscope serves to block out-of-focus light and improve the axial resolution. The function representing the pinhole ($F_{pinhole}$) is typically modeled as a circular top-hat function, which has a uniform value of 1 within a specific circle (the pinhole) and 0 elsewhere (Figure 5.9) The size of the pinhole, often expressed in Airy units (AU), is defined as:

$$AU = 1.22\frac{\lambda}{NA},$$

  where λ is the wavelength of light and NA is the numerical aperture of the microscope objective.

- **Convolution with the pinhole function**: The total PSF is convolved with the pinhole function to obtain the final PSF of the confocal microscope ($PSF_{confocal}$):

$$PSF_{confocal} = PSF_{total} * F_{pinhole},$$

  where the asterisk $*$ denotes the convolution operation. This step accounts for the effects of the pinhole on the detected signal, which helps to improve the axial resolution and reject out-of-focus light.

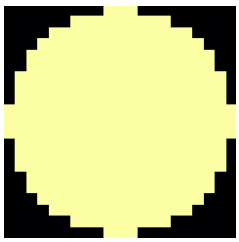


Figure 5.9: We model the pinhole by convolving the detection PSF with a pinhole function

5.3.5 Evaluation of the PSFs

These algorithms allowed us to compute artificial images using a well defined point-spread function. This is a way to test the different point spread function, by using them to create real images.
We inspired us heavily from the python package PyOTF, which uses pupil-function based computations, and combine them with each-other to create photorealistic point spread function of given optical setups, such as confocal microscopes or SLIM microscopes.

Aliasing and Nyquist criterion, a Fourier-Space view

Because of our discretized representation of the signal, we have a cutoff spatial frequency: If we try to represent discretely a signal of too high frequency on a too coarse grid, we will obtain aliasing, which will result in dramatic alterations of our PSF (Figure 5.10). Aliasing can be seen very efficiently by looking at the OTF. When the nyquist criterion is respected, all the frequencies available are well represented. When it is not, the spectrum is truncated, which could result in undesirable effects. The only option is to use a bigger box, to represent a larger set of frequencies.

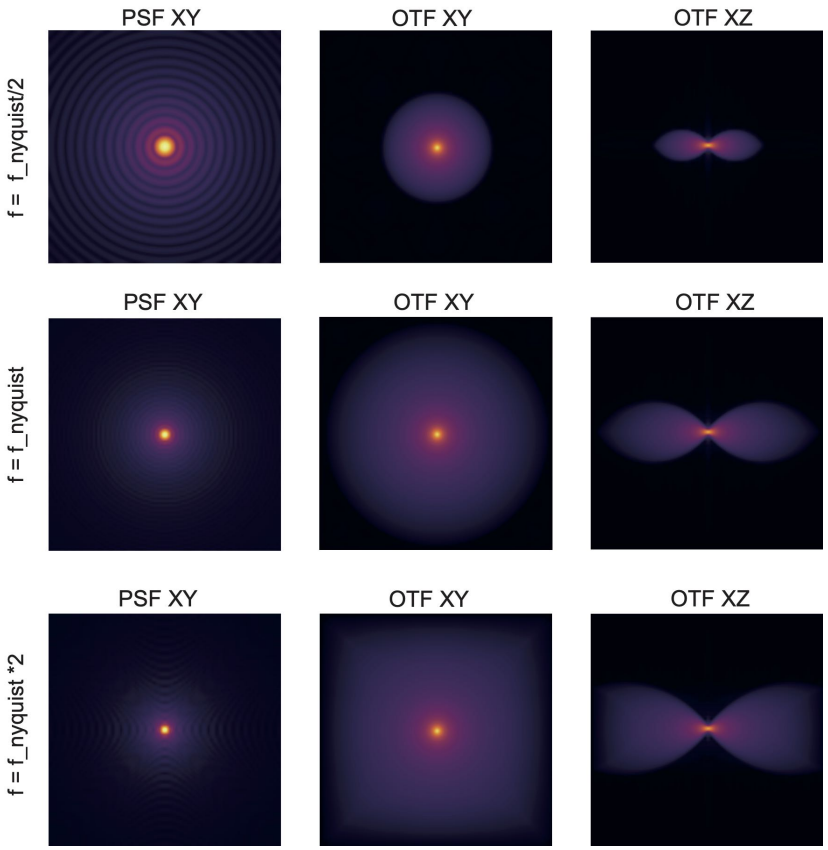


Figure 5.10: Because of aliasing, we have limited in the range of spatial resolution that we can represent numerically, for a given box.

Application: Creation of images with PSF models

All the previous PSFs models were introduced based on physical models. Our artificial image generators allows us to do something that has never been done before: seeing the effects on the different PSFs on a microscopy image.
We generated $[600]^3$ images of a sphere, to observe the results of all the PSFs we talked about. We represent for each image the central plane in the XY and XZ direction, and the correspond values of their PSFs and OTFs (Figure 5.11).

Gibson-Lanni PSF

We create two Gibson Lanni PSFs. We choose the lateral resolution to be $0.055\mu m^{-1}$, which corresponds to half the nyquist criterion. For both of them, we choose $NA = 1.27, \lambda =$

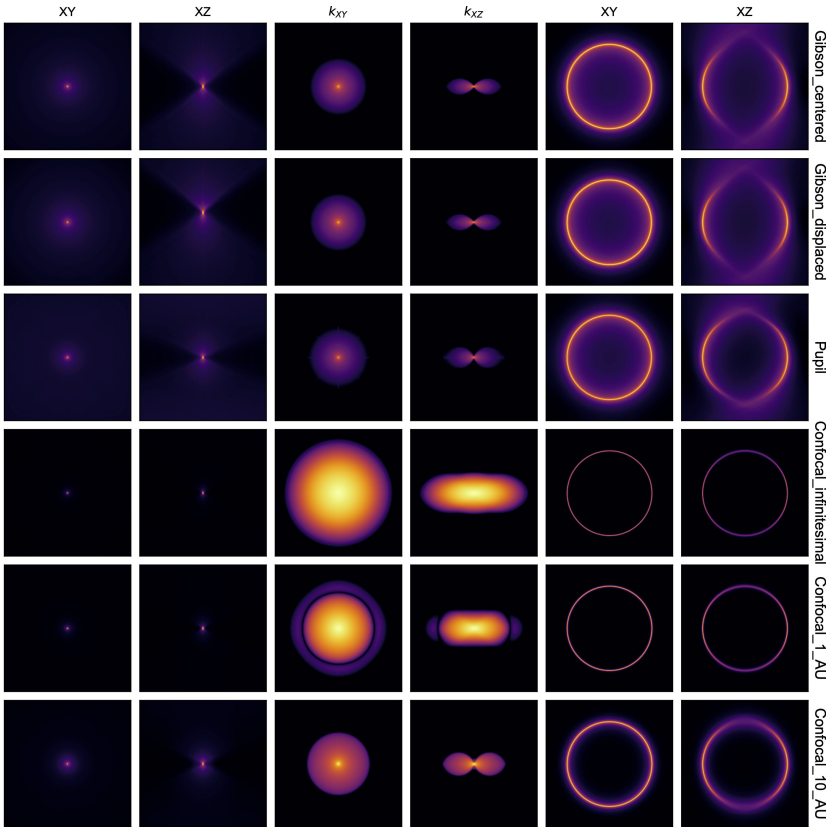


Figure 5.11: We created images using the PSFs that we discussed of, to see if they represent accurately the reality.

$0.585\mu m$, and decide, for simplicity, to not add perturbations due to the optical path differences: $n_i = n_i^*$, $n_g = n_g^*$, $t_g = t_g^*$, $t_i = t_i^*$. The only parameter that we tune is z_p, the sample distance to the coverslip. For the Gibson_centered we put $z_p = 0$, and for the Gibson_displaced we put $z_p = 2\mu m$.
We notice that the images created using the Gibson Lanni model are very perturbed. Even without additional optical path perturbation, we represent a PSF that alterates a lot the signal. Gibson Lanni may thus not be the best suited PSF model.

Pupil function

We also create an image purely based on the pupil function. As before, we choose the lateral resolution to be $0.055\mu m^{-1}$, $NA = 1.27, \lambda = 0.585\mu m$, and do not apply any optical aberrations from Zernike polynomial coefficients, to see the raw effects of the PSF. We use $n_i = 1.33$.

Confocal microscopes

We create images of confocal microscopes. As before, we chose the lateral resolution to be $0.055\mu m^{-1}$, $NA = 1.27, \lambda = 0.561\mu m$ for the excitation PSF and $\lambda = 0.585\mu m$ for the detection PSF. We do not apply any optical aberrations from Zernike polynomial coefficients, and use $n_i = 1.33$.
We compute three PSFs, one with a perfect pinhole function (infinite Aperture), one with a pinhole of 1 AU, and one with a pinhole of 10 AU.

5.4 SuperResolution Microscopy models

Now we will emulate, as well, the effect of several super resolution microscopy methods. We consider that our PSF is $PSF_{confocal}$, with a pinhole of 10 AU, and will mimic the superresolution process trying to enhance the resolution of a microscope design. The common point of all these methods is to create a shift of the PSF in Fourier Space, to enlarge the spectrum of available frequencies and thus beat the resolution limit.

5.4.1 Structured illumination microscopy

To model the point spread function (PSF) for a structured illumination microscopy (SIM) system, including both Wiener and Lattice SIM, we must consider different aspects of the optical system and the structured illumination patterns. Here is a detailed explanation of the process for each type of SIM, accompanied by relevant references:
In standard SIM, the sample is illuminated with a sinusoidal pattern of varying spatial frequency and orientation. The interference between the structured illumination and the sample's spatial frequencies generates moiré fringes, which are detected and processed to reconstruct a high-resolution image [56].
The spatial frequency f and the maximum angle of incidence α are calculated using the refractive index and the numerical aperture:

$$f = 2\pi\frac{n_i}{\lambda}, \quad \alpha = \arcsin\left(\frac{NA}{n_i}\right). \tag{5.25}$$

The structured illumination pattern is calculated for each orientation. The orientations are given by:

$$\phi \in \{0, \frac{2\pi}{3}, \frac{4\pi}{3}\}.$$

For each orientation, the structured illumination pattern s is calculated by summing the exponential terms of the plane waves at angles $-\alpha$, 0, and α:

$$s(x,y,z) = \sum_{\substack{c\in\{1,0.5,1\}\\ \theta\in\{-\alpha,0,\alpha\}}} c \times e^{if[(x\cos(\phi)+y\sin(\phi))\sin(\theta)+z\cos(\theta)]}.$$

Finally, the SIM PSF is obtained by summing the product of the generated PSF with the squared amplitude of the structured illumination pattern s for each orientation:

$$PSF_{SIM} = \sum_{\phi\in\{0,\frac{2\pi}{3},\frac{4\pi}{3}\}} PSF_{confocal} \times |s|^2.$$

These formulas help to generate the point spread function for a structured illumination microscopy system with three orientations, illustrating the interplay between the optical parameters and the structured illumination pattern.

5.4.2 Wiener SIM

Wiener SIM is an extension of standard SIM, where the reconstruction algorithm is based on Wiener filtering [57]. This method attempts to reduce the noise amplification associated with the standard SIM reconstruction process. The Wiener SIM PSF (PSF_{Wiener}) can be modeled by applying a Wiener filter (W_{Wiener}) to the standard SIM PSF:

$$PSF_{Wiener} = W_{Wiener} * PSF_{SIM}.$$

First, the optical transfer function (OTF) is calculated by taking the squared magnitude of the Fourier transform of the PSF:

$$OTF = |FT(PSF)|^2.$$

A Wiener filter is then applied to the OTF using a constant w, which is determined as the maximum value of the OTF divided by 100:

$$w = \frac{\max(OTF)}{100}, \quad OTF_{Wiener} = \frac{OTF}{OTF + w}.$$

The Wiener-filtered PSF is obtained by taking the inverse Fourier transform of the Wiener-filtered OTF:

$$PSF_{Wiener} = FT^{-1}(OTF_{Wiener}).$$

The spatial frequency f and the maximum angle of incidence α are calculated using the refractive index and the numerical aperture:

$$f = 2\pi\frac{n_i}{\lambda}, \quad \alpha = \arcsin\left(\frac{NA}{n_i}\right).$$

The structured illumination pattern is calculated for each orientation. The orientations are given by:

$$\phi \in \{0, \frac{2\pi}{3}, \frac{4\pi}{3}\}.$$

For each orientation, the structured illumination pattern s is calculated by summing the exponential terms of the plane waves at angles $-\alpha$ and α:

$$s(x, y, z) = e^{ixf} + \sum_{\theta \in \{-\alpha, \alpha\}} e^{if[(x\cos(\phi) + y\sin(\phi))\sin(\theta) + z\cos(\theta)]}.$$

Finally, the Wiener-SIM PSF is obtained by summing the product of the Wiener-filtered PSF with the squared amplitude of the structured illumination pattern s for each orientation and subtracting 8 times the Wiener-filtered PSF:

$$PSF_{WienerSIM} = \sum_{\phi} PSF_{Wiener} \times |s|^2 - 8 \times PSF_{Wiener}.$$

5.4.3 Lattice SIM

Lattice SIM is a variant of SIM that uses two-photon excitation and a lattice-shaped illumination pattern to achieve higher imaging speeds and reduced photobleaching compared to other SIM techniques [58]. The Lattice SIM PSF ($PSF_{Lattice}$) can be modeled as the convolution of the two-photon excitation PSF (PSF_{2P}) and the detection PSF (PSF_{det}), convolved with the lattice illumination pattern ($F_{Lattice}$):
The spatial frequency f is calculated using the refractive index and the wavelength:

$$f = 2\pi\frac{n_i}{\lambda}.$$

The structured illumination pattern s is initialized with an exponential term corresponding to a plane wave in the z-direction, summed to the exponential terms of the plane waves at angles $-\alpha$ and α for each orientation. The orientations are given by:

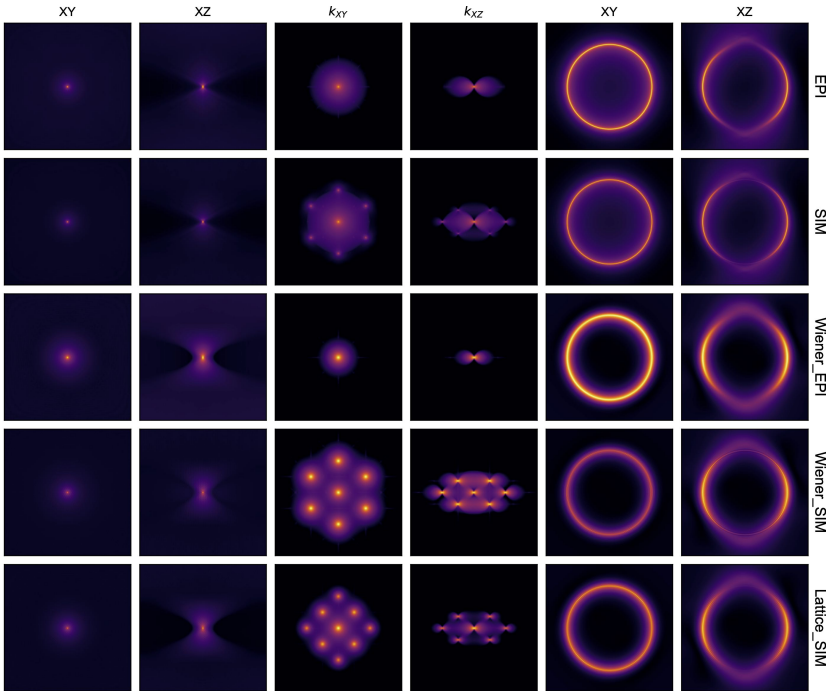


Figure 5.12: We created images using PSFs that were constructed from the pupil function PSFs, that mimic several superresolution modalities.

$$\phi \in \{0, \frac{\pi}{2}\}.$$

The structured illumination pattern s is expressed as follows:

$$s(x,y,z) = e^{izf} + \sum_{\substack{\theta \in \{-\alpha,\alpha\} \\ \phi \in \{0,\frac{\pi}{2}\}}} e^{i[((x\cos(\phi)+y\sin(\phi))\sin(\theta)+z\cos(\theta))\times f]}.$$

Finally, the lattice SIM PSF is obtained by multiplying the Wiener-filtered PSF with the squared amplitude of the structured illumination pattern s:

$$PSF_{LatticeSIM} = PSF_{Wiener}|s|^2.$$

5.4.4 Super resolution and deconvolution

The images generated from these PSFs, are not looking very well. We see that these methods tends to enlarge the spectrum of displayed frequencies (Fig 5.13), but that this spectrum is very non-uniform, which affects the quality of the output image. The role of deconvolution is to flatten this spectrum. We see that the effect of these superresolution methods is thus to enlarge the spectrum of frequencies at the expense of its uniformity, and it is then dependent of a downstream deconvolution, to flatten this spectrum.

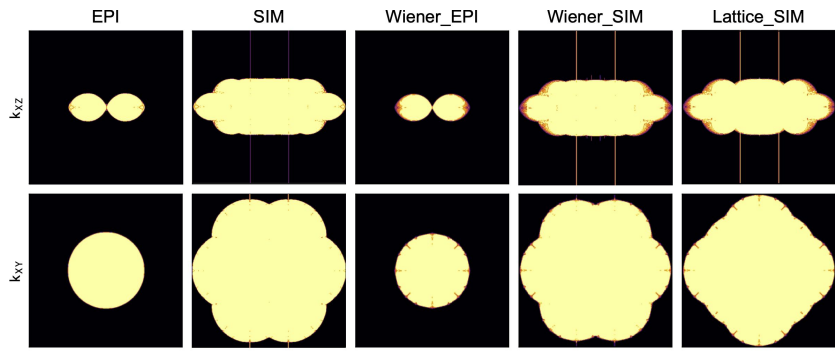


Figure 5.13: Superresolution methods systematically make the spectrum of available frequencies bigger. Combined with deconvolution, they provide better images than what would be possible without them.

5.5 Creation of artificial datasets

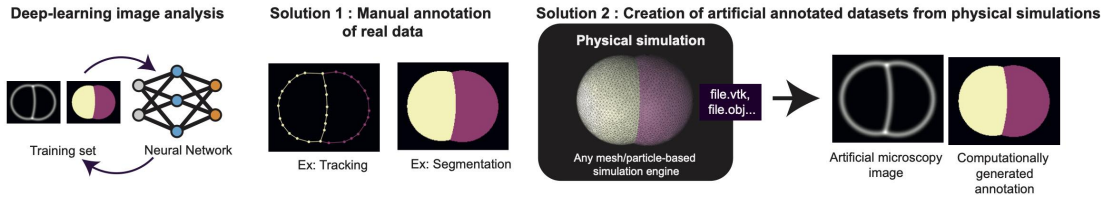


Figure 5.14: Two methods to create annotated datasets

The traditional method for training neural networks for 3D image analysis consists of annotating manually the images by experts. They carefully label the objects or structures of interest to create a ground truth dataset [59] This annotated dataset is then employed to train the neural network, allowing it to learn the features and relationships between the input images and the corresponding labels (Figure 5.14, Solution 1). However, this manual annotation process can be time-consuming, labor-intensive, and costly, especially when dealing with large numbers of 3D images. An alternative approach to address these challenges is the generation of artificial images and their corresponding labels (Figure 5.14, Solution 2). This method offers several advantages compared to manual annotation. Firstly, it significantly reduces the manual effort required, as the need for expert annotation is eliminated. This can result in a more cost-effective solution, particularly when large datasets are required for training purposes. Additionally, the generation of artificial images enables the creation of tailored datasets that can include specific features, structures, or conditions relevant to the problem being addressed, that could not be obtained from annotated images, as there is no way to generate such corresponding image/labels pairs.

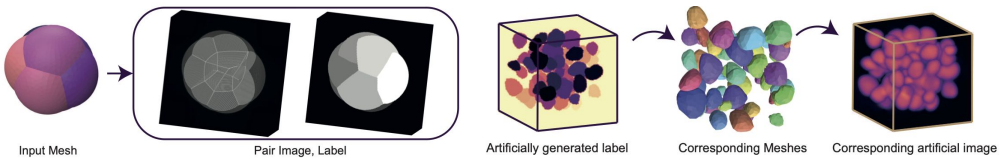


Figure 5.15: Several methodologies can be applied to create datasets: **Left**: starting from an input mesh, and generating a artificial image and label from it. **Right**: Generating artificially a label, constructing the corresponding mesh wiht Delaunay-Watershed 3D, and generating an image from the mesh.

There are two critical aspects that need to be addressed to fully leverage the potential of generating artificial images for neural network training: the automatic generation of labels and the enhancement of image photorealism. The Figure 5.15 displays two different methods proposed to generate datasets. The left image shows the process of starting from an input mesh, and generating an artificial image and label from it. On the other hand, the right image shows the process of generating an artificial label, constructing the corresponding mesh with Delaunay-Watershed 3D, and generating an image from the mesh. We will see in next section how to solve these two problems.

We also propose a graphical user interface (GUI) designed to facilitate the use of our proposed methodology by biologists (Figure 5.16). The GUI enables users to perform various operations in a more intuitive manner, promoting ease of use and accessibility. Among the features of the GUI are an adjustable bounding box, mass parallel GPU implementation in OpenCL, user-defined PSF, adjustable autofluorescence for closed surface meshes, and a hybrid tiff-mesh viewer. Additionally, the GUI provides the generation of semantic segmentation masks for all geometrical primitives, enabling the insertion of meshes of different dimensions and combining them to create realistic composite microscopy images.

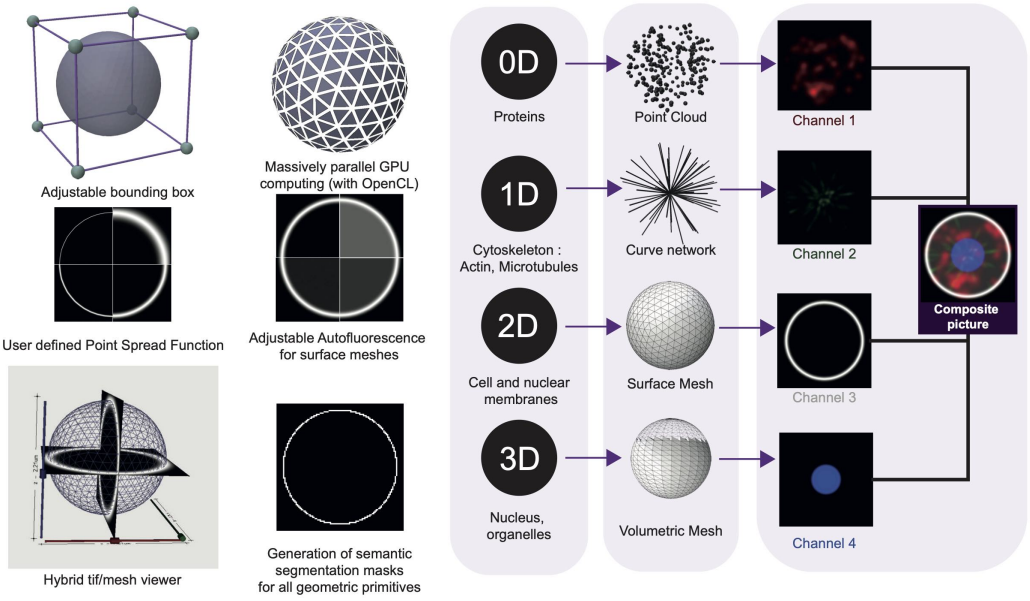


Figure 5.16: Our artificial image creator comes with a user friendly interface

5.5.1 Creation of volumetric masks from meshes in 3D

Creation of semantic segmentation mask

From a given mesh and a box, if one want to generate labels automatically, how should he proceed ? We proposed a simple method to do that: For each vertex of the mesh, find the nearest voxel, and put its value to one (Figure 5.17). The longest step is going to be the nearest neighbor search, which can be implemented very easily using a K-D Tree. However, this is only going to work with a grid that is more coarse than the mesh. More precisely, if the dimensions of a voxel in the (x, y, z) directions is (a, b, c), we need, for each edge of the mesh, to have its two vertices in a neighboring voxel, i.e we need each edge to be shorter than these dimensions a, b, c. To achieve this condition, we refine our mesh by subdividing the triangles

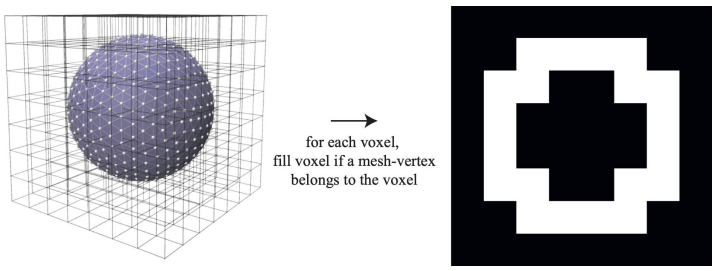


Figure 5.17: A simple heuristic allow to compute semantic segmentation masks on coarse grids

until every edge verifies this condition.
This gives the following algorithm:

```
## voxel size in (x,y,z) directions is (a,b,c)
## We start with a grid of voxels G = {M=0}
d_min = min(a,b,c)
def create_semantic_mask(G, mesh,dmin):
    while min(mesh edge length)>d_min:
        subdivide mesh
    for each vertex in mesh:
        find closest voxel M in G
        set M = 1
    return G
```

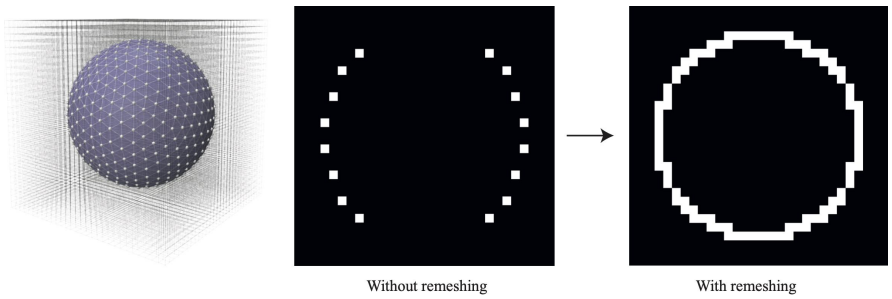


Figure 5.18: Remeshing allow to keep the same methodology

Creation of instance segmentation mask

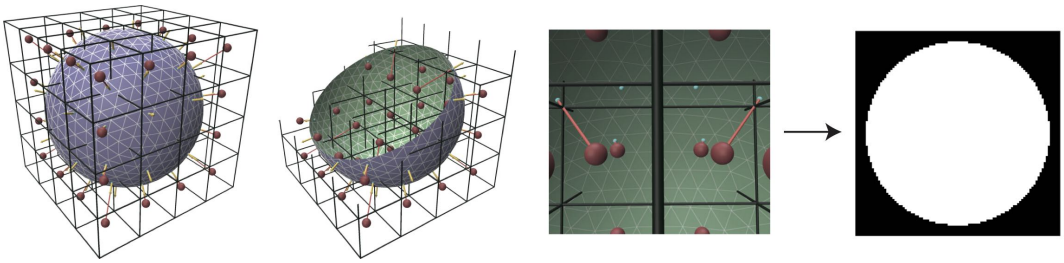


Figure 5.19: Normal orientations allows to check if each voxel is inside or outside the mesh

Now if want to create instance creation masks, i.e indicating with a label to which cell belongs each voxel, we need to use the fact that our mesh is a bounded volume, that defines an inside and an outside space. For each voxel M, we search the closest face centroid C. We call the normal of this face N. If $\vec{CM} \cdot \vec{N} > 0$, the voxel will be inside the volume. Otherwise it will

be outside. As multimaterial meshes describes several bounded volumes in contact with each other, this procedure can be directly applied for such structures.

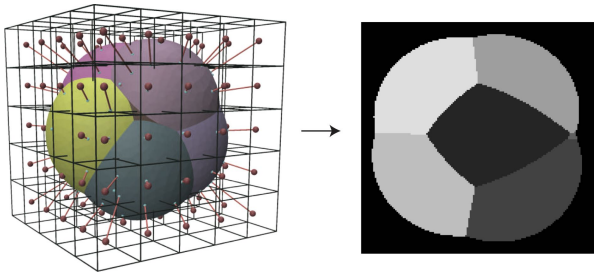


Figure 5.20: This methodology can be applied to multimaterial meshes as well

```
## voxel size in (x,y,z) directions is (a,b,c)
## We start with a grid of voxels G = {V=0}
d_min = min(a,b,c)
def create_instance_mask(G, mesh,dmin):
    while min(mesh edge length)>d_min:
        subdivide mesh
    for each triangle in mesh:
        C = compute_centroid(triangle)
        N = compute_normal(triangle)
        m_1, m_2 = return_label(triangle)
        find M in G, closest voxel to C
        if scalar_product(CM,N)>0:
            set M = m_1
        else :
            set M = m_2
    return G
```

5.5.2 Style Transfer

Image photorealism is a critical aspect when training neural networks, as it helps ensure that the model generalizes well to real microscopy data. Several methods can be explored. The idea is to modify the original artificial microscopy image, while keeping all the features intact, so our artificially generated labels would still correspond to the images.

The first solution is to apply camera noise, to reproduce the random noise observed on microscopy images, coming from the software Aydin [60] (Figure 5.21 up). The result is quite convincing, but such noise generation fails at reproducing the texture of the embryos, their imperfections and irregularities. Such notion of "texture", or "style", is in fact very complicated to define, and has been the object of many studies. Style transfer is a technique that combines the stylistic features of a source image with the content of a target image, resulting in a novel and visually coherent output. Early attempts to computationally model style transfer can be traced back to the work of [61], who introduced the concept of "image analogies" to synthesize textures from one image into another in order to emulate various artistic styles.

A significant advancement in the field was achieved with the introduction of neural style transfer by [62]. This method employs deep learning, specifically Convolutional Neural Networks, to analyze and replicate the style of a source image while preserving the content of a target image. The algorithm optimizes a loss function that balances content and style, effectively merging the two images into a visually appealing output.

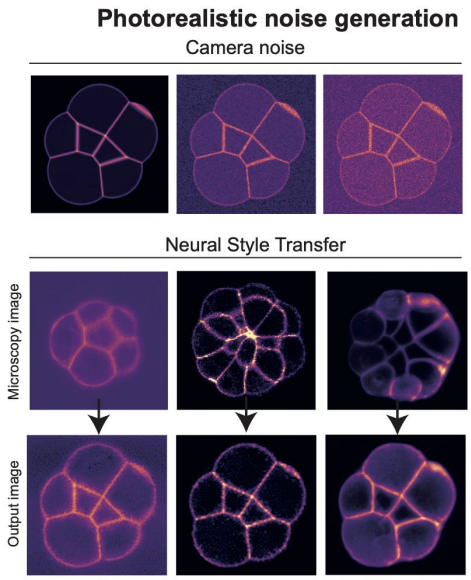


Figure 5.21: Several methods can be studied, along with PSF design, to ameliorate photorealism. Camera noise aims at mimicim the random noise of the camera. Neural style transfer is very good at reproducing the textures of other images, while keeping intact the original details of the image

Recently, diffusion-based methods have emerged as an alternative to neural style transfer. These approaches, such as those described by [63] and [64], utilize anisotropic diffusion, which smooths images while retaining essential details. By iteratively applying this process, diffusion-based style transfer generates a stylized image that emphasizes the aesthetic features of the source artwork. As in other fields using deep-learning based methods, progresses are very fast and never stops. We used the algorithm of [65], to show that we can very well reproduce different styles of imaging with it (Figure 5.21 down). It is very hard to keep track of all the progresses in the field, and thus we prefered not to loose too much time on this topic, in spite of very convicing results.

5.6 Inverse rendering of microscopy images

In spite of the power of neural networks, their expressivity also comes with drawbacks: CNNs are prone to feature "hallucinations" (i.e. detecting aberrant signal from noise), which is problematic for scientific research and medical applications, that cannot suffer from such errors. The goal of fluorescence microscopy is to identify one particular biological structure, which has generally a high degree of organization, such as lipid-membranes, cytoskeletal networks, or other organelles, by making them bright while keeping everything else dark. Therefore, except for unavoidable noise, fluorescence images are generally very sparse and made of well-structured objects such as points, filaments, surfaces or bulk solids. Because of this underlying simplicity, in parallel to the development of deep-learning based tools, one could witness in recent years a reverse trend in microscopy image analysis, going away from deep-learning, in favor of methods that rely on stronger prior knowledge [60, 66, 67].

Here, we demonstrate that differentiable rendering [68] stands out as a powerful approach to insert such priors for fluorescence image analysis: from defined parameters, such as parametrized shapes of objects and optical properties of the imaging system, one can approximate the image formation process. We introduce **deltaMic**, a differentiable renderer for 3D fluorescence

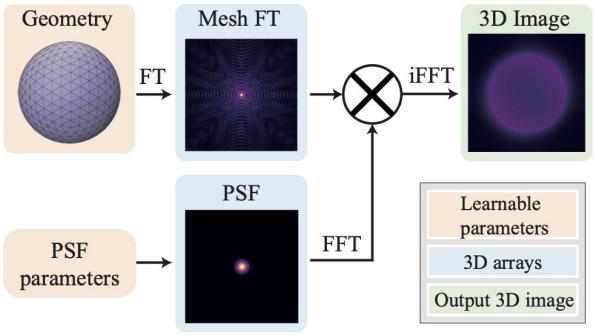


Figure 5.22: Overview of the differentiable fluorescence microscopy rendering pipeline. The renderer takes as input a triangle mesh, that represents the shape of a biological object, and a parameterized point spread function (PSF), that emulates the optics of the microscope, and outputs a 3D artificial microscopy image. By making every element of the pipeline differentiable, the gradients of the voxel image values with respect to the mesh geometry and to the PSF parameters can be computed efficiently with backpropagation. In turn, these gradients can be used to optimize input parameters to make the output image fit a real microscopy image.

microscopy images. Rendered images can be compared to true biological images using a voxel-based norm, and gradients of this loss with respect to input parameters can be determined efficiently with automatic differentiation. Minimizing this loss with a gradient-based optimization allows to fit the geometry of biological objects and to emulate the optical model of the microscope.

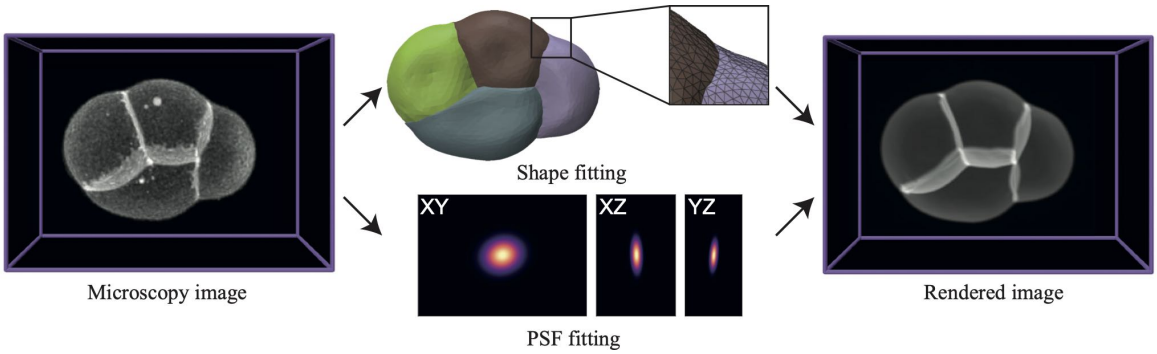


Figure 5.23: We apply our pipeline to learn both vertices position and PSF parameters from 3D confocal microscopy images.

We propose a narrow-band strategy to accelerate massively the computation of the mesh FT, and provide a custom C++/CUDA implementation of both the forward and the backward passes that greatly enhances speed and memory efficiency. Our PyTorch implementation allows for gradients of the rendered 3D image values with respect to input parameters to be computed through automatic differentiation. We demonstrate the inverse-rendering capabilities of our pipeline by reconstructing the PSF and the complex shape of various objects and of early embryos from artificial and true fluorescence images. Importantly, we rely on previously developed approaches in the field of differentiable rendering to regularize the optimization problem [69].

5.6.1 Experiments

Numerical implementation

Acceleration

The computation of the artificial image requires to compute the FT $\hat{u}_\Lambda$ for each of the N voxels of the spatial grid and for each of the n_t triangles, with a runtime complexity of $\mathcal{O}(\mathrm{N}{\cdot}n_t)$. The mesh FT gradient requires to compute a $\mathcal{O}(1)$ sum for each voxel, for each of the n_v vertices, leading to a complexity of $\mathcal{O}(\mathrm{N}_v)$. In practice, a 500^3 confocal microscopy picture has of 125 millions voxels, and a reasonable mesh will have $\approx 10^{3-4}$ vertices and triangles, leading to prohibitive computations times if this operation is not parallelized correctly. This high computational cost is a major restriction that has hampered the development of spectral methods in 3D, in spite of interesting potential applications [70].

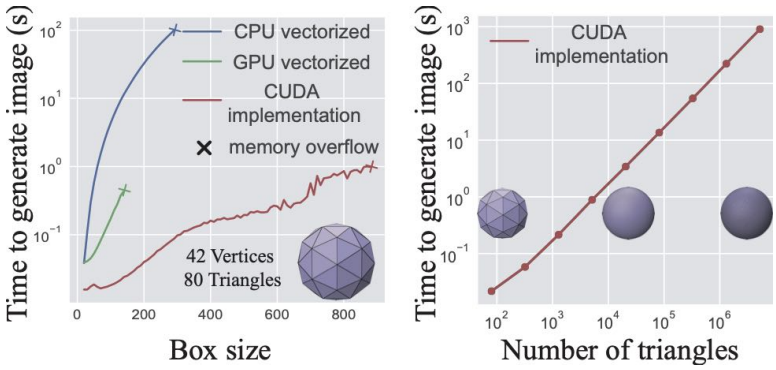


Figure 5.24: Benchmarking of the mesh FT computation. *Left* In a very small 80 triangles sphere, our custom CUDA implementation improves the speed and the memory efficiency of the mesh FT by several orders of magnitude. *Right* In a $[100]^3$ box, the forward pass of the mesh FT scales linearly with the number of triangles.

We propose two complementary strategies to accelerate the mesh FT: GPU parallelization and a narrow-band approximation method in the frequency domain.

GPU parallelization The grid-based structure of a 3D image makes its implementation naturally adapted to massivelly parallel computations on graphical processing units (GPU). We provide a custom CUDA implementation of the forward and the backward pass. As shown in figure 5.25, on a single NVIDIA V100 GPU, we obtain a $\approx 10^3$ speedup compared to the vectorized CPU implementation and we can perform computations with a far greater box size before overflow.

Narrow-band approximation in the frequency domain As stated previously, the FT of the image is expressed as the element-wise product between the FT of the mesh and of the PSF: $\hat{u}_\alpha = \hat{u}_\Lambda \cdot \hat{h}$. Practically, our PSF acts as a low-pass filter, that will give more weight to low spatial frequencies of the mesh. A blurred image has a sparse PSF, with largest amplitudes for frequencies close to 0, therefore obliterating the fine details of the mesh. The more resolved the original biological image, the more frequencies in the PSF are needed to properly render it.
For any given spatial frequency $\boldsymbol{\xi}$, if $\hat{h}(\boldsymbol{\xi}) \approx 0$, one can spare the computation of $\hat{u}_\Lambda(\boldsymbol{\xi})$ as it will have negligible impact on the value of $\hat{u}_\alpha(\boldsymbol{\xi})$. In practice, we apply a cutoff in the frequency domain, and only compute the FT of the mesh for the frequencies for which the

PSF is larger than 1% of its maximal value. For sparse PSFs, this spectral narrow-band method can reduce the computational cost by several orders of magnitude (table 5.1).

A staggered optimization scheme

In spite of all the speed improvements, computing the FT of the mesh and its gradient constitutes the computational bottlenecks of our pipeline. We have two ways of computing the FT of the mesh: a fast and approximate way using the narrow-band method, and a slow and exact way by doing the full computation. The backward path is ≈ 10 times slower than the forward path, it is thus crucial to compute the backward path with the narrow-band method when possible. We decide therefore to decompose the coupled optimization problem into two optimization sub-problems, that are iterated in a staggered manner:

(a) Shape optimization: To learn the shape, one does only need to compute the mesh FT approximately. We compute the approximate FT of the mesh using the narrow-band method, and make an optimization step for the vertex positions, without optimizing the PSF.

(b) PSF optimization: To learn the point spread function, one need to know the value of the FT of the mesh in all the available frequencies, and should not be biased by the narrow-band frequency threshold that relies on the current values of the PSF. We compute the exact FT of the mesh, and make an optimization step for the PSF parameters, without optimizing the shape.

Shape optimization step

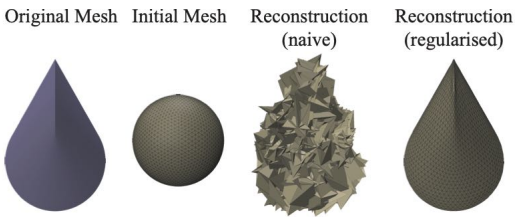


Figure 5.25: Without regularisation, the gradient descent fails to converge on the original shape

We optimize shapes by minimizing a modified L2 norm $\Phi_\kappa = ||u - m||^2 * m^\kappa$ with $\kappa = 1$, that removes external influence from dark regions of the original image m. Inspired by other works in differentiable rendering [69], we optimize vertex positions using AdamUniform and regularize gradient values by adding a diffusion term:

$$\mathbf{v} \leftarrow x - \eta(\mathbf{I} + \lambda\mathbf{L})^{-2}\frac{\partial\Phi_1}{\partial\mathbf{v}}, \tag{5.26}$$

where the Laplacian $\mathbf{L} \in \mathbb{R}^{n_v \times n_v}$ is a connectivity Laplacian defined on our triangle mesh, and $\lambda = 50$. Computing explicitly the inverse $(\mathbf{I} + \lambda\mathbf{L})^{-2}$ requires to use dense algebra, which can lead to prohibitive memory usage when the number of vertices n_v of the mesh grows. Instead, we solve the equivalent problem of finding the solution A to $(\mathbf{I} + \lambda\mathbf{L})^{-2}\mathrm{A} = \frac{\partial\Phi_1}{\partial\mathbf{v}}$ with a sparse Cholesky decomposition [69, 71].

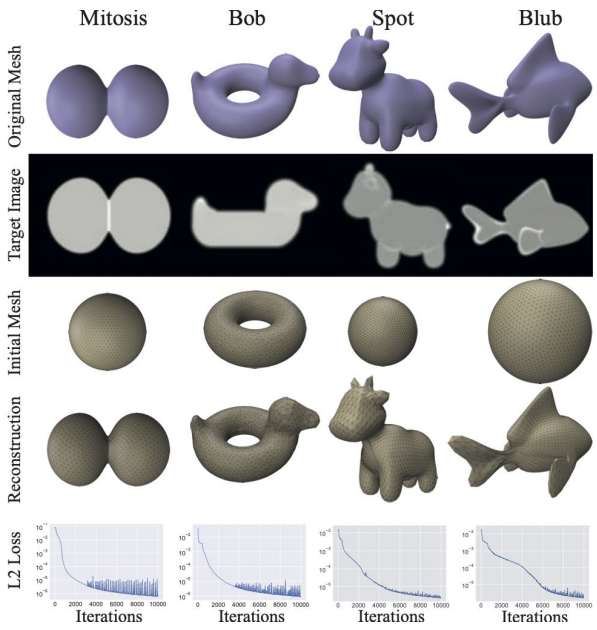


Figure 5.26: Surface mesh reconstruction from 3D artificial images. Artificial 3D images were created from triangle meshes using our renderer. Then, starting from spheres or toruses, we minimize the L2 norm between the rendered and the target images to reconstruct the original shape.

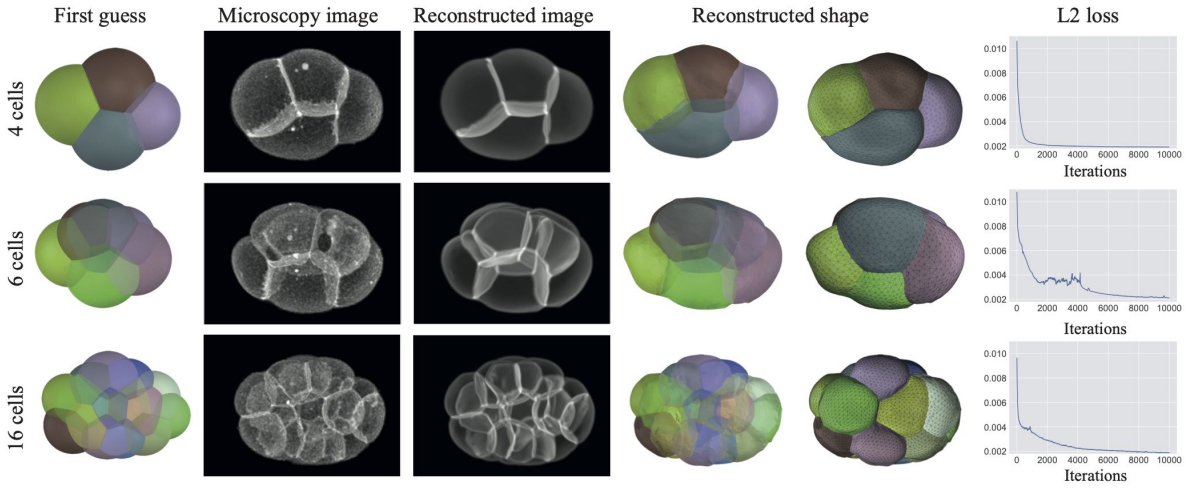


Figure 5.27: Shape reconstruction of *C. elegans* embryos. Starting from a foam ground truth, we optimize both vertices position and PSF parameters to recover individual cell shapes from real confocal microscopy data.

PSF optimization step

The optimization of PSF parameters aims at reproducing the whole image in details, for both light and dark regions. Minimizing the traditional L2 norm $\phi_0 = ||u - m||^2$ is therefore well-adapted to such task. We learn the correlation matrix Σ with the original Adam optimizer [72].

Shape reconstruction from artificial images

We present in fig. 5.26 several examples of shape reconstruction from some classical meshes in computer graphics. Artificial images are generated with a Gaussian PSF characterized by an isotropic correlation matrix. Here, we focus on shape reconstruction and start directly

Optimization step	forward (s)	backward (s)
PSF	0.001	0.004
Mesh (naive)	3.435	49.38
Mesh (narrow-band)	**1.757**	**17.04**

Table 5.1: Average duration of the PSF and mesh optimization steps for the reconstruction described in section 5.6.1 with a naive implementation (naive) and with the spectral narrow-band approximation. The mesh has 4000 vertices.

from the target PSF correlation matrix Σ. Starting from meshes of elementary shapes (a sphere or a torus depending on the topology) we compute 10000 optimization steps for each example. Without any remeshing operation, our regularized shape optimization algorithm accurately converges to original shapes. Interestingly, in each of the four examples studied, the mesh converges with no collisions, although no collision detection or resolution method is implemented [73, 74].

Cell shape reconstruction and Gaussian PSF fitting on *C. elegans* embryo microscopy images

We use our method to fit shapes of cell clusters of *C. elegans* embryos, using 3D confocal microscopy images from [75]. The images are of size $[205 \times 285 \times 134]$. We describe cell clusters as non-manifold multimaterial meshes, as done in previous works [76], and we use a multimaterial mesh-based surface tracking method [77] to manage remeshing, collision detection and T1/T2 topology transition [78] operations. As first guesses, we generate foam-like multimaterial meshes with the correct number of cells by minimizing a surface energy at fixed cell volumes [54]. The resulting cell clusters are of roundish shapes with $\approx$ 4000 - 10000 vertices that are subsequently optimized to fit true microscopy data. Here, we optimize jointly the PSF and the vertex positions in the mesh, following the strategy described in section 5.6.1. The duration of each step with a mesh of 4000 vertices is displayed in table 5.1. On both 4, 6 and 16 cells embryos, we predict individual cell shapes accurately after 10000 optimization steps, and the resulting reconstructed image is very close to the original microscopy image (fig. 5.27).

5.6.2 Extension : 2D Differentiable Rendering

The 2D case, where we consider a 1D curve embedded in 2D, is simpler to tackle than the 3D problem. A thourough study of this case gives a lot of insight about the difficulties encountered in 3D. We will study the discrete example of a polygonal curve embedded in 2D, that can be multimaterial or not. In two dimensions, the formulas and their derivation are close to the one in 3D. We do not give explicit formulas here and refer the reader to the 3D derivation.

Diffusion

As in 3D, we modify the gradient descent by adding a diffusion term:

$$\mathbf{x} \leftarrow x - \eta(\mathbf{I} + \lambda \mathbf{L})^{-2} \frac{\partial \Phi}{\partial \mathbf{x}} \tag{5.27}$$

Where $\mathbf{L} \in \mathbb{R}^{n_v \times n_v}$ is the graph connectivity Laplacian:

$$\mathbf{L}_{i,j} := \begin{cases} \deg(v_i) & \text{if } i = j \\ -1 & \text{if } i \neq j \text{ and } v_i \text{ is adjacent to } v_j \\ 0 & \text{otherwise,} \end{cases} \tag{5.28}$$

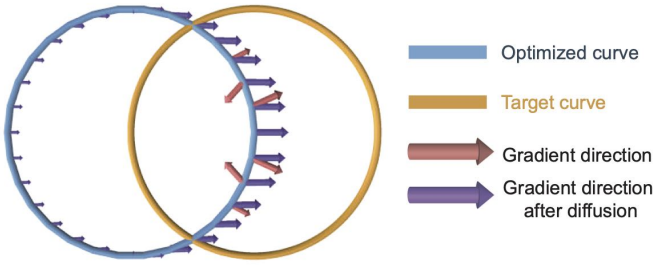


Figure 5.28: The diffusion of the gradient helps regularizing the gradient descent, without adding any constraint in the loss optimized

On our simple example, the gradient descent is smoothed by the gradient diffusion, which helps avoiding mesh intersections, and allows for larger steps at a small memory and speed cost.

Point Spread Functions in 2D

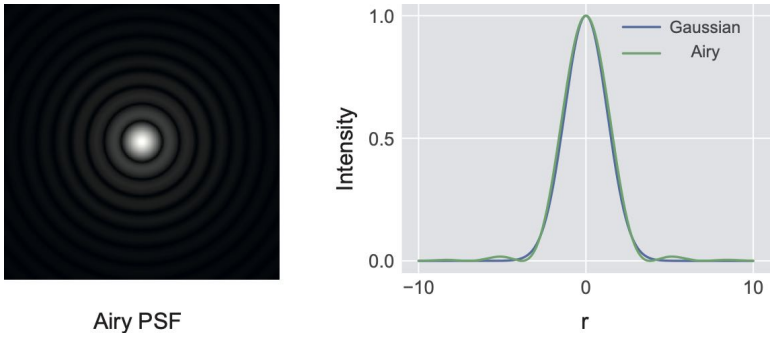


Figure 5.29: Gaussian VS Airy PSF. In 2D, most PSFs gives the same results. We can thus use the simple Gaussian PSF.

In 2D, the best spot of light that can be generated by a lens with a circular aperture is called an Airy disk. Its normalized intensity does only depend to the distance to its central point r and is defined as $h_{airy}(r) = \left(\frac{2J_1(r)}{r}\right)^2$, where J_1 is the bessel function of first kind or order 1. For any parameters, it is hard to distinguish, in practice, between such an Airy pattern and a Gaussian function. We will thus do our analysis in the following with a Gaussian PSF, and learn the covariance matrix $\Sigma \in \mathbb{R}^{2\times 2}$.

Rendering a polygonal line

The shape considered is a closed polygonal line $u_\lambda = (V, E)$ described by the set of its vertices and of its edges. The polyline will be converted into an image u_α, and the position of its vertices will evolve to minimize the objective function ϕ defined as the L2 norm between u_α and the true microscopy image: $\phi(u_\alpha) = |u_\alpha - m|^2$.

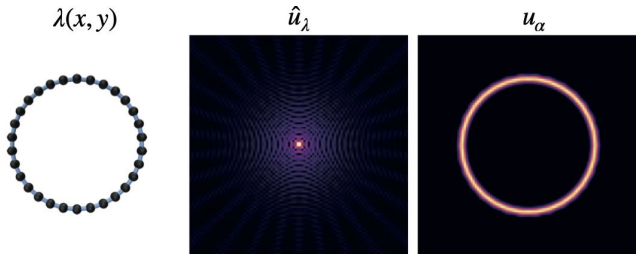


Figure 5.30: We can, as with triangle meshes, render polygonal lines into microscopy images

Results: In-Silico

We study the very simple example of finding the shape of an ellipse, starting from a circle. For example, the circle could model the boundaries of a single cell observed with a given plane from a fluorescence microscope. Our iterative procedure updates the position of the vertices using VectorAdam [79], progressively modifying the shape to modify the image, making it closer and closer to the ground truth, eventually converging to the original shape after 1500 steps.

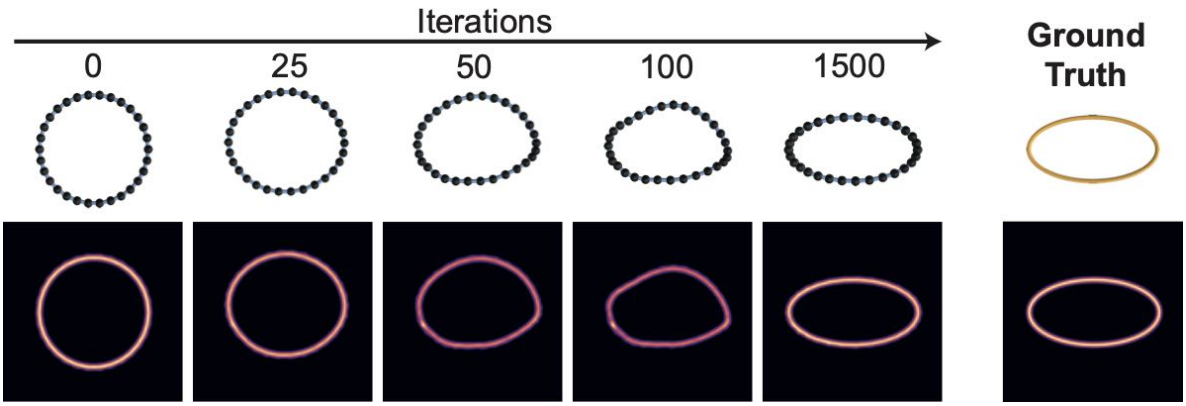


Figure 5.31: Fitting the circles on an ellipse works nicely

5.6.3 Summary

We have introduced **deltaMIC**, a differentiable fluorescence microscopy image renderer that takes a surface mesh and a parametrized PSF to generate realistic 3D confocal fluorescent images. The GPU-parallelized C++/CUDA implementation of the mesh FT allows to compute the forward and backward passes in tractable times, and is further optimized through a spectral narrow-band method. We demonstrate that our differentiable renderer can accurately retrieve fine details of complex shapes from artificial or real 3D microscopy fluorescent images, thanks to its ability to fine-tune the shape loss function via the PSF. Contrary to other active-mesh segmentation methods, it does not require arbitrary shape regularization terms to achieve smooth results. Yet, other priors may be added to constrain shapes, based on physical or geometric knowledge.

Our method constitutes a fundamental building block for inverse rendering of 3D fluorescence microscopy images and paves the way for many applications. Extending our approach to 1D polygonal lines would allow to characterize the geometry of filaments in cytoskeletal networks, whose segmentation remain a major challenge [80]. Furthermore, our method shares several

insights with deconvolution techniques for microscopy, which are fundamentally based on inverse image rendering and optimization [81]. Implementing more photorealistic PSF models based on specific microscope modalities [14,15,82] would allow performing blind deconvolution of microscopy images in an original manner. From an implementation perspective, and in spite of all the improvements proposed, the computational cost of our methoxd remains a bottleneck for large 3D images, and other acceleration strategies should be explored. Eventually, it could be interesting to explore deep-learning based applications, where our renderer would be a differentiable building block of a larger pipeline, as in several 2D applications [51, 83].

5.7 Conclusion

In conclusion, this work has presented a comprehensive approach for bridging the gap between models and data in fluorescence microscopy by developing a differentiable artificial microscope. We have introduced an image formation model and a rendering process for fluorescence microscopy images, laying a solid foundation for generating realistic images. Moreover, we have explored methods to define geometrical models of biological objects, establishing the Fourier Transform of triangulated surfaces and its gradients.

A key application of our renderer involves the systematic investigation of various Point Spread Function (PSF) models, including Gaussian PSF, Gibson and Lanni model, pupil-function-based model, and confocal microscopy PSFs. Evaluating these models has yielded valuable insights into their strengths and limitations. Subsequently, we have delved into super-resolution microscopy in images, examining structured illumination microscopy, Wiener SIM, and Lattice SIM, in light of our image creation method.

We have also provided methods for creating artificial datasets, introducing novel techniques for generating volumetric masks from 3D meshes, and enhancing the photorealism of images through neural style transfer. This contributes to the development of high-quality datasets for validating computer vision algorithms and training computer vision models. Our use of inverse rendering in microscopy images has demonstrated the potential for accurately reconstructing complex shapes from both artificial and real 3D microscopy images. The extension to 2D differentiable rendering presents additional opportunities for future applications and research.

Physicists develop 3D models of biological objects, that we aimed to merge with real microscopy data. Throughout this ambitious journey, we realized that a novel approach would be to create an artificial fluorescence microscope. Our framework encompasses a wide range of biological objects and PSFs. Our inverse rendering approach enables researchers to extract more accurate and detailed information from biological images, ultimately contributing to a deeper understanding of complex biological systems and paving the way for new discoveries in the life sciences.

Bibliography

[1] Jeff W Lichtman and José-Angel Conchello. Fluorescence microscopy. *Nature methods*, 2(12):910–919, 2005.

[2] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells,

and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing.

[3] Özgün Çiçek, Ahmed Abdulkadir, Soeren S. Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: Learning dense volumetric segmentation from sparse annotation. In Sebastien Ourselin, Leo Joskowicz, Mert R. Sabuncu, Gozde Unal, and William Wells, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*, pages 424–432, Cham, 2016. Springer International Publishing.

[4] Chinmay Belthangady and Loic A Royer. Applications, promises, and pitfalls of deep learning for fluorescence image reconstruction. *Nature methods*, 16(12):1215–1225, 2019.

[5] Schmidt U. Boothe T. et al. Weigert, M. Content-aware image restoration: pushing the limits of fluorescence microscopy. *Nat Methods*, 15:1090–1097, 2018.

[6] Uwe Schmidt, Martin Weigert, Coleman Broaddus, and Gene Myers. Cell detection with star-convex polygons. In *Medical Image Computing and Computer Assisted Intervention - MICCAI 2018 - 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part II*, pages 265–273, 2018.

[7] Carsen Stringer, Tim Wang, Michalis Michaelos, and Marius Pachitariu. Cellpose: a generalist algorithm for cellular segmentation. *Nature methods*, 18(1):100–106, 2021.

[8] Hirofumi Kobayashi, Keith C Cheveralls, Manuel D Leonetti, and Loic A Royer. Self-supervised deep learning encodes high-resolution features of protein subcellular localization. *Nature methods*, 19(8):995–1003, 2022.

[9] Chiyu "Max" Jiang, Dequan Wang, Jingwei Huang, Philip Marcus, and Matthias Nießner. Convolutional neural networks on non-uniform geometrical signals using euclidean spectral transformation. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[10] Shung-Wu Lee and R. Mittra. Fourier transform of a polygonal shape function and its application in electromagnetics. *IEEE Transactions on Antennas and Propagation*, 31(1):99–103, 1983.

[11] Shinya Inoué. *Foundations of Confocal Scanned Imaging in Light Microscopy*, pages 1–19. Springer US, Boston, MA, 2006.

[12] Sarah Frisken Gibson and Frederick Lanni. Experimental test of an analytical model of aberration in an oil-immersion objective lens used in three-dimensional light microscopy. *J. Opt. Soc. Am. A*, 9(1):154–166, Jan 1992.

[13] Hal G. Kraus. Huygens–fresnel–kirchhoff wave-front diffraction formulation: spherical waves. *J. Opt. Soc. Am. A*, 6(8):1196–1205, Aug 1989.

[14] Bridget M. Hanser, Mats G. L. Gustafsson, David A. Agard, and John W. Sedat. Phase retrieval for high-numerical-aperture optical systems. *Opt. Lett.*, 28(10):801–803, May 2003.

[15] B. M. Hanser, M. G. L. Gustafsson, D. A. Agard, and J. W. Sedat. Phase-retrieved pupil functions in wide-field fluorescence microscopy. *Journal of Microscopy*, 216(1):32–48, 2004.

[16] Matthew R. Arnison and Colin J.R. Sheppard. A 3d vectorial optical transfer function suitable for arbitrary pupil functions. *Optics Communications*, 211(1):53–63, 2002.

[17] Daniel Sage, Lauréne Donati, Ferréol Soulez, Denis Fortun, Guillaume Schmit, Arne Seitz, Romain Guiet, Cédric Vonesch, and Michael Unser. Deconvolutionlab2: An open-source software for deconvolution microscopy. *Methods*, 115:28–41, 2017.

[18] Florian Luisier, Thierry Blu, and Michael Unser. Image denoising in mixed poisson–gaussian noise. *IEEE Transactions on Image Processing*, 20(3):696–708, 2011.

[19] Jizhou Li, Feng Xue, and Thierry Blu. Fast and accurate three-dimensional point spread function computation for fluorescence microscopy. *J. Opt. Soc. Am. A*, 34(6):1029–1034, Jun 2017.

[20] H. Kirshner, F. Aguet, D. Sage, and M. Unser. 3-d psf fitting for fluorescence microscopy: implementation and localization application. *Journal of Microscopy*, 249(1):13–25, 2013.

[21] Ferréol Soulez, Loïc Denis, Yves Tourneur, and Éric Thiébaut. Blind deconvolution of 3d data in wide field fluorescence microscopy. In *2012 9th IEEE International Symposium on Biomedical Imaging (ISBI)*, pages 1735–1738, 2012.

[22] Satwik Rajaram, Benjamin Pavie, Nicholas EF Hac, Steven J Altschuler, and Lani F Wu. Simucell: a flexible framework for creating synthetic microscopy images. *Nature methods*, 9(7):634–635, 2012.

[23] Leonid Mill, David Wolff, Nele Gerrits, Patrick Philipp, Lasse Kling, Florian Vollnhals, Andrew Ignatenko, Christian Jaremenko, Yixing Huang, Olivier De Castro, et al. Synthetic image rendering solves annotation problem in deep learning nanoparticle segmentation. *Small Methods*, 5(7):2100223, 2021.

[24] Serge Dmitrieff and François Nédélec. Confocalgn: A minimalistic confocal image generator. *SoftwareX*, 6:243–247, 2017.

[25] Patrik Malm, Anders Brun, and Ewert Bengtsson. Simulation of bright-field microscopy images depicting pap-smear specimen. *Cytometry Part A*, 87(3):212–226, 2015.

[26] Veit Wiesmann, Matthias Bergler, Ralf Palmisano, Martin Prinzen, Daniela Franz, and Thomas Wittenberg. Using simulated fluorescence cell micrographs for the evaluation of cell image segmentation algorithms. *BMC bioinformatics*, 18(1):1–12, 2017.

[27] Reka Hollandi, Abel Szkalisity, Timea Toth, Ervin Tasnadi, Csaba Molnar, Botond Mathe, Istvan Grexa, Jozsef Molnar, Arpad Balind, Mate Gorbe, Maria Kovacs, Ede Migh, Allen Goodman, Tamas Balassa, Krisztian Koos, Wenyu Wang, Juan Carlos Caicedo, Norbert Bara, Ferenc Kovacs, Lassi Paavolainen, Tivadar Danka, Andras Kriston, Anne Elizabeth Carpenter, Kevin Smith, and Peter Horvath. nucleaizer: A parameter-free deep learning framework for nucleus segmentation using image style transfer. *Cell Systems*, 10(5):453–458.e6, 2020.

[28] Dennis Eschweiler, Malte Rethwisch, Mareike Jarchow, Simon Koppers, and Johannes Stegmaier. 3d fluorescence microscopy data synthesis for segmentation and benchmarking. *Plos one*, 16(12):e0260509, 2021.

[29] Erik Meijering. Cell segmentation: 50 years down the road [life sciences]. *IEEE Signal Processing Magazine*, 29(5):140–145, 2012.

[30] Serge Beucher. The watershed transformation applied to image segmentation. *Scanning Microscopy*, 1992(6):28, 1992.

[31] Serge Beucher and Fernand Meyer. The morphological approach to segmentation: the watershed transformation. In *Mathematical morphology in image processing*, pages 433–481. CRC Press, 2018.

[32] Romain Fernandez, Pradeep Das, Vincent Mirabet, Eric Moscardi, Jan Traas, Jean-Luc Verdeil, Grégoire Malandain, and Christophe Godin. Imaging plant growth in 4d: robust tissue reconstruction and lineaging at cell resolution. *Nature methods*, 7(7):547–553, 2010.

[33] Yuri Boykov and Gareth Funka-Lea. Graph cuts and efficient nd image segmentation. *International journal of computer vision*, 70(2), 2006.

[34] Kevin J Cutler, Carsen Stringer, Teresa W Lo, Luca Rappez, Nicholas Stroustrup, S Brook Peterson, Paul A Wiggins, and Joseph D Mougous. Omnipose: a high-precision morphology-independent solution for bacterial cell segmentation. *Nature methods*, 19(11):1438–1448, 2022.

[35] David Bryant Mumford and Jayant Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on pure and applied mathematics*, 1989.

[36] Ravi Malladi, James A Sethian, and Baba C Vemuri. Shape modeling with front propagation: A level set approach. *IEEE transactions on pattern analysis and machine intelligence*, 17(2):158–175, 1995.

[37] T.F. Chan and L.A. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–277, 2001.

[38] Michael Kass, Andrew P. Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *Int. J. Comput. Vis.*, 1(4):321–331, 1988.

[39] Alexandre Dufour, Roman Thibeaux, Elisabeth Labruyere, Nancy Guillen, and Jean-Christophe Olivo-Marin. 3-d active meshes: fast discrete deformable models for cell tracking in 3-d time-lapse microscopy. *IEEE transactions on image processing*, 20(7):1925–1937, 2010.

[40] Mercier-V. Chiaruttini N. Machado, S. Limeseg: a coarse-grained lipid membrane simulation for 3d image segmentation. *BMC Bioinformatics*, 20, 2019.

[41] Matthew B Smith, Hugh Sparks, Jorge Almagro, Agathe Chaigne, Axel Behrens, Chris Dunsby, and Guillaume Salbreux. Active mesh and neural network pipeline for cell aggregate segmentation. *Biophysical Journal*, 122(9):1586–1599, 2023.

[42] Wim Thiels, Bart Smeets, Maxim Cuvelier, Francesca Caroti, and Rob Jelier. spheresDT/Mpacts-PiCS: cell tracking and shape retrieval in membrane-labeled embryos. *Bioinformatics*, 37(24):4851–4856, 07 2021.

[43] Jo A. Helmuth and Ivo F. Sbalzarini. Deconvolving active contours for fluorescence microscopy images. In George Bebis, Richard Boyle, Bahram Parvin, Darko Koracin, Yoshinori Kuno, Junxian Wang, Jun-Xuan Wang, Junxian Wang, Renato Pajarola, Peter Lindstrom, André Hinkenjann, Miguel L. Encarnação, Cláudio T. Silva, and Daniel Coming, editors, *Advances in Visual Computing*, pages 544–553, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[44] Jeffrey P. Nguyen, Benjamin P. Bratton, and Joshua W. Shaevitz. *Biophysical Measurements of Bacterial Cell Shape*, pages 227–245. Springer New York, New York, NY, 2016.

[45] Tzu-Mao Li, Michal Lukáč, Michaël Gharbi, and Jonathan Ragan-Kelley. Differentiable vector graphics rasterization for editing and learning. *ACM Trans. Graph.*, 39(6), nov 2020.

[46] Merlin Nimier-David, Delio Vicini, Tizian Zeltner, and Wenzel Jakob. Mitsuba 2: A retargetable forward and inverse renderer. *ACM Trans. Graph.*, 38(6), nov 2019.

[47] Hiroharu Kato, Deniz Beker, Mihai Morariu, Takahiro Ando, Toru Matsuoka, Wadim Kehl, and Adrien Gaidon. Differentiable rendering: A survey. *CoRR*, abs/2006.12057, 2020.

[48] Nikhila Ravi, Jeremy Reizenstein, David Novotný, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. Accelerating 3d deep learning with pytorch3d. *CoRR*, abs/2007.08501, 2020.

[49] Shichen Liu, Tianye Li, Weikai Chen, and Hao Li. Soft rasterizer: A differentiable renderer for image-based 3d reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[50] Samuli Laine, Janne Hellsten, Tero Karras, Yeongho Seol, Jaakko Lehtinen, and Timo Aila. Modular primitives for high-performance differentiable rendering. *ACM Trans. Graph.*, 39(6), nov 2020.

[51] Hiroharu Kato, Yoshitaka Ushiku, and Tatsuya Harada. Neural 3d mesh renderer. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 3907–3916. Computer Vision Foundation / IEEE Computer Society, 2018.

[52] 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*. Curran Associates Inc., Red Hook, NY, USA, 2019.

[53] Roy Frostig, Matthew Johnson, and Chris Leary. Compiling machine learning programs via high-level tracing. 2018.

[54] Jean-Léon Maître, Hervé Turlier, Rukshala Illukkumbura, Björn Eismann, Ritsuya Niwayama, François Nédélec, and Takashi Hiiragi. Asymmetric division of contractile domains couples cell positioning and fate specification. *Nature*, 536(7616):344–348, 2016.

[55] Max Born and Emil Wolf. *Principles of optics: electromagnetic theory of propagation, interference and diffraction of light*. Elsevier, 2013.

[56] Mats GL Gustafsson. Surpassing the lateral resolution limit by a factor of two using structured illumination microscopy. *Journal of microscopy*, 198(2):82–87, 2000.

[57] Amit Lal, Chunyan Shan, and Peng Xi. Structured illumination microscopy image reconstruction algorithm. *IEEE Journal of Selected Topics in Quantum Electronics*, 22(4):50–63, 2016.

[58] Bi-Chang Chen, Wesley R Legant, Kai Wang, Lin Shao, Daniel E Milkie, Michael W Davidson, Chris Janetopoulos, Xufeng S Wu, John A Hammer III, Zhe Liu, et al. Lattice light-sheet microscopy: imaging molecules to embryos at high spatiotemporal resolution. *Science*, 346(6208):1257998, 2014.

[59] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2016: 19th International Conference, Athens, Greece, October 17-21, 2016, Proceedings, Part II 19*, pages 424–432. Springer, 2016.

[60] Ahmet Can Solak, Loic A. Royer, Abdur-Rahmaan Janhangeer, and Hirofumi Kobayashi. royerlab/aydin: v0.1.15, October 2022.

[61] Aaron Hertzmann, Charles E Jacobs, Nuria Oliver, Brian Curless, and David H Salesin. Image analogies. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 327–340, 2001.

[62] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015.

[63] Manuel Ruder, Alexey Dosovitskiy, and Thomas Brox. Artistic style transfer for videos. In *German Conference on Pattern Recognition*, pages 26–36. Springer, Cham, 2016.

[64] Artsiom Sanakoyeu, Dmytro Kotovenko, Sabine Lang, and Bj"orn Ommer. A style-aware content loss for real-time hd style transfer. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 698–714, 2018.

[65] Nicholas Kolkin, Jason Salavon, and Gregory Shakhnarovich. Style transfer by relaxed optimal transport and self-similarity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10051–10060, 2019.

[66] Joshua Batson and Loïc Royer. Noise2self: Blind denoising by self-supervision. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 524–533. PMLR, 2019.

[67] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2noise: Learning image restoration without clean data. In Jennifer G. Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 2971–2980. PMLR, 2018.

[68] Hiroharu Kato, Deniz Beker, Mihai Morariu, Takahiro Ando, Toru Matsuoka, Wadim Kehl, and Adrien Gaidon. Differentiable rendering: A survey, 2020.

[69] Baptiste Nicolet, Alec Jacobson, and Wenzel Jakob. Large steps in inverse rendering of geometry. *ACM Trans. Graph.*, 40(6), dec 2021.

[70] Chiyu Jiang, Dequan Wang, Jingwei Huang, Philip Marcus, Matthias Nießner, et al. Convolutional neural networks on non-uniform geometrical signals using euclidean spectral transformation. *arXiv preprint arXiv:1901.02070*, 2019.

[71] Maxim Naumov. Parallel solution of sparse triangular linear systems in the preconditioned iterative methods on the gpu. *NVIDIA Corp., Westford, MA, USA, Tech. Rep. NVR-2011*, 1, 2011.

[72] DP Kingma. Adam: a method for stochastic optimization. In *Int Conf Learn Represent*, 2014.

[73] Tyson Brochu, Essex Edwards, and Robert Bridson. Efficient geometrically exact continuous collision detection. *ACM Transactions on Graphics (TOG)*, 31(4):1–7, 2012.

[74] Ming C Lin, Dinesh Manocha, and Young J Kim. Collision and proximity queries. In *Handbook of discrete and computational geometry*, pages 1029–1056. Chapman and Hall/CRC, 2017.

[75] Jianfeng Cao, Guoye Guan, Vincy Wing Sze Ho, Ming-Kin Wong, Lu-Yan Chan, Chao Tang, Zhongying Zhao, and Hong Yan. Establishment of a morphological atlas of the caenorhabditis elegans embryo using deep-learning-based 4d segmentation. *Nature communications*, 11(1):1–14, 2020.

[76] Kenneth A Brakke. The surface evolver. *Experimental mathematics*, 1(2):141–165, 1992.

[77] Fang Da, Christopher Batty, and Eitan Grinspun. Multimaterial mesh-based surface tracking. *ACM Trans. Graph.*, 33(4):112–1, 2014.

[78] Denis L Weaire and Stefan Hutzler. *The physics of foams*. Oxford University Press, 2001.

[79] Selena Ling, Nicholas Sharp, and Alec Jacobson. Vectoradam for rotation equivariant geometry optimization. *CoRR*, abs/2205.13599, 2022.

[80] Bugra Özdemir and Ralf Reski. Automated and semi-automated enhancement, segmentation and tracing of cytoskeletal networks in microscopic images: A review. *Computational and Structural Biotechnology Journal*, 19:2106–2120, 2021.

[81] Pinaki Sarder and Arye Nehorai. Deconvolution methods for 3-d fluorescence microscopy images. *IEEE signal processing magazine*, 23(3):32–45, 2006.

[82] Vasudevan Lakshminarayanan and Andre Fleck. Zernike polynomials: a guide. *Journal of Modern Optics*, 58(7):545–561, 2011.

[83] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *CoRR*, abs/2003.08934, 2020.

Chapter 6

Inference of flows from kymographs with optimal transport

Written in collaboration with Hervé Turlier. Experiments were performed by Henry de Belly, Shannon Yan, supervised by Carlos Bustamante and Orion D. Weiner.

The application of this work is present in a publication in the peer-reviewed journal Cell. Another paper presenting the method is currently in preparation.

This project started as a side project where I developped a data analysis pipeline for a cell biology project. We wanted to determine the intensity of membrane flows. We wanted to apply traditionnal methods to measure flows but all of them were unsufficient. I developped this new method to measure membrane flows using the classical methodology of formulating a forward problem explaining the data, to learn from the data by solving the inverse problem. This collaboration worked very well, and was very importantly a way to discover optogenetics, which is one of the best methods to study biological systems in a reproducible manner.

Methods to measure membrane flows in an accurate and quantitative manner are key to understand the complex dynamics of biophysical processes taking place in cells. We introduce a novel method for the inference of membrane flows from kymographs, that adresses some limitations encountered by current methods such as Particle Image Velocimetry (PIV) and Particle Tracking Velocimetry (PTV). Our method links particle distribution and velocity fields through a flow model that can then be inverted using optimal transport theory. This approach allow to infer flows in scenarios where tracing individual particle trajectories is unfeasible or inefficient, using kymographs as a coarse-grained representation of particle distribution from which flows can be inferred. We validate our algorithm using artificially generated data, and apply it to real microscopy data of optogenetically activable cells that generate protrusions and consequently, repeatable flows. The inferred flows, matches both qualitatively and quantitatively data obtained with FRAP experiments, and provides a finer spatial resolution.

Chapitre 6

Inférence d'écoulements à partir de kymographies avec transport optimal

Écrit en collaboration avec Hervé Turlier. Les expériences ont été réalisées par Henry de Belly, Shannon Yan, sous la supervision de Carlos Bustamante et Orion D. Weiner.

L'application de ce travail est présente dans une publication dans la revue scientifique à comité de lecture Cell. Un autre article présentant la méthode est actuellement en préparation. Ce projet a commencé comme un projet annexe où j'ai développé une pipeline d'analyse de données pour un projet de biologie cellulaire. Nous voulions déterminer l'intensité des flux de membrane. Nous souhaitions initialement appliquer des méthodes classiques pour mesurer ces flux, mais toutes étaient insuffisantes. J'ai développé cette nouvelle méthode pour mesurer les flux de membrane en formulant un problème direct expliquant les données, pour ensuite apprendre des données en résolvant le problème inverse. Cette collaboration a très bien fonctionné et a été un excellent moyen de découvrir l'optogénétique, qui est l'une des meilleures méthodes pour étudier les systèmes biologiques de manière reproductible.

Pour saisir la dynamique complexe des processus biophysiques qui ont lieu dans les cellules, il est essentiel de disposer de méthodes permettant de mesurer avec précision et de manière quantitative les flux de membrane. Nous introduisons une nouvelle méthode pour l'inférence des flux à partir de kymographies, qui passe outre certaines limitations rencontrées par les méthodes actuelles telles que la vélocimétrie par images de particules (PIV) et la vélocimétrie de suivi de particules (PTV). Notre méthode relie la distribution de particules et les champs de vitesse à travers un modèle de flux qui peut ensuite être inversé en utilisant des méthodes de transport optimal. Cette approche permet d'inférer des flux dans des scénarios où le traçage des trajectoires de particules individuelles est irréalisable ou inefficace, en utilisant des kymographies comme une représentation grossière de la distribution de particules à partir de laquelle les flux peuvent être inférés. Nous validons notre algorithme en utilisant des données générées artificiellement, et l'appliquons à de vraies données de microscopie de cellules activables optogénétiquement qui génèrent des protrusions et, par conséquent, des flux, de façon répétable. Les flux inférés correspondent à la fois qualitativement et quantitativement aux données obtenues avec des expériences FRAP, tout en fournissant une résolution spatiale plus fine.

6.1 Introduction

The advent of novel microscopy paradigms has revealed the intricate and dynamic nature of cell biology. These techniques shed light on the spatiotemporal organization within cells, revealing how various biological molecules interact, move, and distribute themselves in the dynamic environment. To fully understand these processes, flow measurement techniques plays a major role, enabling us to decode the inner workings of cells by tracking the paths of biomolecules, organelles, and other sub-cellular entities. In particular, the membrane is involved in various processes, from cell motility to polarization and mitosis [1–4]. Measuring flows on membrane is difficult, but is absolutely required to better understand these processes.

In order to measure these flows, several methods have been developped. Particle Image Velocimetry (PIV) (first developed in fluid mechanics [5–7] it has then been applied with considerable success in biology [8, 9]) and Particle Tracking methods [10–12], involves the use of fluorescent particles as tracers in a fluid medium. These particles are tracked across successive frames, enabling the estimation of velocity fields. However, the adaptation of such methods for tracking membrane flows in biological systems remains difficult. In this case, the method requires to embed fluorescent particles within the cell membrane. The insertion of such particles is difficult, as cell membranes constitutes a complex medium that is far from an homogeneous lipid bilayer. Moreover, the fluorescent particles would have to be small enough to avoid disrupting the membrane dynamics, yet large enough to be detectable and trackable under a microscope. Balancing these opposing requirements poses a significant challenge. Thus, while PIV can provide valuable insights into bulk flow dynamics, its adaptation for membrane flow tracking in biological systems is tedious.

Optical flow estimation [13, 14] emerges as another powerful technique for understanding biological movement. Unlike PIV or particle tracking, this technique analyzes the apparent motion of brightness patterns to extract quantitative estimations of the velocity and direction of movement across the field of view. This method is particularly valuable in situations where tracking individual particles is impossible or impractical, such as when estimating movements of deformable elements such as biological objects. Applications in cell biology have included tracking cell movements in embryos [15] and movements of extracellular matrix [16]. While offering interesting insights, it is not well adapted to study internal rearrangements happening on the surface of the membrane.

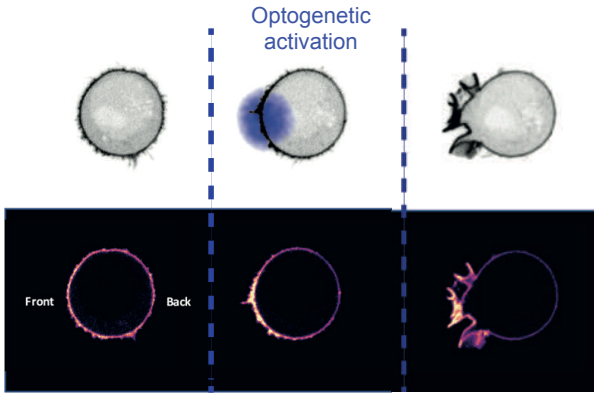


Figure 6.1: In our biological use-case, we observe flows when the cell protrudes (membrane become thinner at the back and thicker at the front). However, tracking the individual fluorophores is impossible.

Another widely used method to provide information about movement across membranes is Fluorescence Recovery After Photobleaching (FRAP) [17–21]. It involves a pre-bleaching phase where fluorescent structures are imaged, followed by a photobleaching phase where a targeted region within the cell is exposed to high-intensity light, resulting in irreversible fluorescence loss. Following photobleaching, the cell is imaged as fluorescence recovers due to the migration of unbleached, fluorescent molecules into the bleached area. Analysis of the fluorescence recovery curve yields information about diffusion coefficients, binding kinetics, and proportions of mobile and immobile molecules. FRAP has proven to be very effective for investigating molecular mobility and estimating diffusion rates [22, 23]. However, its low spatio-temporal resolution makes it less suited for studying active transport and flow dynamics of molecules over space and time.

In response to these challenges, we introduce a novel method to infer flow dynamics from kymographs. We postulate a flow model, that allow to formulate a master equation linking the particle distribution and the velocity field. This equation can be inverted using optimal transport, which provides a natural regularisation rooted in the physics of the system for this ill-posed inverse problem.. Our approach is designed for scenarios where tracking individual particle movement is impossible or impractical, instead focusing on kymographs, that constitutes a coarse-grained representation of particle distribution. We first validate our algorithm with artificially generated data. Then, we apply our method on optogenetically activable cells (Fig 6.1), that can generate protrusions at illumination, allowing to generate flows in a repeatable manner. By comparing the inferred flows with velocities deduced from FRAP experiments, we establish the biological relevance of the measured flows in both a qualitative and quantitative manner. Our code is available at *https://github.com/sacha-ichbiah/optiflow*.

6.2 Flow model

6.2.1 Biological context, presentation of the problem

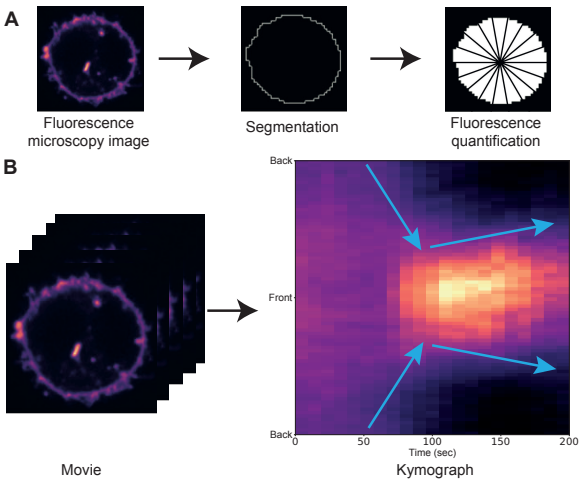


Figure 6.2: **Kymograph generation pipeline** (A) Cells are segmented, and the fluorescence intensity can be measured along the membrane contour across given angular sectors. (B) The quantification on every frame of a movie allow to generate a kymograph. Visually, it is possible to visualize two flows from the kymograph: The forward flow, from them back to the front, followed by a backward flow, from the front to the back.

Kymographs will be used for tracking dynamic membrane movements. They are created by

segmenting and quantifying the cellular structure's fluorescence distribution in each frame. The cell membrane is divided into uniformly spaced angular sectors, from which one can integrate the flurorophore distribution across space (Fig 6.2A). This process is done across all frames, generating a kymograph—a chronological display of membrane distribution across time (Fig 6.2A).

On the kymograph previously established, one may observe that at first, the distribution of membrane is uniform. After the illumination, the distribution becomes peaked at the front of the cell, becoming weaker at the back of the cell, before recovering to a uniform distribution after the illumination is stopped. From this visual analysis, several questions arise:

- What kind of information can we extract from these observations?
- What kind of hypotheses are needed to infer a flow field?

When looking at the kymograph, one can deduce visually the occurence of the flow, represented in Fig 6.2B as directional arrows representing movement. The slope of these arrows indicates the flow speed, providing a quantitative understanding of membrane dynamics. The observation that we can deduce flows visually from the kymograph is at the heart of our method. We will formalize these insights, and will study the collective displacement of the membranes with a simple microscopic model.

6.2.2 Microscopic model

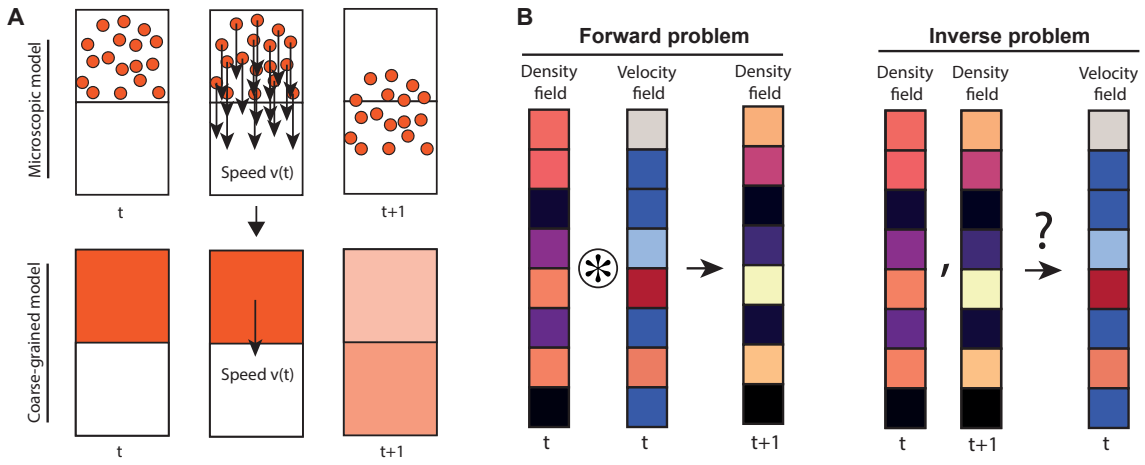


Figure 6.3: **Mathematical model used to infer flows from kymographs.** (A) Spatial averaging of independent fluorophore intensities leads to a flow model where the concentration at each discrete spatial position evolves depending on the velocity attached to it (B) Forward and inverse problems.

We will introduce a simple model to describe flow. Given an initial distribution and a time and space-varying velocity field, we can generate kymographs, i.e a table describing the history of the evolution of the flow. Inverting this model will allow us to infer the flows.

A discrete kymograph is defined by dividing the space into a finite number n of discrete boxes of spatial coordinates $x_k = k\Delta x$ ($k \in \mathbb{Z}/n\mathbb{Z}$) and the time in discrete time steps separated by a sampling time Δt. Inside each of these boxes, we hypothesize that a large number of fluorescent particles are present at random positions x. Biologically speaking, the particles

are proteins tagged with a fluorophore such as GFP, and they are transported with a time-variable velocity $v(x,t)$. Depending on their initial position, they can either remain into their initial box, or land into a neighboring one between a time t and a time $t+\Delta t$. Here, we assume the temporal sampling frequency high enough (i.e. the sampling time Δt small enough) such that, during a time Δt, a fluorophore will not travel a distance larger than the spatial sampling size Δx. For an ensemble of fluorescent particles, we can define the normalized velocity $\bar{v}(x,t) = |v(x,t)|\frac{\Delta t}{\Delta x}$. If $\Delta t \lesssim \Delta x / v_{\max}$, where $v_{\max}$ is the maximum velocity of a particle in the box, the normalized velocity measures a local probability for a particle to leave the box. This condition implies that our sampling time must be adapted to the velocity scale and small enough with respect to the sampling size.

If one can observe each particle and assign them a discrete position given the box in which they fall into, he will have:

- If $v(x,t) > 0$: $x(t+\Delta t) = x(t) + \Delta x$ with a probability $\bar{v}(x,t)$ and $x(t+\delta t) = x(t)$ with a probability $1 - \bar{v}(x,t)$
- If $v(x,t) < 0$: $x(t+\Delta t) = x(t) - \Delta x$ with a probability $\bar{v}(x,t)$ and $x(t+\delta t) = x(t)$ with a probability $1 - \bar{v}(x,t)$

6.2.3 Coarse-grained model

With classical fluorescence microscopy, following the displacement of each individual fluorophore is usually out of reach. We observe instead an average intensity, which represents the sum of individual fluorophores. Assuming a relatively uniform distribution of particles inside each box at each time step, the previous microscopic model allows us to describe in a coarse-grained manner the effect of a flow velocity $v(x_k,t)$ on an initial concentration $\mu^t(x_k)$, where the space is discretized as $x_k = k\Delta x$ ($k \in \mathbb{Z}/n\mathbb{Z}$).

To do so, we define a normalized velocity field, average of the true velocity of all fluorophores in each box located at x_k as $V^t(x_k) = \int_{x_k-\Delta x/2}^{x_k+\Delta x/2} |v(x,t)|\frac{\Delta t}{\Delta x^2}\mathrm{d}x$. Following our reasoning above, this normalized velocity field also describes the fraction of particles that will jump to the next box x_{k+1} (or to the previous one if $v(x,t) < 0$) in a time Δt, while a fraction $1 - \bar{V}^t(x_k)$ will remain in its initial box centered at x_k.

If we consider the concentration field of particles $\mu^t(x_k)_{x_k \in \mathbb{Z}/n\mathbb{Z}}$, and a normalized velocity field $V^t(x)_{x\in\mathbb{Z}/n\mathbb{Z}}$, the evolution of the concentration may be cast into a master equation, for all k:

$$\begin{aligned}
\mu^{t+\Delta t}(x_k) &= \mathcal{F}[\mu^t, V^t](x_k) \\
&= \mu^t(x_{k-1})V^t(x_{k-1})\mathbb{1}_{V^t(x_{k-1})>0} \\
&\quad + \mu^t(x_{k+1})V^t(x_{k+1})\mathbb{1}_{V^t(x_{k+1})<0} \\
&\quad + \mu^t(x_k)(1 - |V^t(x_k)|)
\end{aligned} \tag{6.1}$$

where $\mathbb{1}_{V^t(x)>0}$ and $\mathbb{1}_{V^t(x)<0}$ are indicator functions that are 1 when $V^t(x) > 0$, respectively $V^t(x) < 0$ and 0 otherwise.

6.3 Inverse problem

6.3.1 Flow inference

When we observe the distributions of membrane intensity at two consecutive time steps μ^t and $\mu^{t+\Delta t}$, we observe that there has been a collective displacement of the particles between the two positions. To determine which particle goes where on average, we want to invert the master equation (6.1): given the fields μ^t and $\mu^{t+\Delta t}$, find V^t such that $\forall k \in \mathbb{Z}/n\mathbb{Z}, \mu^{t+\Delta t}(x_k) = \mathcal{F}[\mu^t, V^t](x_k)$.

This is a complicated inverse problem, due to the fact that the above expression changes with the sign of $V^t(x)$. Moreover, it is ill-posed: the solution is not unique and there is generally an infinite number of velocity fields that can explain the temporal evolution of the intensity distributions. To regularise this inverse problem, we implement prior knowledge based on physical principles. A first important physical hypothesis is that there is a notion of dissipation associated here to the flow. This dissipation is modeled by a cost function that we aim at minimizing. This is physically a fair hypothesis for inferring the flows of the membrane, that is limited by the relative friction with the cortex.

We therefore choose a cost function that depends on local the velocity v_P of the path P as $\mathcal{C} \propto v_P^2$, which is consistent with a classical frictional dissipation. The local concentration of fluorophores being good indicator of the membrane density and, as a consequence, of the number of friction points, the local dissipation cost is furthermore expected to be proportional to the local concentration and we define $\mathcal{C}^t(x) = \mu^t(x)|V^t(x)|^2$.

We now reformulate the flow inference problem between two time steps t and $t + \Delta t$ as a minimisation of a cost function under the constraint of the master equation (6.1):

$$\min_{V^t \text{ s.t } \mathcal{F}(\mu^t, V^t) = \mu^{t+\Delta t}} \sum_k \mu^t(x_k)|V^t(x_k)|^q \tag{6.2}$$

where $q = 2$ corresponds to the physical cost function $\mathcal{C}^t(x) = \mu^t(x)|V^t(x)|^2$.
Remarking that the time Δt between two sampling points remains the same, we can alternatively express the dissipation cost as function of the length l_P of the path P, as far as it remains small: $\mathcal{C}^t(x) = \mu_t(t)\frac{l_P(x)^2}{\Delta t^2}$.

We propose below a way to solve this constrained optimisation problem using optimal transport. In fact, this method allows us to solve this problem for any $q \in \mathbb{R}$ in equation (6.2).

6.3.2 Optimal transport

Continuous transportation theory

Constrained optimisation problems as formulated in equation (6.2) can be cast into the generic class of optimal transport problems, which is a mathematical field formalized by Gaspard Monge [24], and revisited by Leonid Kantorovich who gave it its contemporary formulation [25].

To introduce shortly the optimal transport formulation, we first define a distance function $d(\cdot, \cdot)$, that takes as arguments two spatial positions along the membrane and returns a scalar value that represents the distance between two points. We write $c(\cdot, \cdot)$ the cost function that

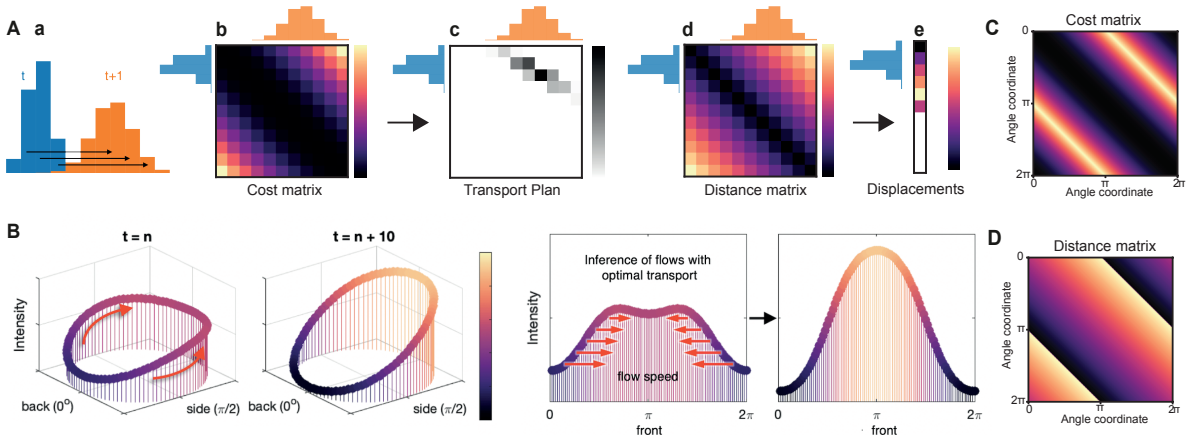


Figure 6.4: **Optimal transport flow inference procedure** (A) (a) An illustrative example of optimal transport between two discrete 1-dimensional distributions, at time t (blue) and time t + 1 (orange), which represent the amounts of membrane (or actin) along the membrane contour at two different time points. (b) Cost matrix C, in which C[i,j] indicates the value of the cost to displace an element from position i to the position j. Here, the cost function shown is the square of the curvilinear distance. (c) Transport Plan to go from the distribution at time t to the distribution at time t + 1, minimizing the total cost of the displacement, computed from the cost matrix in (b). (d) Distance matrix D, in which D[i,j] indicates the value of the distance between an element at the position i and an element at the position j. The distance chosen is the curvilinear distance. (e) The transport plan and the distance matrix allow to compute the mean displacement for every position between times t and t + 1. (B) When inferring flow on cell membranes, we need to take into account the periodicity of the contour, and modify cost and distance matrices accordingly. (C) Modified cost matrix (D) Modified distance matrix.

takes as arguments spatial positions along the membrane and returns a scalar value that represents the cost to go from one place to another. In the following, we will use $c(\cdot,\cdot) = d^q(\cdot,\cdot)$. We call the transport m a map such that $m(\mu^t) = \mu^{t+1}$.

In the continuous setting, we want to find the optimal transport m^* that minimizes:

$$\min_{m, m(\mu^t)=\mu^{t+1}} \int c(x, m(x)) d\mu^t(x), \tag{6.3}$$

where the integration is done on the probability measure $\mu^t(x)$.
The optimal cost $W_c(\mu^t, \mu^{t+1}) = \int c(x, m^*(x)) d\mu^t(x)$ is called the Wasserstein distance between two measures. Here we are more interested on the mapping on itself, that is called the optimal transport matrix. Indeed, as this matrix gives a notion of distance traveled at each point on the membrane, we will determine the speed of the membrane flows using these discrete displacements.

Discrete optimal transport

Here we study discrete distributions. Following the notations of [26], we define a cost matrix C, a transport matrix M and a distance matrix D (Fig 6.4A.a-d), discretized versions of their continuous surrogates $c(\cdot,\cdot)$, $m(\cdot)$ and $d(\cdot,\cdot)$. For each discrete boxes i and j, of positions x_i and x_j, we have : $D_{i,j} = d(x_i, x_j)$. The cost matrix is defined generally as $C_{i,j} = |D_{i,j}|^q = |d^q(x_i, x_j)|$, following the continuous case.

We write $\mathbb{1}$ the column vector where each value is one. For two simple distributions a and b, the optimal transport M^* is defined by:

$$M^* = \underset{M \in \mathbb{R}^{m \times n}}{\operatorname{argmin}} \sum_{i,j} M_{i,j} C_{i,j} \tag{6.4}$$

Where $M\mathbb{1} = a; M^T\mathbb{1} = b; \forall (i,j), M_{i,j} > 0$. We compute the optimal transport M^* using the standard Earth Mover's Distance (EMD) with [26]. This exact computation does not scales well with the number of boxes n (It has a $O(n^3)$ complexity [27]) but the 1D case has a low enough number of points to obtain results in a reasonable amount of time.

The velocity field V is obtained by averaging the distance traveled for every element (Fig 6.4A.e), and dividing by the timestep Δt:

$$\forall i, V_i = \frac{1}{\Delta t}(D_i \star M_i / a)^T \times \mathbb{1} \tag{6.5}$$

Where $\star$ is an element-wise multiplication, / the element wise division and $\times$ is the standard matrix product. X_i denotes the ith column of the matrix $X_{i,j}$.

Optimal transport with periodic boundary conditions

In our use-case, the cells are circular, and thus our distributions have periodic boundary conditions. One can travel along the membrane contour using a clockwise and an anticlockwise path. To account for this periodicity, we adapted the cost and distances matrices (Fig 6.4B-D). Moreover, we decided to compute algebraic distances (i.e positive or negative values), to determine the direction of the flow from its sign (Fig 6.4D).

6.4 Results

6.4.1 Generation of velocity fields

We generate flow fields V from random Gaussian distributions of mean 0 and variance 1, with values clipped between -1 and 1, that are smoothed twice using a sliding window of varying length between 1 and 20, to average the sample values, both in space and time. Raising the length of the convolution window raises the smoothness of the generated velocity field, allowing to study the efficiency of inference with different regularities (Fig 6.5A).
The master equation 6.1 allows us to simulate flows when given an initial uniform concentration field μ^0 and a flow field V (i.e we solve the forward problem of Fig 6.3B). We generate data with n = 200 spatial coordinates and 50 timesteps, that is similar to the biological data we will use later.

6.4.2 In-silico validation

Then we use our inference method to reconstruct a velocity field from the kymograph artificially generated (Fig 6.5B). With n = 200 spatial coordinates and 50 timesteps, the inference procedure with [26] takes on average 0.612s[1]. The reconstruction error depends weakly on the time convolution window, and mostly on the spatial convolution window (Fig 6.5C), which is relevant with the fact that we infer the flows timestep by timestep. The flow reconstruction is thus easier when inferring flows with a lower spatial heterogeneity (Fig 6.5C,D). As explained before, we solved the problem for every distance $q \in \mathbb{N}^*$. We evaluated the average error taking $q \in [1, 10]$ (Fig 6.5E), and observe that the error is low with $q \in [2, 6]$, and raises a

[1] average over 100 iterations on a 8-core intel i9 CPU

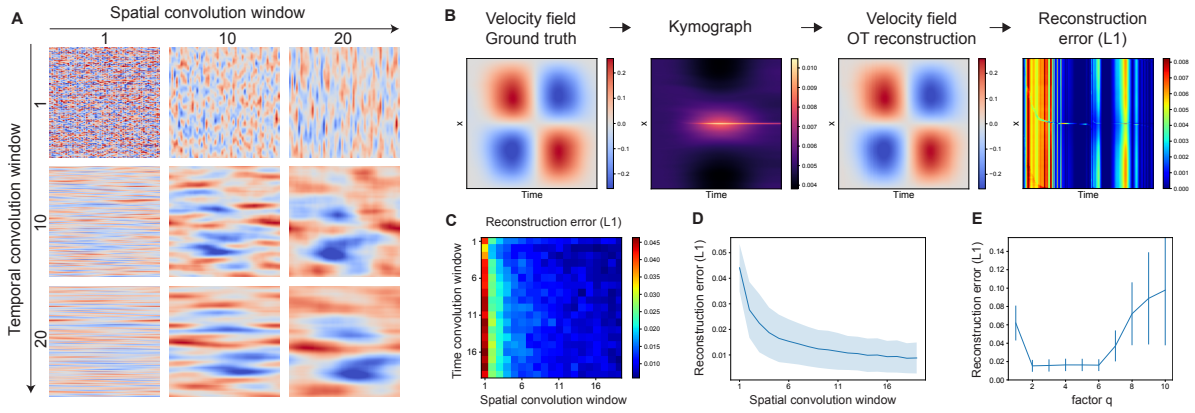


Figure 6.5: **Numerical validation of the flow inference method on various simulated velocity fields.** (A) We use various spatio-temporal convolution windows to create smooth velocity fields from random gaussian values. (B) We use these velocity fields to generate kymographs from an initial uniform distribution. We evaluate the error between the reconstruction and the ground truth using an L1 error. (C) Heatmap of the average (n_samples = 10) reconstruction error as a function of the time and spatial convolution window. (D) Plot of the average reconstruction error as a function of the spatial convolution window (n_samples = 200, 10 samples for each temporal convolution window ranging from 1 to 20) (E) Reconstruction error as a function of the factor q.

lot for the other powers, most probably because of numerical errors, confirming our choice of $q = 2$.

6.4.3 Application to neutrophil-like HL60 cells

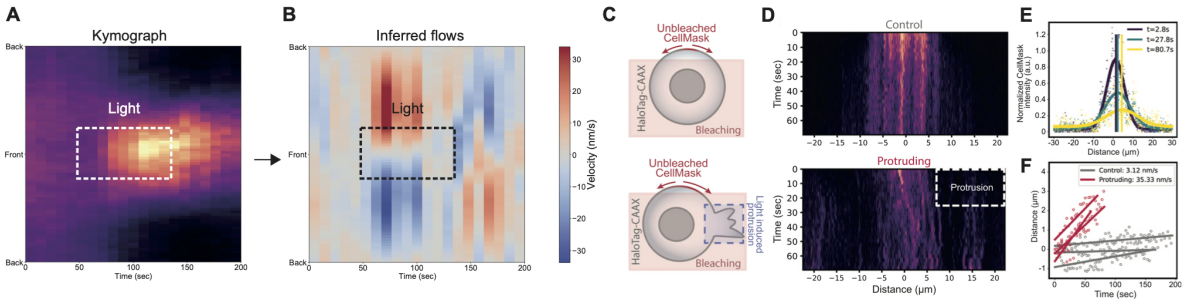


Figure 6.6: **Experimental validation of the flow inference method with FRAP experiments.** (A) Kymographs of membrane fluorescence along the normalized cell circumference (y axis) show that over time (x axis) membrane accumulates toward the protruding cell front and is depleted from the back (n > 50, N = 6) (B) Inferred flows with our procedure (B). (C) FRAP membrane diffusion assay (D) example kymograph of unbiased diffusion in a control cell and a protruding cell (E) sample fits of individual timepoints of kymograph data with a Gaussian equation (F) Quantification of mean shifts fit by linear regression to assay membrane flow rate in control cells and protruding cells. The quantifications exhibit no apparent flow in control cells (u = 3.34 nm/s) and a biased flow toward the side of protrusion in protruding cells (u = 35.51 nm/s) (N = 3, n = 3).

Then, we apply our method to neutrophil-like HL60 cells, using kymographs from our optogenetics experiments to infer flows, with n =80 and $\delta t = 10s$ (Fig 6.6A,B). We recover two flows, consistently with what we observed visually in Fig 6.2B (blue arrows). The forward flow, from the back to the front, occur briefly after illumination (starting at t=70s, until 110s). It is followed by the backward flow, starting at t=140s until t=180s.

We validate these measurements with FRAP experiments (Fig. 6.6C). The experiments show a time-dependent diffusion of fluorescence intensity consistent with a Gaussian fit of the FRAP data (Fig 6.6D-E), allowing to measure the velociy of the membrane flow during protrusion . The maximum velocity obtained from our method during the cell's protrusion phase is 33.87 nm/s (Fig 6.6B), aligning closely with the 35.33 nm/s velocity from the FRAP experiments during similar circumstances (Fig 6.6F). This close agreement between our method's results and FRAP data reinforces the validity of our inferences.

6.5 Conclusion

In this study, we introduced and evaluated a novel method for the inference of membrane flows from kymographs, providing an alternative to traditional techniques such as PIV and FRAP. Our method leverages a flow model, linking particle distribution and velocity fields that can be inverted using optimal transport theory. The results validated on both artificially generated data and real microscopy data indicate that our method can successfully infer flow dynamics in situations where tracing individual particle trajectories is unfeasible.

An application to 3D data would hold many promises, but is much more complicated, requiring a much more complex parametrization [28, 29] and registration [30–32] based on differential geometry. However, our approach based on the inversion of a forward problem is very general, and can be applied to many problems to make physical measurements from microscopy data. The implementation of our method, available at *https://github.com/sacha-ichbiah/optiflow*, provides a new and accessible tool for biologists, enrishing the toolbox to study and understand the dynamic nature of cellular processes. As the need for more accurate, reliable, and high-resolution flow measurement techniques continues to grow, our method offers a promising solution for the challenges encountered in measuring membrane flows in biological systems.

Bibliography

[1] Hervé Turlier, Basile Audoly, Jacques Prost, and Jean-François Joanny. Furrow constriction in animal cell cytokinesis. *Biophysical journal*, 106(1):114–123, 2014.

[2] Peter Gross, K Vijay Kumar, Nathan W Goehring, Justin S Bois, Carsten Hoege, Frank Jülicher, and Stephan W Grill. Guiding self-organized pattern formation in cell polarity establishment. *Nature Physics*, 15(3):293–300, 2019.

[3] Veneta Gerganova, Iker Lamas, David M Rutkowski, Aleksandar Vještica, Daniela Gallo Castro, Vincent Vincenzetti, Dimitrios Vavylonis, and Sophie G Martin. Cell patterning by secretion-induced plasma membrane flows. *Science Advances*, 7(38):eabg6718, 2021.

[4] Henry De Belly, Shannon Yan, Hudson Borja da Rocha, Sacha Ichbiah, Jason P Town, Patrick J Zager, Dorothy C Estrada, Kirstin Meyer, Hervé Turlier, Carlos Bustamante, et al. Cell protrusions and contractions generate long-range membrane tension propagation. *Cell*, 2023.

[5] Ronald J Adrian. Particle-imaging techniques for experimental fluid mechanics. *Annual review of fluid mechanics*, 23(1):261–304, 1991.

[6] Adrian Melling. Tracer particles and seeding for particle image velocimetry. *Measurement science and technology*, 8(12):1406, 1997.

[7] Carl D Meinhart, Steve T Wereley, and Juan G Santiago. Piv measurements of a microchannel flow. *Experiments in fluids*, 27(5):414–419, 1999.

[8] Dhruv K Vig, Alex E Hamby, and Charles W Wolgemuth. On the quantification of cellular velocity fields. *Biophysical journal*, 110(7):1469–1475, 2016.

[9] Yasuhiko Sugii, Shigeru Nishio, and Koji Okamoto. In vivo piv measurement of red blood cell velocity field in microvessels considering mesentery motion. *Physiological measurement*, 23(2):403, 2002.

[10] Jean-Yves Tinevez, Nick Perry, Johannes Schindelin, Genevieve M Hoopes, Gregory D Reynolds, Emmanuel Laplantine, Sebastian Y Bednarek, Spencer L Shorte, and Kevin W Eliceiri. Trackmate: An open and extensible platform for single-particle tracking. *Methods*, 115:80–90, 2017.

[11] Dmitry Ershov, Minh-Son Phan, Joanna W Pylvänäinen, Stéphane U Rigaud, Laure Le Blanc, Arthur Charles-Orszag, James RW Conway, Romain F Laine, Nathan H Roy, Daria Bonazzi, et al. Trackmate 7: integrating state-of-the-art segmentation algorithms into tracking pipelines. *Nature Methods*, 19(7):829–832, 2022.

[12] Nicolas Chenouard, Ihor Smal, Fabrice De Chaumont, Martin Maška, Ivo F Sbalzarini, Yuanhao Gong, Janick Cardinale, Craig Carthel, Stefano Coraluppi, Mark Winter, et al. Objective comparison of particle tracking methods. *Nature methods*, 11(3):281–289, 2014.

[13] Berthold KP Horn and Brian G Schunck. Determining optical flow. *Artificial intelligence*, 17(1-3):185–203, 1981.

[14] David Fleet and Yair Weiss. Optical flow estimation. In *Handbook of mathematical models in computer vision*, pages 237–257. Springer, 2006.

[15] Fernando Amat, Eugene W Myers, and Philipp J Keller. Fast and robust optical flow for time-lapse microscopy using super-voxels. *Bioinformatics*, 29(3):373–380, 2013.

[16] Sandeep Manandhar, Patrick Bouthemy, Eric Welf, Philippe Roudot, and Charles Kervrann. 3d optical flow estimation combining 3d census signature and total variation regularization. In *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*, pages 965–968. IEEE, 2020.

[17] Hellen C Ishikawa-Ankerhold, Richard Ankerhold, and Gregor PC Drummen. Advanced fluorescence microscopy techniques—frap, flip, flap, fret and flim. *Molecules*, 17(4):4047–4132, 2012.

[18] Gustavo Carrero, Darin McDonald, Ellen Crawford, Gerda de Vries, and Michael J Hendzel. Using frap and mathematical modeling to determine the in vivo kinetics of nuclear proteins. *Methods*, 29(1):14–28, 2003.

[19] James G McNally. Quantitative frap in analysis of molecular binding dynamics in vivo. *Methods in cell biology*, 85:329–351, 2008.

[20] Brian L Sprague and James G McNally. Frap analysis of binding: proper and fitting. *Trends in cell biology*, 15(2):84–91, 2005.

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

[22] Hendrik Deschout, Joel Hagman, Sophia Fransson, Jenny Jonasson, Mats Rudemo, Niklas Lorén, and Kevin Braeckmans. Straightforward frap for quantitative diffusion measurements with a laser scanning microscope. *Optics express*, 18(22):22886–22905, 2010.

[23] Minchul Kang, Charles A Day, Anne K Kenworthy, and Emmanuele DiBenedetto. Simplified equation to extract diffusion coefficients from confocal frap data. *Traffic*, 13(12):1589–1600, 2012.

[24] Monge. G. Mémoire sur la théorie des déblais et des remblais. *Histoire de l'Académie Royale des Sciences de Paris*, pages 666–704, 1781.

[25] L.V. Kantorovich. On the translocation of masses. *Acad. Sci. URSS*, 37:199–201, 1942.

[26] Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Z Alaya, Aurélie Boisbunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras, Nemo Fournier, et al. Pot: Python optimal transport. *The Journal of Machine Learning Research*, 22(1):3571–3578, 2021.

[27] Ravindra K Ahuja, Thomas L Magnanti, and James B Orlin. *Network flows: theory, algorithms and applications*. Prentice Hall, 1995.

[28] Nicolas Romeo, Alasdair Hastewell, Alexander Mietke, and Jörn Dunkel. Learning developmental mode dynamics from single-cell trajectories. *Elife*, 10:e68679, 2021.

[29] Nina Miolane, Nicolas Guigui, Alice Le Brigant, Johan Mathe, Benjamin Hou, Yann Thanwerdas, Stefan Heyder, Olivier Peltre, Niklas Koep, Hadi Zaatiti, et al. Geomstats: a python package for riemannian geometry in machine learning. *The Journal of Machine Learning Research*, 21(1):9203–9211, 2020.

[30] Mark Gillespie, Boris Springborn, and Keenan Crane. Discrete conformal equivalence of polyhedral surfaces. *ACM Transactions on Graphics (TOG)*, 40(4):1–20, 2021.

[31] James Klatzow, Giovanni Dalmasso, Neus Martínez-Abadías, James Sharpe, and Virginie Uhlmann. μmatch: 3d shape correspondence for biological image data. *Frontiers in Computer Science*, 4:777615, 2022.

[32] Alexandre Bône, Maxime Louis, Benoît Martin, and Stanley Durrleman. Deformetrica 4: an open-source software for statistical shape analysis. In *ShapeMI @ MICCAI 2018*, Granada, Spain, September 2018.

General conclusion

Understanding the world, always and forever, consists of creating bridges between bits and atoms. Models represent processed information (bits), and their value lies in how they correspond to the information present in physical phenomena (including biological systems), that is, atoms. The field of study known as the "Inverse Problem" is focused on making models match data. It has two aspects: first, to determine a model's parameters from the data; and second, to determine the model itself, given certain constraints. There are two complementary approaches to solve an inverse problem: Going from the physics to the model, or the other way, from the model to the physics. Throughout my three years of PhD research, I have worked to develop tools that facilitate both approaches, aiming to better understand the mechanical aspects of embryonic development.

The most conventional approach is to go from the physics to the model. The idea is to process as much as possible the physical phenomena, to simplify its representation in a physical model. Our process began with observing the biological phenomena under study through a microscope, capturing photons emitted by the biological samples with a CMOS (complementary metal-oxide-semiconductor) image sensor to produce a digital image. The features of this image can then be extracted, analyzed, and compared with models. In Chapters 3 and 4, we segmented images of early embryos and extracted their geometry to fit the parameters of an active foam model, using mean-square error minimization, or a more sophisticated optimization procedure where gradients are computed with the adjoint-state method. In Chapter 6, we segmented films of cells and generated kymographs from the fluorescence quantifications of the membrane contour. Then we used these kymographs to compute membrane flows by inverting a physical model.

With our vast computational resources, we can explore new methodologies that proceed in the opposite direction, from the model to the physics. Deep learning enables us to work with highly abstract data, reducing the need to preprocess input images, and allows to directly fit models from raw data. In Chapter 5, we developed an artificial microscope to compare the model and biology in the space of images. The differentiability of this pipeline allow to fit the parameters of any differentiable physical model. In particular, we retrieve the cell geometry and a model of the point-spread function of the microscope.

I'd like to end with a quote from Nobel prize laureate James Watson from his famous book *The Double Helix: A Personal Account of the Discovery of the Structure of DNA*: "I have never seen Francis Crick in a modest mood. Perhaps in other company he is that way, but I have never had reason so to judge him".

When pursuing innovative ideas, challenging established theories, and persevering through setbacks, just forget the modest mood. Be bold. God leads the way.

Do not forget, we are not trying to live forever. Biology is not this journey and will never be. Keep it real. With certainty, one day you will die, just like every human being before you, most probably in pain. Make peace with that.

We are not here to make ourselves God. We are just here to make human existence more meaningful. We are here to invent new cures, to prevent illness, and to engineer the living to solve the most pressing problems. We are here to free ourselves from the curse of hereditary diseases. We are here to contemplate the wonders of nature, in its infinite complexity. We are here to resurrect ancient species so the world can witness their beauty again. We are here to do great things.

And great things we will do.

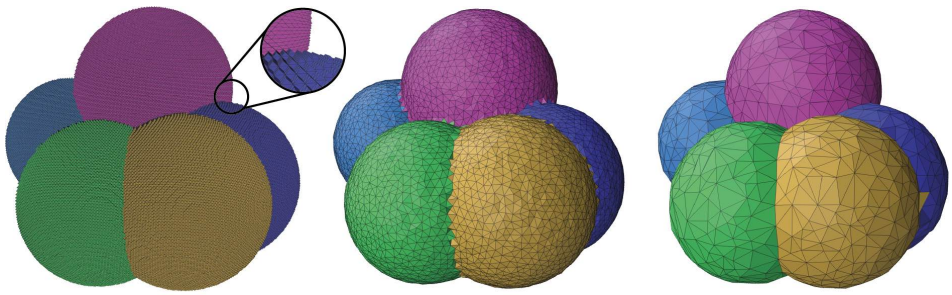
Appendix A

Supplementary note of chapter 3

In this Supplementary Note, we provide additional theoretical and implementation details on the Delaunay-watershed mesh generation algorithm, the 3D active foam simulations, and the force inference pipeline.

A.1 Multimaterial mesh generation

Our multimaterial nonmanifold triangle mesh data structure for surfaces is quite specific, and there is currently no library available to generate them directly. To overcome this difficulty, we used procedures to generate multimaterial tetrahedral volume meshes and developed methods to extract their boundaries as multimaterial triangle surface meshes.



A.1.1 Instance segmentation

If we decompose each voxel into four tetrahedra, an instance segmentation can be viewed as a tetrahedral mesh. The task of generating a mesh from a voxel segmentation can thus be viewed as a remeshing procedure, where we convert a very fine mesh into a coarser one, which is better suited to our analysis.

A.1.2 CGAL-based 3D surface mesh generation

CGAL [1] provides a popular tetrahedral mesh generation algorithm, which is widely used in biomechanical modeling, to extract volumetric meshes from 3D scans from different imaging modalities. It constitutes our baseline, from which we compared the results obtained with our proposed remeshing algorithm, Delaunay-Watershed.

A.1.3 Delaunay-watershed mesh generation algorithm (2D and 3D)

Distance transform map

Given an instance segmentation, we find the boundaries of the different instances with the function *find_boundaries* of the Python package *scikit-image* ([2]) These borders can be converted into a distance map using a Euclidean distance transform with the function *distance_transform_edt* of the Python package *SciPy* ([3]).

Sampling of control points

To compute a tessellation of the space, we need to choose control points that will form the vertices of the tessellation. The points of interest lie into the borders of the objects, i.e. where the EDT is minimal. To regularize the tessellation, one can also add the local maxima. These extrema are efficiently computed using the *maxpool3d* function of *PyTorch* [4], with a kernel size of (d,d,d), $d \in \mathbb{Z}^+$, that will define the refinement of our mesh, and a stride of (1,1,1).

Delaunay tesselation

From these points, n-dimensional Delaunay tessellations are generated using the Delaunay function from the Python library scipy.spatial.

Dual Voronoi graph

The dual Voronoi graph $\mathcal{G} = (\mathcal{N}, \mathcal{L}, \mathcal{W})$ is then constructed from the tessellation, by looking at the adjacency of the simplices obtained. In this dual representation:
• In 2D, $\mathcal{N}$ contains triangles, $\mathcal{L}$ edges, and weights are calculated by averaging the value of the Euclidean distance transform of $n_s = 10$ points distributed regularly along the edge.
• In 3D, $\mathcal{N}$ contains the tetrahedra, $\mathcal{L}$ the triangles, and the weights are calculated by computed the average value of the euclidean distance transform of $n_s = 9$ regularly sampled points distributed within the triangle. This graph is converted into a *Networkx* ([5]) graph, better suited to do operations such as node clustering.

Watershed algorithm

We use the algorithm 2 from [6] to implement the seeded watershed. The seeds are computed using the EDT: for each label l, we take the pixel px_l^* corresponding to the global minimum of the EDT of the pixels having this label. However, to apply our watershed algorithm, the seed needs to be a node of our graph, i.e., a simplex of the tessellation: a tetrahedra in 3D or a triangle in 2D. For each label, we take the simplex of centroid c that minimizes the distance to the pixel px_l^* previously found as a seed.

A.1.4 Discrete expressions of geometrical quantities on triangular surface meshes

Below, we provide analytical formulas for calculating several discrete geometrical quantities and their derivatives on triangular surface meshes. In the following, we will use Roman indices (i, j, k) for quantities associated with cells (or regions), such as a tension γ_{ij} between cells of indices i and j, and Greek indices α for quantities associated with the mesh, such as the vertices denoted $\{\vec{x}_\alpha\}_{\alpha=1}^{n_v}$.

Interface area and derivatives

Each interface area m and its gradient may be decomposed into a sum on its triangles

$$\mathcal{A}_m = \sum_{t\in\mathcal{A}_m} A_t, \qquad \frac{\partial \mathcal{A}_m}{\partial \vec{x}_\alpha} = \sum_{t\in\mathcal{A}_m} \frac{\partial A_t}{\partial \vec{x}_\alpha}. \tag{A.1}$$

Therefore, we provide the formula for only a single triangle, defined by its three vertices: $t = (\vec{x}_\alpha, \vec{x}_\beta, \vec{x}_\gamma)$. The non-unit and unit normals to the triangle t are defined as

$$\vec{N}_t = (\vec{x}_\beta - \vec{x}_\alpha) \times (\vec{x}_\gamma - \vec{x}_\alpha), \qquad \vec{n}_t = \frac{\vec{N}_t}{\|\vec{N}_t\|}. \tag{A.2}$$

The area of the triangle t follows as:

$$A_t = \frac{1}{2}\|\vec{N}_t\|. \tag{A.3}$$

To obtain the area gradient with respect to $\vec{x}_\alpha$, we calculate the variation of $A_t^2 = \frac{1}{4}\vec{N}_t \cdot \vec{N}_t$:

$$\delta(S_t^2) = \frac{1}{2}\vec{N}_t \cdot \delta\vec{N}_t = \frac{1}{2}\vec{N}_t \cdot \Big[(\vec{x}_\gamma - \vec{x}_\beta) \times \delta\vec{x}_\alpha\Big] = \frac{1}{2}\vec{N}_t \times \Big[(\vec{x}_\gamma - \vec{x}_\beta)\Big] \cdot \delta\vec{x}_\alpha.$$

From $\delta(A_t^2) = 2A_t\,\delta(A_t)$, we deduce

$$\frac{\partial A_t}{\partial \vec{x}_\alpha} = \frac{1}{2}\vec{n}_t \times (\vec{x}_\gamma - \vec{x}_\beta). \tag{A.4}$$

Cell volumes and derivatives

Stokes' theorem tells us that for a closed manifold, for any vectorial function f we have :

$$\iiint_{\mathcal{V}} \mathrm{div}(\vec{f})\mathrm{d}\tau = \oint_S \vec{f}\cdot\vec{n}\,\mathrm{d}s.$$

If we take $\vec{f}(\vec{x}) = \frac{1}{3}\vec{x}$, for which $div(\vec{f})(\vec{x}) = \frac{1}{3}(1+1+1)$, we have :

$$\mathcal{V} = \iiint_{\mathcal{V}} \mathrm{d}\tau = \frac{1}{3}\oint_S \vec{x}\cdot\vec{n}\,\mathrm{d}s = \frac{1}{3}\sum_{t=(\vec{x}_\alpha,\vec{x}_\beta,\vec{x}_\gamma)\in S} \iint_t \vec{x}\cdot\vec{n}\,\mathrm{d}s.$$

We will evaluate this integral on every triangle $t = (\vec{x}_\alpha, \vec{x}_\beta, \vec{x}_\gamma) \in S$.
The center of gravity of the triangle, defined as $\iint_{\vec{x}\in t}(\vec{x} - \vec{x}_G)\cdot\vec{n}\,\mathrm{d}s = 0$ reads

$$\vec{x}_G = \frac{\vec{x}_\alpha + \vec{x}_\beta + \vec{x}_\gamma}{3}. \tag{A.5}$$

We have

$$\iint_{\vec{x}\in t} \vec{x}\cdot\vec{n}\,\mathrm{d}s = \iint_{\vec{x}\in t} \vec{x}_G\cdot\vec{n}\,\mathrm{d}s = (\vec{n}\cdot\vec{x}_G)S_t = (\vec{n}\cdot\vec{x}_G)\frac{\|(\vec{x}_\gamma - \vec{x}_\alpha)\times(\vec{x}_\beta - \vec{x}_\alpha)\|}{2}.$$

By developing the expressions of $\vec{x}_G$ and S_t, we can write:

$$\begin{aligned}
\iint_{\vec{x}\in t} \vec{x}\cdot\vec{n}\,\mathrm{d}s &= \frac{1}{6}(\vec{x}_\alpha + \vec{x}_\beta + \vec{x}_\gamma)\cdot[(\vec{x}_\gamma - \vec{x}_\alpha)\times(\vec{x}_\beta - \vec{x}_\alpha)] \\
&= \frac{1}{6}(\vec{x}_\gamma - \vec{x}_\alpha + \vec{x}_\beta - \vec{x}_\alpha + 3\vec{x}_\alpha + \vec{x}_\beta + \vec{x}_\gamma)\cdot[(\vec{x}_\gamma - \vec{x}_\alpha)\times(\vec{x}_\beta - \vec{x}_\alpha)] \\
&= \frac{1}{2}\vec{x}_\alpha\cdot(\vec{x}_\gamma\times\vec{x}_\beta) \\
&= \frac{1}{2}\det(\vec{x}_\alpha, \vec{x}_\beta, \vec{x}_\gamma).
\end{aligned}$$

Finally,

$$\mathcal{V} = \frac{1}{3}\oint_S \vec{x}\cdot\vec{n}\,\mathrm{d}s = \frac{1}{6}\sum_{(\vec{x}_\alpha,\vec{x}_\beta,\vec{x}_\gamma)\in S} \det(\vec{x}_\alpha,\vec{x}_\beta,\vec{x}_\gamma). \tag{A.6}$$

Thus, the derivative of the volume with respect to the vertex writes

$$\frac{\partial\mathcal{V}}{\partial\vec{x}_\alpha} = \frac{1}{6}\sum_{(\vec{x}_\alpha,\vec{x}_\beta,\vec{x}_\gamma)\in triangles} \vec{x}_\beta\times\vec{x}_\gamma. \tag{A.7}$$

Junction length and derivatives

The length of a discrete curve can be written :

$$\mathcal{L} = \sum_{(\vec{x_\alpha},\vec{x_\beta})\in edges} \|\vec{x_\alpha}-\vec{x_\beta}\|. \tag{A.8}$$

Consider an edge $(\vec{x_\alpha},\vec{x_\beta})$ of length $\mathrm{L}(\vec{x_\alpha},\vec{x_\beta}) = \|\vec{x_\alpha}-\vec{x_\beta}\|$. To compute its derivative, we place ourselves on a basis where $\vec{e}_1 = \frac{\vec{x_\alpha}-\vec{x_\beta}}{\|\vec{x_\alpha}-\vec{x_\beta}\|}$ and

$$\mathrm{L}(\vec{x_\alpha}+\vec{\delta},\vec{x_\beta}) = \|\vec{x_\alpha}-\vec{x_\beta}\| + \vec{\delta}\cdot\vec{e}_1.$$

The derivative with respect to each vertex is deduced as follows

$$\frac{\partial \mathrm{L}}{\partial\vec{x_\alpha}} = \frac{\vec{x_\alpha}-\vec{x_\beta}}{\|\vec{x_\alpha}-\vec{x_\beta}\|}. \tag{A.9}$$

Contact angles

For each junction (i,j,k) between three cells (or the exterior), each edge is connected to three triangles, making the interface between two materials: t_{ij}, t_{ik}, t_{jk} of normals $\vec{n}_{ij}, \vec{n}_{ik}, \vec{n}_{jk}$ (See Ext Fig. 1a)

$$\alpha_{ab} = \arccos(\vec{n}_{ac}\cdot\vec{n}_{bc}) \text{ for any (a,b,c) permutation of (i,j,k).} \tag{A.10}$$

For force inference the junctional angles that appears in Young-Dupré force balance variants are defined as the average of the dihedral angles computed on every edges belonging to the junction.

Mean curvature

For a smooth manifold, the mean curvature $\bar{\kappa}$ at a given point is, by definition, the divergence of the normal vector at this point. A possible differential geometry definition of the curvature normal $H\,\vec{n}$ is

$$H\vec{n} = \lim_{\mathrm{diam(A)}\to 0} \frac{\vec{\nabla}A}{AS},$$

where A is a small area around the point where the curvature is evaluated and diam(S) its diameter.
A discrete geometry equivalent to the previous formula was derived by [7]

$$-H_i\,\vec{n}_i = \frac{1}{4}\sum_{j\in N_1(i)} (\cot\alpha_{ij} + \cot\beta_{ij})(\vec{x}_j - \vec{x}_i), \tag{A.11}$$

where α_j and β_j are the two angles opposite to the edge in the two triangles that share the edge $(\vec{x}_i,\vec{x}_j)$ as shown in the Ext Fig. 1d, and $N_1(i)$ is the set of neighbors with one ring of

the vertex i.

For force inference, the interfacial mean curvatures that appears in Laplace equations are defined as the average of the mean-curvature computed on every vertices belonging to the interface.

A.2 Mechanical equilibrium in a heterogeneous foam

A.2.1 Active foam model hypotheses

We assume that the shape of the blastomeres in the early embryos of interest is primarily controlled by cortical tensions [8,9]. These surface tensions originate from cortical contractility and are considered homogeneous and isotropic at each cell-cell interface. The tension of the plasma membrane is lower, generically by an order of magnitude [8] but it may be counted in the surface tensions if necessary. The direct negative contribution to the surface tension of adhesion molecules (cadherin) is also negligible in general [8], but may also be counted in interface tensions, as long as all effective interface tensions remain strictly positive. Therefore, an assembly of cells is supposed to be akin to a heterogeneous foam-like structure but where each interface may have a different surface tension, which is controlled actively by adjacent cells. The evolution of cell shapes is assumed to be quasistatic, which means that viscous dissipation is neglected. This hypothesis is well justified by comparing the typical timescale associated with viscous relaxation of the cortex, on the order of dozens of seconds [10,11], and the timescale associated with typical morphogenetic events in early embryos, rather than of the order of dozens of minutes. Moreover, we assume that blastomeres conserve their volumes on typical morphogenetic timescales (between two divisions).

A.2.2 Active-foam 3D simulations

Lagrangian function and force on a vertex

Our numerical simulations consist of a constrained optimization of a surface energy defined on an initial nonmanifold triangular mesh. The surface energy and Lagrangian function for a set of n_c cells are defined as follows:

$$\mathcal{E} = \sum_{m=1}^{n_m} \gamma_m \mathcal{A}_m, \tag{A.12}$$

$$\mathcal{L} = \mathcal{E} - \sum_{k=1}^{n_c} p_k (\mathcal{V}_k - \mathcal{V}_k^0), \tag{A.13}$$

where γ_m and $\mathcal{A}_m$ are, respectively, the surface tension and area of the interface between the regions a_m and b_m where a_m, $b_m \in [0, n_c]$, $m \in [1, n_m]$ are an index of a membrane, which identifies a pair of regions. p_k and $\mathcal{V}_k^0$ are, respectively, the pressure and target volume value of the cell k. Note that for interfaces a_m and b_m span $[0, n_c]$, where 0 refers to the external medium, while for cells k span $[1, n_c]$.

From the Lagrangian function, one can calculate the force $\vec{f_x} = -\dfrac{\partial \mathcal{L}}{\partial \vec{x}}$ at each vertex of the mesh $\vec{x} \in \{\vec{x}_\alpha\}_{\alpha=1}^{n_v}$ as follows

$$\vec{f_x} = -\frac{\partial \mathcal{E}}{\partial \vec{x}} + \sum_{k=1}^{n_c} p_k \frac{\partial \mathcal{V}_k}{\partial \vec{x}} = -\sum_{m=1}^{n_m} \gamma_m \frac{\partial \mathcal{A}_m}{\partial \vec{x}} + \sum_{k=1}^{n_c} p_k \frac{\partial \mathcal{V}_k}{\partial \vec{x}}. \tag{A.14}$$

Lagrangian function for embryos within a shell

For the example of C. Elegans, the optimization of a surface energy must satisfy an additional constraint, that is, that every vertex must lie in the shell. We call $\mathcal{S}$ the shell, κ the stiffness of the shell, $\mathcal{F}_{el}$ the elastic force exerted by the shell on the vertices outside its bounds, defined by :

$$\mathcal{F}_{el}(\vec{x}_\alpha) = \|\vec{x}_\alpha - \vec{x}_\mathcal{S}(\vec{x}_\alpha)\|^2 \cdot \mathbf{N}_\mathcal{S}(\vec{x}_\mathcal{S}(\vec{x}_\alpha)), \tag{A.15}$$

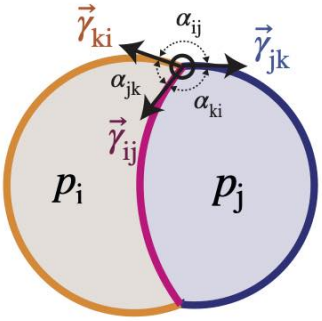
where $\mathbf{N}_\mathcal{S}(\vec{x})$ is the normal of the surface of the shell at the point $\vec{x}$ (belonging to the shell), and $\vec{x}_\mathcal{S}(\vec{x})$ is the orthogonal projection of the vertex $\vec{x}$ on the surface of the shell $\mathcal{S}$.

The surface energy and Lagrangian function for a set of N cells then take the following form

$$\mathcal{E} = \sum_{m=1}^{n_m} \gamma_m \mathcal{A}_m + \frac{\kappa}{2} \sum_{\vec{x}_\alpha \notin \mathcal{S}} \mathcal{F}_{el}(\vec{x}_\alpha), \tag{A.16}$$

$$\mathcal{L} = \mathcal{E} - \sum_{k=1}^{n_c} p_k(\mathcal{V}_k - \mathcal{V}_k^0). \tag{A.17}$$

A.2.3 Variants of the Young-Dupré tension balance



To obtain scalar values of the tensions, we first choose to fix the average surface tension to 1, with a first equation:

$$\sum_{m=1}^{n_m} \gamma_m = n_m. \tag{A.18}$$

Tension-balance gives relative relationships between surface tensions, and can be expressed in numerous ways. For all the following formulas, (i, j, k) represents three materials (cells or exterior media) that meet at a triple junction.

Young-Dupré

$$\forall (i,j,k), \quad \gamma_{ij} + \gamma_{ik}\cos\alpha_{jk} + \gamma_{jk}\cos\alpha_{ik} = 0 \tag{A.19}$$

$$\forall (i,j,k), \quad \gamma_{ij}\cos\alpha_{jk} + \gamma_{ik} + \gamma_{jk}\cos\alpha_{ij} = 0 \tag{A.20}$$

$$\forall (i,j,k), \quad \gamma_{ij}\cos\alpha_{ik} + \gamma_{ik}\cos\alpha_{ij} + \gamma_{jk} = 0 \tag{A.21}$$

Projection Young-Dupré

$$\forall (i,j,k), \quad \gamma_{ij} + \gamma_{ik}\cos\alpha_{jk} + \gamma_{jk}\cos\alpha_{ik} = 0 \tag{A.22}$$

$$\forall (i,j,k), \quad \gamma_{ik}\sin\alpha_{jk} - \gamma_{jk}\sin\alpha_{ik} = 0 \tag{A.23}$$

Lami

$$\forall(i,j,k), \quad \frac{\gamma_{ij}}{\sin\alpha_{ij}} = \frac{\gamma_{ik}}{\sin\alpha_{ik}} = \frac{\gamma_{jk}}{\sin\alpha_{jk}} \tag{A.24}$$

Inverse Lami

$$\forall(i,j,k), \quad \gamma_{ij}\sin\alpha_{ik} = \gamma_{ik}\sin\alpha_{ij} \tag{A.25}$$

$$\forall(i,j,k), \quad \gamma_{ij}\sin\alpha_{jk} = \gamma_{jk}\sin\alpha_{ij} \tag{A.26}$$

$$\forall(i,j,k), \quad \gamma_{ik}\sin\alpha_{jk} = \gamma_{jk}\sin\alpha_{ik} \tag{A.27}$$

Lami logarithm

$$\forall(i,j,k), \log\gamma_{ij} - \log\sin\alpha_{ij} = \log\gamma_{ik} - \log\sin\alpha_{ik} = \log\gamma_{jk} - \log\sin\alpha_{jk} \tag{A.28}$$

A.2.4 Mesh-based variational force balance

In this study, we propose new force-balance equations, *Variational Young-Dupré* and *Variational Laplace* equations, that can be used for tension inference. In this section we propose a detailed derivation of these equations. Our mesh, which describes the surface of the embryo, is composed of n_v vertices, n_c cells of volumes V_l^0 for $l \in 1 \dots n_c$ and of n_m membranes (or interfaces) of area A_m. The equilibrium shape is given by minimizing the energy of the membranes, which have a defined surface tension γ_m, under the constraint of volume conservation $\mathcal{V}$. We call the set of vertices of the mesh $\mathbb{V} = \{\vec{x}_\alpha\}_{\alpha=1}^{n_v}$,the set of cells $\mathcal{C}$, and the set of membranes $\mathcal{M}$.

Geometrical model

Keeping the same notation as before, the Lagrangian function that is minimized is:

$$\mathcal{L} = \sum_{m=1}^{n_m} \gamma_m \mathcal{A}_m - \sum_{k=1}^{n_c} p_k(\mathcal{V}_k - \mathcal{V}_k^0), \tag{A.29}$$

In equilibrium, we have $\dfrac{\partial\mathcal{L}}{\partial\vec{x}_\alpha} = \vec{0}$.
The equilibrium of the forces gives us, for each vertex $\vec{x}_\alpha$, the following equation :

$$\vec{0} = \sum_{m=1}^{n_m} \gamma_m \frac{\partial\mathcal{A}_m}{\partial\vec{x}} - \sum_{k=1}^{n_c} p_k \frac{\partial\mathcal{V}_k}{\partial\vec{x}}. \tag{A.30}$$

This can be summarized as a product between a huge tensor $Q \in \mathbb{R}^{n_v\times(n_m+n_c)\times 3}$ that contains the derivatives of the areas and volumes with respect to each vertex, and the matrix $\Pi \in \mathbb{R}^{n_m+n_c}$ that contains the surface tensions and pressures. This product gives the equilibrium equations for each of the vertices. The tensor Q can be conveniently expressed as a matrix of $\mathbb{R}^3$ vectors:

$$Q\times\Pi = \begin{pmatrix} \nabla_{\vec{x_1}}\mathcal{A}_1 & \dots & \nabla_{\vec{x_1}}\mathcal{A}_{n_m} & \nabla_{\vec{x_1}}\mathcal{V}_1 & \dots & \nabla_{\vec{x_1}}\mathcal{V}_{n_c} \\ \cdot & & \cdot & \cdot & & \cdot \\ \cdot & & \cdot & \cdot & & \cdot \\ \cdot & & \cdot & \cdot & & \cdot \\ \cdot & & \cdot & \cdot & & \cdot \\ \nabla_{\vec{x_{n_v}}}\mathcal{A}_1 & \dots & \nabla_{\vec{x_{n_v}}}\mathcal{A}_{n_m} & \nabla_{\vec{x_{n_v}}}\mathcal{V}_1 & \dots & \nabla_{\vec{x_{n_v}}}\mathcal{V}_{n_c} \end{pmatrix} \begin{pmatrix} \gamma_1 \\ \dots \\ \gamma_{n_m} \\ -p_1 \\ \dots \\ -p_{n_c} \end{pmatrix} = 0$$

State Equations

We will define a matrix $R \in \mathbb{R}^{(n_c+n_m)\times(n_c+n_m)} : R = Q^T Q$. We note that $R \times \Pi = 0$.

$$R = \begin{pmatrix} \sum_{i\in\mathbb{V}} \|\nabla_{\vec{x}_i}\mathcal{A}_1\|^2 & \dots & \sum_{i\in\mathbb{V}} (\nabla_{\vec{x}_i}\mathcal{A}_1 \cdot \nabla_{\vec{x}_i}\mathcal{A}_{n_m}) & \sum_{i\in\mathbb{V}} (\nabla_{\vec{x}_i}\mathcal{A}_1 \cdot \nabla_{\vec{x}_i}\mathcal{V}_1) & \dots & \sum_{i\in\mathbb{V}} (\nabla_{\vec{x}_i}\mathcal{A}_1 \cdot \nabla_{\vec{x}_i}\mathcal{V}_{n_c}) \\ \vdots & & \vdots & \vdots & & \vdots \\ \sum_{i\in\mathbb{V}} (\nabla_{\vec{x}_i}\mathcal{A}_{n_m} \cdot \nabla_{\vec{x}_i}\mathcal{A}_1) & \dots & \sum_{i\in\mathbb{V}} \|\nabla_{\vec{x}_i}\mathcal{A}_{n_m}\|^2 & \sum_{i\in\mathbb{V}} (\nabla_{\vec{x}_i}\mathcal{A}_{n_m} \cdot \nabla_{\vec{x}_i}\mathcal{V}_1) & \dots & \sum_{i\in\mathbb{V}} (\nabla_{\vec{x}_i}\mathcal{A}_{n_m} \cdot \nabla_{\vec{x}_i}\mathcal{V}_{n_c}) \\ \sum_{i\in\mathbb{V}} (\nabla_{\vec{x}_i}\mathcal{V}_1 \cdot \nabla_{\vec{x}_i}\mathcal{A}_1) & \dots & \sum_{i\in\mathbb{V}} (\nabla_{\vec{x}_i}\mathcal{V}_1 \cdot \nabla_{\vec{x}_i}\mathcal{V}_1) & \sum_{i\in\mathbb{V}} \|\nabla_{\vec{x}_i}\mathcal{V}_1\|^2 & \dots & \sum_{i\in\mathbb{V}} (\nabla_{\vec{x}_i}\mathcal{V}_1 \cdot \nabla_{\vec{x}_i}\mathcal{V}_{n_c}) \\ \vdots & & \vdots & \vdots & & \vdots \\ \sum_{i\in\mathbb{V}} (\nabla_{\vec{x}_i}\mathcal{V}_{n_c} \cdot \nabla_{\vec{x}_i}\mathcal{A}_1) & \dots & \sum_{i\in\mathbb{V}} (\nabla_{\vec{x}_i}\mathcal{V}_{n_c} \cdot \nabla_{\vec{x}_i}\mathcal{A}_{n_m}) & \sum_{i\in\mathbb{V}} (\nabla_{\vec{x}_i}\mathcal{V}_{n_c} \cdot \nabla_{\vec{x}_i}\mathcal{V}_1) & \dots & \sum_{i\in\mathbb{V}} \|\nabla_{\vec{x}_i}\mathcal{V}_{n_c}\|^2 \end{pmatrix}$$

We note that matrix R has a kernel of dimension 1. We need to add another equation to fix the tension scales (by setting the mean to 1 for example).
The state equations that we choose are $\mathcal{S} = R \times \Pi = 0$ which gives us discretized equations:

$$\forall l \in \mathcal{M}, \quad 0 = \sum_{m=1}^{n_m} \gamma_m \left(\sum_{i\in\mathbb{V}} \frac{\partial \mathcal{A}_m}{\partial \vec{x}_i} \cdot \frac{\partial \mathcal{A}_l}{\partial \vec{x}_i} \right) - \sum_{k=1}^{n_c} p_n \left(\sum_{i\in\mathbb{V}} \frac{\partial \mathcal{V}_k}{\partial \vec{x}_i} \cdot \frac{\partial \mathcal{A}_l}{\partial \vec{x}_i} \right) \tag{A.31}$$

$$\forall n \in \mathcal{C}, \quad 0 = \sum_{m=1}^{n_m} \gamma_m \left(\sum_{i\in\mathbb{V}} \frac{\partial \mathcal{A}_m}{\partial \vec{x}_i} \cdot \frac{\partial \mathcal{V}_n}{\partial \vec{x}_i} \right) - \sum_{k=1}^{n_c} p_n \left(\sum_{i\in\mathbb{V}} \frac{\partial \mathcal{V}_k}{\partial \vec{x}_i} \cdot \frac{\partial \mathcal{V}_n}{\partial \vec{x}_i} \right) \tag{A.32}$$

Matrix equations

We will note $\Gamma = (\gamma_1, \gamma_2 \dots \gamma_{n_m})^T$ the generalized vector of tensions and $\mathrm{P} = (p_1, p_2 \dots p_{n_c})^T$ the generalized vector of pressures.

We can represent the first set of discretized equations using matrix notation $\mathrm{G}_\Gamma \in \mathbb{R}^{n_m \times n_m}$, and $\mathrm{B}_P \in \mathbb{R}^{n_m \times n_c}$ verifying $\mathrm{G}_\Gamma \Gamma - \mathrm{B}_P \mathrm{P} = 0$:

$$\forall l, m \in (1 \dots n_m) \quad \mathrm{G}_\Gamma^{lm} = \sum_{i\in\mathbb{V}} \frac{\partial \mathcal{A}_m}{\partial \vec{x}_i} \cdot \frac{\partial \mathcal{A}_l}{\partial \vec{x}_i}$$

$$\forall k \in (1 \dots n_c), l \in (1 \dots n_m) \quad \mathrm{B}_P^{lk} = \sum_{i\in\mathbb{V}} \frac{\partial \mathcal{V}_k}{\partial \vec{x}_i} \cdot \frac{\partial \mathcal{A}_l}{\partial \vec{x}_i}$$

G_Γ is generally not invertible; all tensions are fixed to one parameter.
We can represent the second set of discretized equations using matrix notation $\mathrm{B}_\Gamma \in \mathbb{R}^{n_c \times n_m}$, and $\mathrm{G}_P \in \mathbb{R}^{n_c \times n_c}$ verifying $\mathrm{B}_\Gamma \Gamma - \mathrm{G}_P \mathrm{P} = 0$:

$$\forall n \in (1 \dots n_c), m \in (1 \dots n_m) \quad \mathrm{B}_\Gamma^{nm} = \sum_{i\in\mathbb{V}} \frac{\partial \mathcal{A}_m}{\partial \vec{x}_i} \cdot \frac{\partial \mathcal{V}_n}{\partial \vec{x}_i}$$

$$\forall k, n \in (1 \dots n_c) \quad \mathrm{G}_P^{nk} = \sum_{i\in\mathbb{V}} \frac{\partial \mathcal{V}_k}{\partial \vec{x}_i} \cdot \frac{\partial \mathcal{V}_n}{\partial \vec{x}_i}$$

G_P is invertible thus knowing Γ and K, we can deduce P from these equations, that we call *Variational Laplace* equations:

$$\mathrm{P} = (\mathrm{G}_P)^{-1} \mathrm{B}_\Gamma \Gamma. \tag{A.33}$$

We can write $\Pi = \begin{pmatrix} \Gamma \\ \mathrm{P} \end{pmatrix}$ and write the previous equations into a block matrix M:

$$\mathrm{M} \times \Pi = 0 \iff \begin{pmatrix} \mathrm{G}_\Gamma & -\mathrm{B}_P \\ \mathrm{B}_\Gamma & -\mathrm{G}_P \end{pmatrix} \times \begin{pmatrix} \Gamma \\ \mathrm{P} \end{pmatrix} = 0 \tag{A.34}$$

Thus, the *Variational Young-Dupré* equations for tension inference are

$$(\mathrm{G}_\Gamma - \mathrm{B}_P(\mathrm{G}_P)^{-1}\mathrm{B}_\Gamma) \times \Gamma = 0, \tag{A.35}$$

$$(1 \cdots 1) \times \Gamma = n_m. \tag{A.36}$$

This last equation fixes the average of the surface tensions to 1.

Proof that the matrix $\mathbf{G}_P$ is invertible

In the previous section, we assumed that the matrix G^p is invertible. To demonstrate this hypothesis, we have to show that the matrix of volume gradients has full rank (linearly independent columns). For simplicity, assume that each face of each discretized cell has an interior point and that the cells are numbered in "decreasing" order, from the outermost(s) to the innermost(s), more precisely, in the sense that there exists $\vec{x}_{i_1} \in \mathrm{Cell}_1$ such that $\vec{x}_{i_1} \notin \bigcup_{k>=2} \mathrm{Cell}_k$, and recursively, for all $k \in (1 \dots n_c)$, there exists $\vec{x}_{i_k} \in \mathrm{Cell}_k$ such that $\vec{x}_{i_k} \notin \bigcup_{k'>=k+1} \mathrm{Cell}_{k'}$. Now assume that the volume gradients are not linearly independent, then there exists $P_1, \dots, P_{n_c} \in \mathbb{R}$ such that for all $i \in (1 \dots n_v)$

$$\sum_{k=1}^{n_c} P_i \nabla_{\vec{x}_i} \mathcal{V}_k = 0. \tag{A.37}$$

Hence, for $i = i_1$, we simply have $P_1 \nabla_{\vec{x}_{i_1}} \mathcal{V}_1 = 0$. Furthermore, the gradient of $\mathcal{V}_1$ with respect to $\vec{x}_{i_1}$ is the sum of the gradients with respect to $\vec{x}_{i_1}$ of the volumes of the tetrahedra defined by the apex 0 and the triangles surrounding $\vec{x}_{i_1}$, so $\nabla_{\vec{x}_{i_1}} \mathcal{V}_1 \neq 0$ so that $P_1 = 0$ and

$$\sum_{k=2}^{n_c} P_i \nabla_{\vec{x}_i} \mathcal{V}_k = 0. \tag{A.38}$$

Thus, by recursion, we have $P_i = 0$ for all $i \in (1 \dots n_c)$, so that the volume gradients are linearly independent. Consequently, the matrix G^p is invertible.

A.3 Force inference

A.3.1 Hypotheses for tension and pressure inference

We assume that each pressure p is positive and put the pressure of the exterior medium at 0. The surface tensions y are also positive, and as explained before we arbitrarily put their mean at 1. This choice of scale for the tensions imposes a scale on the pressures, as appears in the Laplace law $2\gamma_{ij}H_{ij} = p_i - p_j$. However, the scale of the mean curvature H_{ij} depends on the size of the embryo. To compute the curvature with the right scaling, one would need to have a mesh that has the precise dimensions of the biological sample. This is not out of reach experimentally, but since we were interested in relative and not absolute values (absolute measurement of surface tensions being not available in any way), we found it more convenient to fix the biggest pressure at 1.

A.3.2 In silico validation

With our previously introduced 3D active foam model, we generated 47 simulations of cell clusters, ranging from 2 to 11 cells, with random tensions.
From these meshes, we generated artificial images by defining a bounding box and convolving the mesh by a Gaussian Point Spread Function. To do the convolution efficiently, we make the operation in Fourier space before doing an inverse Fourier transform to go back into the spatial domain.
A simple thresholding of these images allowed us to compute the membrane boundaries. The segmentation masks were then computed using a watershed algorithm. Masks were expanded until they merged to avoid holes inside embryos that would have perturbed meshing algorithms, and thus the computation of geometrical quantities.

A.3.3 Ordinary Least Squares

Each of the linear least squares problems encountered is solved in the same manner: we have $A \in \mathbb{R}^{n\times m}$, $X \in \mathbb{R}^m$ and $C \in \mathbb{R}^m$ such that:

$$AX = C \tag{A.39}$$

And we want to determine X from A and C.
To determine an estimator of X, we choose the ordinary least squares (OLS) method that minimizes the sum of squared residuals, that is, we want to find X such that $\|AX - C\|^2$ is minimal. The solution can be expressed in a closed form:

$$X = (A^T A)^{-1} A^T C \tag{A.40}$$

A.3.4 Ordinary Least Squares for tension inference

Form of the matrices A and C for Tension inference

The form of the matrix A and C depends on the number of equations n_{eq}, which depends on the formula choosen.

$$A = \begin{pmatrix} \alpha_{11} & \dots & \alpha_{1n_m} \\ \dots & & \dots \\ \alpha_{n_{eq}1} & \dots & \alpha_{n_{eq}n_m} \\ 1 & \dots & 1 \end{pmatrix}, \quad X = \begin{pmatrix} \Gamma \end{pmatrix}, \quad C = \begin{pmatrix} 0 \\ \dots \\ 0 \\ n_m \end{pmatrix} \tag{A.41}$$

Note that A is very sparse, as each line has only three non-zero terms (except the last one).

Weighting by the junctional length

For the tension inference equations (except for the variational Young-Dupré variant), each equation is linked to a given triple junction. The longer the junction, the better the determination of the angles of the junction will be. We decided to weight each of the equations by the length of the junction associated with it. Thus, if we consider the original unscaled matrices A and C used to infer tension, instead of minimizing $\|AX - C\|^2 = \sum_i ((AX)_i - C_i)^2$ we minimize:

$$\sum_i ((AX)_i - C_i)^2 * L_i^2, \tag{A.42}$$

where L_i is the length of the junction considered in the i^{th} equation.

Here is a table of the median relative error of the different methods of tension inference, with and without weighting of junctional length:

	Young-Dupré	Proj. Young-Dupré	Lami	Inverse Lami	Lami Log
With length weight (%)	10.1	10.4	11.7	11.0	11.8
No length weight (%)	10.8	11.7	12.7	11.9	12.4

Form with Variational equations

With *Variational Young-Dupré* equations, we can express A and C as block matrices:

$$A = \begin{pmatrix} \mathrm{G}_\Gamma - \mathrm{B}_P(\mathrm{G}_P)^{-1}\mathrm{B}_\Gamma \\ 1 \end{pmatrix}, \quad X = \begin{pmatrix} \Gamma \end{pmatrix}, \quad C = \begin{pmatrix} 0 \\ n_m \end{pmatrix} \tag{A.43}$$

A.3.5 Ordinary Least Squares for pressure inference

Variational pressure inference

The *Variational Laplace* equations offers a way to infer pressures.

$$\mathrm{G}_P\mathrm{P} = \mathrm{B}_\Gamma\Gamma. \tag{A.44}$$

It is an OLS problem with $A = \mathrm{G}_P, X = \mathrm{P}, C = \mathrm{B}_\Gamma\Gamma$.

Form of the matrices A and C for Pressure inference with Laplace Law

There is one Laplace equation for each interface, so n_m equations in total, thus $A \in \mathbb{R}^{n_m \times n_c}, X \in \mathbb{R}^{n_c}, C \in \mathbb{R}^{n_m}$. Each relation writes $p_{a_m} - p_{b_m} = \gamma_m H_m$ where a_m and b_m are the regions delimited by the interface m, of surface tension γ_m and mean curvature H_m. Thus A and C writes:

$$A = \begin{pmatrix} \beta_{11} & \dots & \beta_{1n_c} \\ \dots & & \dots \\ \beta_{n_m 1} & \dots & \beta_{n_m n_c} \end{pmatrix}, \quad X = \begin{pmatrix} \mathrm{P} \end{pmatrix}, \quad C = \begin{pmatrix} 2\gamma_1 H_{a_1 b_1} \\ \dots \\ 2\gamma_{n_m} H_{a_{n_m} b_{n_m}} \end{pmatrix} \tag{A.45}$$

,
Where $\beta_{ij} = 1$ if $j \in \{a_i, b_i\}$ and 0 otherwise.

Weighting by the interfacial area

For the pressure inference equations with Laplace's law, each equation is linked to a given interface. The larger the area, the better the determination of the curvature of the interface will be. We decided to weight each of the equations by the area of the interface associated with it.
Thus, if we consider the original unweighted matrices A and C used to infer pressure, instead of minimizing $\|AX - C\|^2 = \sum_i ((AX)_i - C_i)^2$ we will minimize:

$$\sum_i ((AX)_i - C_i)^2 * \mathcal{A}_i^2, \tag{A.46}$$

where $\mathcal{A}_i$ is the area of the interface considered in the i^{th} equation.

Here is a table of the median relative error of the different methods of pressure inference, with the tensions previously inferred using the Young-Dupré formula:

	Laplace's Law	Variational Laplace
With area weight (%)	12.5	-
No area weight (%)	12.6	12.0

A.4 Stress tensor

We consider a cell a. The cell surface is composed of membrane portion $m \in a$ of constant surface tension γ_m^a. The energy of the total foam writes $\mathcal{E} = \sum_{m=1}^{n_m} \gamma_m \mathcal{A}_m$, but alternatively, we can express the energy of the foam by summing up the contributions of each cell:

$$\mathcal{E} = \sum_a \sum_{m \in a} \gamma_m^a \mathcal{A}_m \tag{A.47}$$

For a membrane portion m that is in contact with the exterior medium, we can identify γ_m and γ_m^a. However, for a surface shared between two cells, each membrane area is counted twice, so $\gamma_m^a = \dfrac{\gamma_m}{2}$.
We note $\mathcal{V}_{a+}$ the total volume of the cell and $\mathcal{V}_{a-}$ the volume of the cell enclosed by the membrane. We note A_0 the surface of the membrane The mean stress writes :

$$\overline{\underline{\sigma}}_a = \frac{1}{\mathcal{V}_a} \int_{\mathcal{V}_{a+}} \underline{\underline{\sigma}}(\vec{x})\, \mathrm{d}V \tag{A.48}$$

We decompose it between the bulk and membrane components :

$$\overline{\underline{\sigma}}_a = \frac{1}{\mathcal{V}_a} \int_{\mathcal{V}_{a-}} \underline{\underline{\sigma}}(\vec{x})\, \mathrm{d}V + \frac{1}{\mathcal{V}_a} \left(\int_{\mathcal{V}_{a+}} \underline{\underline{\sigma}}(\vec{x}) \mathrm{d}V - \int_{\mathcal{V}_{a-}} \underline{\underline{\sigma}}(\vec{x}) \mathrm{d}V \right) \tag{A.49}$$

The stress $\underline{\underline{\sigma}}$ is constant within the cell, its value being defined by hydrostatic pressure p_a. In the cell membrane, the small thickness limit established in [12] leads to the following relation:

$$\left(\int_{\mathcal{V}_{a+}} \underline{\underline{\sigma}}(\vec{x})\, \mathrm{d}V - \int_{\mathcal{V}_{a-}} \underline{\underline{\sigma}}(\vec{x})\, \mathrm{d}V \right) = \int_{A_0} \gamma^a (\underline{\underline{\delta}} - \vec{n} \otimes \vec{n})\, \mathrm{d}S \tag{A.50}$$

We note $\underline{\underline{\delta}}$ the identity tensor and $\vec{n} = \begin{bmatrix} n_1 \\ n_2 \\ n_3 \end{bmatrix}$ and $\vec{n} \otimes \vec{n} = \begin{bmatrix} n_1 n_1 & n_1 n_2 & n_1 n_3 \\ n_2 n_1 & n_2 n_2 & n_2 n_3 \\ n_3 n_1 & n_3 n_2 & n_3 n_3 \end{bmatrix}$.
Eventually, the mean stress reads

$$\overline{\underline{\sigma}}_a = -p_a \underline{\underline{\delta}} + \sum_{m \in a} \frac{\gamma_m^a}{\mathcal{V}_a} \int_m (\underline{\underline{\delta}} - \vec{n} \otimes \vec{n})\, \mathrm{d}S \tag{A.51}$$

We can decompose each surface membrane m into individual triangles that make up it. As the normal is constant on a triangle, the integral simplifies itself :

$$\int_m (\underline{\underline{\delta}} - \vec{n} \otimes \vec{n})\, \mathrm{d}S = \sum_{t \in m} \int_t (\underline{\underline{\delta}} - \vec{n} \otimes \vec{n})\, \mathrm{d}S = \sum_{t \in m} A_t (\underline{\underline{\delta}} - \vec{n}_t \otimes \vec{n}_t) \tag{A.52}$$

$$\overline{\underline{\sigma}}_a = -p_a \underline{\underline{\delta}} + \sum_{m \in a} \frac{\gamma_m^a}{\mathcal{V}_a} \sum_{t \in m} A_t (\underline{\underline{\delta}} - \vec{n}_t \otimes \vec{n}_t) \tag{A.53}$$

We can compute it numerically from our mesh once the force inference is done and plot it on the mesh to show the principal axes of the stress tensor in the cell.

Bibliography

[1] Pierre Alliez, Clément Jamin, Laurent Rineau, Stéphane Tayeb, Jane Tournois, and Mariette Yvinec. 3D mesh generation. In *CGAL User and Reference Manual*. CGAL Editorial Board, 5.5.1 edition, 2022.

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

[3] Pauli Virtanen, Ralf Gommers, Travis E Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, et al. Scipy 1.0: fundamental algorithms for scientific computing in python. *Nature methods*, 17(3):261–272, 2020.

[4] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[5] Aric Hagberg, Pieter Swart, and Daniel S Chult. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.

[6] Jean Cousty, Gilles Bertrand, Laurent Najman, and Michel Couprie. Watershed cuts: Minimum spanning forests and the drop of water principle. *IEEE transactions on pattern analysis and machine intelligence*, 31(8):1362–1374, 2008.

[7] Mathieu Desbrun, Mark Meyer, Peter Schröder, and Alan H Barr. Implicit fairing of irregular meshes using diffusion and curvature flow. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 317–324, 1999.

[8] Jean-Léon Maître, Ritsuya Niwayama, Hervé Turlier, François Nédélec, and Takashi Hiiragi. Pulsatile cell-autonomous contractility drives compaction in the mouse embryo. *Nature cell biology*, 17(7):849–855, 2015.

[9] Jean-Léon Maître, Hervé Turlier, Rukshala Illukkumbura, Björn Eismann, Ritsuya Niwayama, François Nédélec, and Takashi Hiiragi. Asymmetric division of contractile domains couples cell positioning and fate specification. *Nature*, 536(7616):344–348, 2016.

[10] Hervé Turlier, Basile Audoly, Jacques Prost, and Jean-François Joanny. Furrow constriction in animal cell cytokinesis. *Biophysical journal*, 106(1):114–123, 2014.

[11] Nargess Khalilgharibi, Jonathan Fouchard, Nina Asadipour, Ricardo Barrientos, Maria Duda, Alessandra Bonfanti, Amina Yonis, Andrew Harris, Payman Mosaffa, Yasuyuki Fujita, et al. Stress relaxation in epithelial monolayers is controlled by the actomyosin cortex. *Nature physics*, 15(8):839–847, 2019.

[12] GK Batchelor. The stress system in a suspension of force-free particles. *Journal of fluid mechanics*, 41(3):545–570, 1970.

Appendix B

Supplementary note of chapter 4

In this Supplementary Note, we provide additional theoretical and implementation details on the Adjoint-state method procedure for embryos with line tensions.

B.1 An introduction to line tensions

Our method could be applied with success to infer line tension on embryos. It would allow the first measurement of line tensions in a biological system, which would be a fundamental landmark discovery in the study of the mechanics of biological systems.

With line tensions, the system is described by the following Lagrangian:

$$\mathcal{L} = \sum_{m=1}^{n_m} \gamma_m \mathcal{A}_m + \sum_{j=1}^{n_j} \kappa_j \mathrm{L}_j - \sum_{k=1}^{n_c} p_k(\mathcal{V}_k - \mathcal{V}_k^0), \tag{B.1}$$

The mechanical equilibrium of an active foam can be described from the following equation:

$$\vec{0} = \frac{\partial \mathcal{L}}{\partial \vec{x}_\alpha} = \sum_{m=1}^{n_m} \gamma_m \frac{\partial \mathcal{A}_m}{\partial \vec{x}_\alpha} + \sum_{j=1}^{n_j} \kappa_j \frac{\partial \mathrm{L}_j}{\partial \vec{x}_\alpha} - \sum_{k>1}^{n_c} p_k \frac{\partial \mathcal{V}_k}{\partial \vec{x}_\alpha} \tag{B.2}$$

where κ_j and L_j are, respectively, the line tension and length of the junctional line between the regions q_j, r_j and s_j where $\{q_j, r_j, s_j\} \in [\![0, N]\!]^3$, $j \in [\![1, n_j]\!]$ is an index of a junction, which identifies a triplet of regions. To simulate active foams with line tension, one just need to add the line tensions as input parameters, and compute the gradient of the junctional length.

We will start by studying the case of the doublet with line tension, to see how force-balance equations are affected by the presence of line tensions.

B.1.1 The doublet with line tensions

Lets take back the parametrization of the doublet that we introduced in the chapter 2, and see what happens when we add line tension at the junction. The surface tensions of each surfaces are denoted by t_1, t_2, t_3. The line tension at the junction between the two cell is denoted by κ. The respective volumes of cells 1 and 2 are denoted by $\mathcal{V}_1$ and $\mathcal{V}_2$.

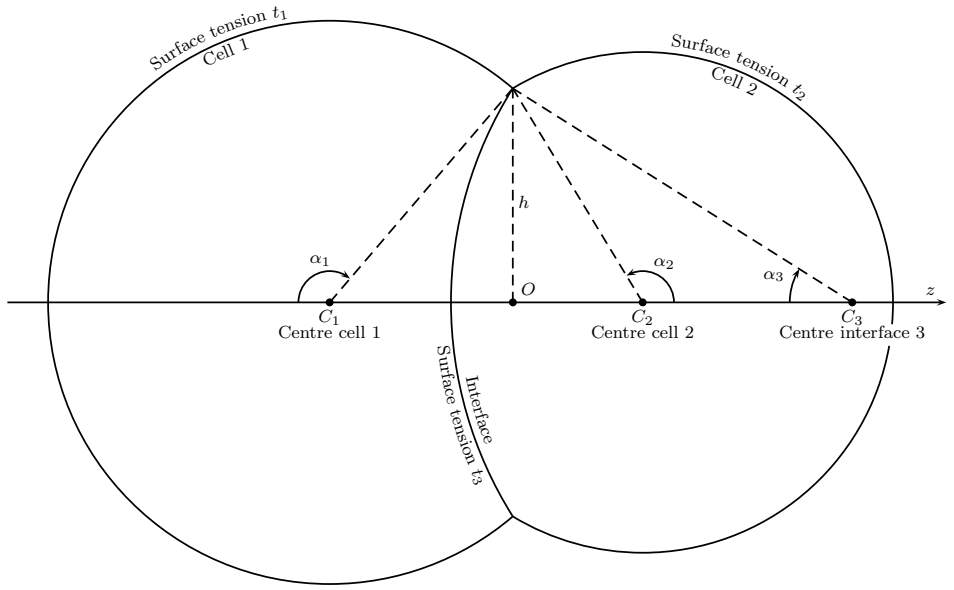


Figure B.1: Cells configuration

Configuration at minimum energy

We are interested in finding the configuration of the cells at minimum energy of the system. The energy $\mathcal{E}$ of the system is given by the sum of surface tensions times the areas, that is

$$\mathcal{E} = 2\pi h^2 \left(\frac{t_1}{1+c_1} + \frac{t_2}{1+c_2} + \frac{t_3}{1+c_3} \right) + 2\pi h\kappa,$$

where

$$c_1 = \cos\alpha_1, \qquad s_1 = \sin\alpha_1, \qquad c_2 = \cos\alpha_2, \qquad s_2 = \sin\alpha_2, \qquad c_3 = \cos\alpha_3, \qquad s_3 = \sin\alpha_3.$$

The volumes of cells 1 and 2 are given by

$$\mathcal{V}_1 = 2\pi h^3 \left(-\frac{s_1(2+c_1)}{(1+c_1)^2} + \frac{s_3(2+c_3)}{(1+c_3)^2} \right),$$
$$\mathcal{V}_2 = 2\pi h^3 \left(\frac{s_2(2+c_2)}{(1+c_2)^2} - \frac{s_3(2+c_3)}{(1+c_3)^2} \right),$$

In the following of the documents, for sake of simplicity, we define $E = \dfrac{\mathcal{E}}{2\pi}$ and $v_k = \dfrac{\mathcal{V}_k}{2\pi}$ for $k \in \{1, 2\}$ so that we have

$$E = h^2 \left(\frac{t_1}{1+c_1} + \frac{t_2}{1+c_2} + \frac{t_3}{1+c_3} \right) + h\kappa, \tag{B.3}$$

and

$$v_1 = h^3 \left(-\frac{s_1(2+c_1)}{(1+c_1)^2} + \frac{s_3(2+c_3)}{(1+c_3)^2} \right),$$
$$v_2 = h^3 \left(\frac{s_2(2+c_2)}{(1+c_2)^2} - \frac{s_3(2+c_3)}{(1+c_3)^2} \right).$$

To impose conservation of the volume, we will find the solution of the Lagrangian:

$$\mathcal{L}(h, \alpha_1, \alpha_2, \alpha_3) = E - P_1(v_1 - v_1^0) - P_2(v_2 - v_2^0) \tag{B.4}$$

Systems for the possible local extrema

Deriving the Lagrangian $E - P_1 w_1 - P_2 w_2$ with respect to α_1 and α_2, we easily get the Lagrange multipliers

$$P_1 = -\frac{t_1 s_1}{3h}, \qquad P_2 = \frac{t_2 s_2}{3h}.$$

Deriving with respect to h and α_3 and replacing by the values of P_1 and P_2, we get the balance of forces at the joining points

$$t_1 c_1 + t_2 c_2 + t_3 c_3 + \frac{\kappa}{h} = 0, \qquad t_1 s_1 + t_2 s_2 + t_3 s_3 = 0.$$

Thus, the introduction of a line tension leads to an additional term in the Young-Dupré equation. This additional term is the line tension divided by the radius of curvature of the junction, and is directed towards the center of curvature of the junction.

Generalisation to arbitrary cell configurations

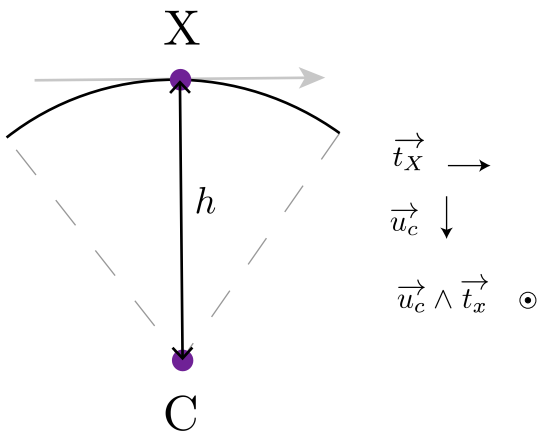


Figure B.2: Notations

With a number of cell higher than two, the energy keeps the same form so the same energetical derivation can be done. For n_c cells, n_m membranes of area A_i and n_j junctions of length l_i (on which there are line tensions) :

$$\mathcal{E} = \sum_{m \in 1 \dots n_m} t_m A_m + \sum_{i \in 1 \dots n_j} \kappa_i l_i$$

from which we can always deduce generalized Young-Dupré equations for each junction between the cells (i, j, k), of circumradius h and of line tension κ_{ijk} :

$$\overrightarrow{t_{ij}} + \overrightarrow{t_{jk}} + \overrightarrow{t_{ik}} + \frac{\kappa_{ijk}}{h} \overrightarrow{u_c} = \overrightarrow{0}$$

Where $\overrightarrow{u_c}$ is the unit vector linking the point of the junction considered X and the circumcenter of the junction C :

$$\overrightarrow{u_c} = \frac{\overrightarrow{XC}}{||\overrightarrow{XC}||}$$

$\overrightarrow{t_X}$ is the tangent vector of the junction, considered at the point X We can project this relation on $\overrightarrow{u_c}$ and on $\overrightarrow{u_c} \wedge \overrightarrow{t_x}$ to obtain :

$$t_{ij}cos(\theta_{ij}) + t_{jk}cos(\theta_{jk}) + t_{ik}cos(\theta_{ik}) + \frac{\kappa_{ijk}}{h} = 0$$

$$t_{ij}sin(\theta_{ij}) + t_{jk}sin(\theta_{jk}) + t_{ik}sin(\theta_{ik}) = 0$$

Where the angles θ_{ab} are defined on the figure bellow

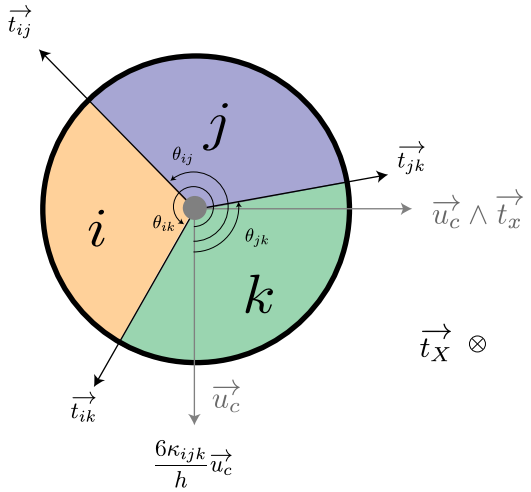


Figure B.3: Definition of the angles at a triple junction

B.2 State Equations with line tensions

To find the state equations in this case, we first rewrite the equilibrium as a product between a huge tensor $Q \in \mathbb{R}^{n_v \times (n_m+n_c+n_j)\times 3}$ that contains the derivatives of the areas and volumes and junctional length with respect to each vertex, and the matrix $\Pi \in \mathbb{R}^{n_m+n_c+n_j}$ that contains the surface tensions, the line tensions and pressures. The tensor Q can be conveniently expressed as a matrix of $\mathbb{R}^3$ vectors:

$$Q = \begin{pmatrix} \nabla_{\vec{x_1}}\mathcal{A}_1 & \dots & \nabla_{\vec{x_1}}\mathcal{A}_{n_m} & \nabla_{\vec{x_1}}\mathcal{V}_1 & \dots & \nabla_{\vec{x_1}}\mathcal{V}_{n_c} & \nabla_{\vec{x_1}}\mathrm{L}_1 & \dots & \nabla_{\vec{x_1}}\mathrm{L}_{n_j} \\ \cdot & & \cdot & \cdot & & \cdot & \cdot & & \cdot \\ \cdot & & \cdot & \cdot & & \cdot & \cdot & & \cdot \\ \cdot & & \cdot & \cdot & & \cdot & \cdot & & \cdot \\ \cdot & & \cdot & \cdot & & \cdot & \cdot & & \cdot \\ \nabla_{\vec{x_{n_v}}}\mathcal{A}_1 & \dots & \nabla_{\vec{x_{n_v}}}\mathcal{A}_{n_m} & \nabla_{\vec{x_{n_v}}}\mathcal{V}_1 & \dots & \nabla_{\vec{x_{n_v}}}\mathcal{V}_{n_c} & \nabla_{\vec{x_1}}\mathrm{L}_1 & \dots & \nabla_{\vec{x_1}}\mathrm{L}_{n_j} \end{pmatrix}$$

and the matrix P is expressed as $\Pi = \begin{pmatrix} \gamma_1 \dots \gamma_{n_m}, & -p_1 \dots - p_{n_c}, & \kappa_1 \dots \kappa_{n_j} \end{pmatrix}^{\mathrm{T}}$

Then as before, we define a matrix $R \in \mathbb{R}^{(n_m+n_c+n_j)\times(n_m+n_c+n_j)}$: $R = Q^T Q$. $R \times \Pi = 0$, thus we can establish three sets of equations:

$$\forall l \in [\![1 \dots n_m]\!], \quad 0 = \sum_{m\in\mathcal{M}} \gamma_m \left(\sum_i \frac{\partial \mathcal{A}_m}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{A}_l}{\partial \vec{x}_\alpha}\right) + \sum_{j\in\mathcal{J}} \kappa_j \left(\sum_i \frac{\partial \mathrm{L}_j}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{A}_l}{\partial \vec{x}_\alpha}\right) - \sum_{n\in\mathcal{C}} p_n \left(\sum_i \frac{\partial \mathcal{V}_n}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{A}_l}{\partial \vec{x}_\alpha}\right) \tag{B.5}$$

$$\forall k \in [\![1 \dots n_c]\!], \quad 0 = \sum_{m \in \mathcal{M}} \gamma_m \left(\sum_i \frac{\partial \mathcal{A}_m}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{V}_k}{\partial \vec{x}_\alpha}\right) + \sum_{j \in \mathcal{J}} \kappa_j \left(\sum_i \frac{\partial \mathrm{L}_j}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{V}_k}{\partial \vec{x}_\alpha}\right) - \sum_{n \in \mathcal{C}} p_n \left(\sum_i \frac{\partial \mathcal{V}_n}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{V}_k}{\partial \vec{x}_\alpha}\right) \tag{B.6}$$

$$\forall q \in [\![1 \dots n_j]\!] \quad 0 = \sum_m \gamma_m \left(\sum_i \frac{\partial \mathcal{A}_m}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathrm{L}_q}{\partial \vec{x}_\alpha}\right) + \sum_j \kappa_j \left(\sum_i \frac{\partial \mathrm{L}_j}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathrm{L}_q}{\partial \vec{x}_\alpha}\right) - \sum_n p_n \left(\sum_i \frac{\partial \mathcal{V}_n}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathrm{L}_q}{\partial \vec{x}_\alpha}\right) \tag{B.7}$$

We define nine matrices, $\mathrm{G}_\Gamma^{lm} = \sum_{\vec{x}_\alpha} \frac{\partial \mathcal{A}_m}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{A}_l}{\partial \vec{x}_\alpha}, \mathrm{G}_P^{lk} = \sum_{\vec{x}_\alpha} \frac{\partial \mathcal{V}_k}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{A}_l}{\partial \vec{x}_\alpha}, \mathrm{G}_\kappa^{lj} = \sum_{\vec{x}_\alpha} \frac{\partial \mathrm{L}_j}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{A}_l}{\partial \vec{x}_\alpha}, \mathrm{B}_\Gamma^{nm} = \sum_{\vec{x}_\alpha} \frac{\partial \mathcal{A}_m}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{V}_n}{\partial \vec{x}_\alpha}, \mathrm{B}_P^{nk} = \sum_{\vec{x}_\alpha} \frac{\partial \mathcal{V}_k}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{V}_n}{\partial \vec{x}_\alpha}, \mathrm{B}_\kappa^{nj} = \sum_{\vec{x}_\alpha} \frac{\partial \mathrm{L}_j}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathcal{V}_n}{\partial \vec{x}_\alpha}, \mathrm{H}_\Gamma^{qm} = \sum_{\vec{x}_\alpha} \frac{\partial \mathcal{A}_m}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathrm{L}_q}{\partial \vec{x}_\alpha}, \mathrm{H}_P^{qk} = \sum_{\vec{x}_\alpha} \frac{\partial \mathcal{V}_k}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathrm{L}_q}{\partial \vec{x}_\alpha}, \mathrm{H}_\kappa^{qj} = \sum_{\vec{x}_\alpha} \frac{\partial \mathrm{L}_j}{\partial \vec{x}_\alpha} \cdot \frac{\partial \mathrm{L}_q}{\partial \vec{x}_\alpha}$, and uses them to reexpress our equations using the following block matrix:

$$\begin{pmatrix} \mathrm{G}_\Gamma & -\mathrm{G}_P & \mathrm{G}_\kappa \\ \mathrm{B}_\Gamma & -\mathrm{B}_P & \mathrm{B}_\kappa \\ \mathrm{H}_\Gamma & -\mathrm{H}_P & \mathrm{H}_\kappa \end{pmatrix} \times \begin{pmatrix} \Gamma \\ \mathrm{P} \\ \mathrm{K} \end{pmatrix} = 0, \tag{B.8}$$

$\mathrm{G}_{\Gamma,P,\kappa}$ are matrices of sizes respectively n_m^2, $n_m \times n_c$ and $n_m \times n_j$, $B_{\Gamma,P,\kappa}$ are rectangular matrices of sizes respectively $n_c \times n_m$, n_c^2 and $n_c \times n_j$, and $H_{\Gamma,P,\kappa}$ are rectangular matrices of sizes respectively $n_j \times n_m$, $n_j \times n_c$ and n_j^2. The central line gives the *Variational Laplace* equations:

$$\mathrm{B}_P \mathrm{P} = B_\Gamma \Gamma + B_\kappa K \tag{B.9}$$

In the appendix, we demonstrate that the matrix B_P is invertible. We can thus write a closed-form linear system for the tensions and line tensions alone, that constitutes our state equations, the *Variational Young-Dupré* equations:

$$\boxed{\begin{pmatrix} \mathrm{G}^\gamma - \mathrm{G}^p (\mathrm{B}^p)^{-1} \mathrm{B}^\gamma & \mathrm{G}^\kappa - \mathrm{G}^p (\mathrm{B}^p)^{-1} \mathrm{B}^\kappa \\ \mathrm{H}^\gamma - \mathrm{H}^p (\mathrm{B}^p)^{-1} \mathrm{B}^\gamma & \mathrm{H}^\kappa - \mathrm{H}^p (\mathrm{B}^p)^{-1} \mathrm{B}^\kappa \end{pmatrix} \times \begin{pmatrix} \Gamma \\ \mathrm{K} \end{pmatrix} = 0} \tag{B.10}$$

As before, we can in theory use these state equations to solve the inverse problem. The first guess can be based on the tension inference, without line tension, and the optimization will thus learn the right line tensions necessary to represent the mesh properly.

RÉSUMÉ

De nouveaux paradigmes d'imagerie par microscopie ont engendré une profusion de données, en deux et trois dimensions, avec une résolution spatiale et temporelle inégalée. Pour comprendre quantitativement les phénomènes mécaniques qui déterminent la forme des cellules et des embryons, il est nécessaire de faire correspondre les informations issues de ces images avec des modèles physiques, tâche désignée sous l'appellation de résolution d'un "problème inverse". C'est l'objectif principal de cette thèse. Dans la première partie, nous établissons une analogie entre les premiers stades du developpement de l'embryon et des modèles de mousses hétérogènes. Nous explorons les complexités de la simulation de ces mousses, déduisant les tensions de surface et les pressions à partir des images de deux manières : d'abord par l'intermédiaire de la minimisation de l'erreur quadratique moyenne des equations d'équilibre physique, puis grâce à une procédure d'optimisation plus sophistiquée où les gradients sont calculés en utilisant la méthode de l'état adjoint. Par la suite, nous présentons un nouvel outil pour l'analyse d'image de microscopie: alphaMic, un microscope artificiel qui génère des images 3D à partir de modèles numériques. Cet outil permet de tester des algorithmes d'analyse d'images et de générer des données artificielles pour entrainer des réseaux de neurones profonds. Sa version différentielle, deltaMic, est capable de determiner, a partir de vraies images, la geometrie de cellules et les caractéristiques optiques d'un microscope. Enfin, nous présentons une méthode basée sur le transport optimal pour inferer les flux de membrane à partir de kymographes.

MOTS CLÉS

Microscopie, Modélisation 3D, développement embryonnaire, problème inverse, simulation de mousses, méthode de l'état adjoint, réseaux de neurones profonds, transport optimal, biologie cellulaire, Biophysique.

ABSTRACT

Novel microscopy paradigms have given rise to an abundance of data, both in two and three dimensions, with an unprecedented spatial and temporal resolution. A quantitative comprehension of the mechanical phenomena shaping cells and embryos requires to fit information obtained from these images to physical models, a task referred to as solving an "Inverse problem". This is the central focus of this thesis. In the initial part, we establish an analogy between early embryos and heterogeneous foams. We delve into the complexities of foam simulation, inferring surface tensions and pressures from images in two ways, first by minimizing the mean-square error of physical equations, then using a more sophisticated optimization procedure in which gradients are calculated with the adjoint-state method. Subsequently, we introduce another tool for microscopy image analysis: alphaMic, an artificial microscope, that generates 3D images from numerical models. This novel tool provides the capability to benchmark algorithms, produce artificial data to train deep-neural networks. Its differentiable version, deltaMic, is able to fit geometry and point spread function models from real microscopy images. Ultimately, we present a method based on optimal transport for inferring membrane flows from kymographs.

KEYWORDS

Microscopy, 3D Modeling, embryonic development, inverse problem, foam simulation, adjoint state method, deep neural networks, optimal transport, cellular biology, Biophysics.