University of Groningen

On the road to simulating life with molecular detail

Bruininks, Bart

DOI:
[10.33612/diss.169939156](https://doi.org/10.33612/diss.169939156)

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date:
2021

[Link to publication in University of Groningen/UMCG research database](#)

Citation for published version (APA):
Bruininks, B. (2021). *On the road to simulating life with molecular detail*. [Thesis fully internal (DIV), University of Groningen]. University of Groningen. <https://doi.org/10.33612/diss.169939156>

Copyright

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

The publication may also be distributed here under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license. More information can be found on the University of Groningen website: <https://www.rug.nl/library/open-access/self-archiving-pure/taverne-amendment>.

Take-down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.

On the road to simulating life with molecular detail

On the road to simulating life with molecular detail

Proefschrift

ter verkrijging van de graad van doctor aan de
Rijksuniversiteit Groningen
op gezag van de
rector magnificus prof. dr. C. Wijmenga
en volgens besluit van het College voor Promoties.
De openbare verdediging zal plaatsvinden op
vrijdag 21 mei 2021 om 16.15 uur

door

Bart Marlon Herwig Bruininks

geboren op 4 mei 1990
te Winsum

Dit proefschrift is goedgekeurd door de

promotor: prof. dr. Siewert-Jan Marrink

copromotor: dr. Alex H. de Vries

supervisor: dr. Paulo C. Telles de Souza

Samenstelling promotiecommissie:

Prof. dr. B. Poolman

Prof. dr. P. A. J. Hilbers

Prof. dr. A. Aksimentiev

Prof. dr. Siewert-Jan Marrink en dr. Paulo C. Telles de Souza hebben in belangrijke mate aan de totstandkoming van het proefschrift bijgedragen.

rijksuniversiteit
 groningen

Keywords: Molecular Dynamics, Martini, Lipids, Biomolecules

Printed by: Ridderprint

Front & Back: Randy Wind, Native Development

Copyright © 2021 by Bart M. H. Bruininks

*This thesis is like nature,
its constituents are all over the place.*

The author

Contents

Contents	vii
Preface	xi
0.1 Taking a closer look at nature	xi
0.1.1 Currently theoretically impossible.	xii
0.1.2 Currently theoretically possible	xiv
0.2 Outline of this thesis.	xiv
0.3 References	xv
1 What it means to be an atom	1
1.1 The atom.	2
1.2 An atomistic toy model	3
1.2.1 Start with placing some marbles.	4
1.2.2 Movement	5
1.2.3 Heat pressure space and density	5
1.2.4 Taking a closer look	7
1.2.5 Predictions and validations	9
1.2.6 Formalizing the marbles on rails model	10
1.2.7 Implementation.	14
1.2.8 Weaknesses of our current model	16
1.2.9 To conclude	17
1.3 References	19
2 A Practical View of the Martini Force Field	21
2.1 Introduction.	22
2.2 Hands-On: cationic lipid-DNA lipoplexes for gene transfer	25
2.2.1 Building a liquid lipoplex crystal.	26
2.2.2 Solvating a liquid lipoplex crystal	30
2.2.3 Lipoplex fusion with an asymmetric complex membrane .	32
2.3 Outlook	34
2.4 Notes	35
2.4.1 Limited stability of fluid phase and water freezing prob-	
lem	35
2.4.2 Entropy-enthalpy compensation	35
2.4.3 "Sticky problem" in sugars and proteins	35
2.4.4 Electrostatic interactions and implicit screening	36
2.4.5 Fixed structure for proteins and nucleic acids	36
2.4.6 Time step	37
2.4.7 Effective timescale	38

2.5	Acknowledgments	38
2.6	References	39
3	CHARMM-GUI Martini Maker for modeling and simulation of complex bacterial membranes with lipopolysaccharides	45
3.1	Introduction	47
3.2	Methods	48
3.2.1	CHARMM-GUI implementation.	48
3.2.2	Martini force fields used in this study	49
3.2.3	Bilayer systems	50
3.2.4	Nanodisc systems	50
3.2.5	Vesicle systems	52
3.2.6	Micelle and random systems	52
3.3	Results and discussion	53
3.3.1	Bilayer systems	53
3.3.2	Nanodisc systems	54
3.3.3	Vesicle systems	55
3.3.4	Micelle and random systems	57
3.4	Conclusions	59
3.5	Acknowledgments	59
3.6	Supplementary Information	61
3.7	References	70
4	A molecular view on the escape of lipoplexed DNA from the endosome	77
4.1	Introduction	78
4.2	Results and discussion	78
4.2.1	Construction and validation of the lipoplex model	78
4.2.2	Two distinct mechanisms leading to gene transfection.	79
4.2.3	DxTAP unsaturation is mandatory for efficient transfection	82
4.2.4	Target membrane composition severely affects fusion rate	83
4.2.5	Larger lipoplexes are more stable	84
4.2.6	Escape from the endosome	86
4.2.7	Conclusion	87
4.3	Materials and methods	88
4.3.1	Building the small periodic lipoplex	88
4.3.2	Building the small solvated lipoplex.	89
4.3.3	Building the large lipoplex	89
4.3.4	Building the lipoplex-bilayer system (small)	89
4.3.5	Building the lipoplex-bilayer system (large)	90
4.3.6	Building the lipoplex-vesicle system (large)	90
4.3.7	Analysis	91

4.4	Acknowledgments	91
4.5	Supplementary Information	93
4.6	References	97
5	Characterization and skin cell membrane penetration of a multicomponent lipid nanodroplet loaded with vitamins	103
5.1	Introduction	104
5.2	Methods	105
5.2.1	Building the lipid nanodroplet	105
5.2.2	Building the skin bilayer models.	106
5.2.3	Fusion simulations.	107
5.2.4	CG models.	107
5.2.5	Simulation parameters	107
5.2.6	Analysis of nanodroplet and fusion simulations	108
5.3	Results	108
5.3.1	Assembling the nanodroplet	108
5.3.2	Characterization of the nanodroplet.	109
5.3.3	Fusion with skin membrane models	110
5.4	Discussion and Conclusion.	115
5.5	Acknowledgments	116
5.6	Supplementary Information	117
5.7	References	120
6	Sequential voxel-based leaflet segmentation of complex lipid morphologies	127
6.1	Introduction	128
6.2	Design and implementation	129
6.2.1	Pseudo code for spatial segmentation of leaflets	129
6.2.2	Pseudo code for temporal segmentation	131
6.3	Results	133
6.3.1	Performance.	141
6.4	Discussion.	142
6.4.1	Future development of MDVoxelSegmentation	142
6.4.2	Potential extra features	142
6.4.3	How to contribute	143
6.4.4	To conclude	143
6.5	Methods	144
6.5.1	Molecular dynamics simulations.	144
6.5.2	Leaflet segmentation.	144
6.5.3	Lipid flip-flop	145
6.5.4	Visualization	145
6.6	Acknowledgements.	145
6.7	Supplementary Information	146
6.7.1	Bilayer pore	146
6.7.2	Protein bilayer	146
6.7.3	Lipid phase transition	146

6.7.4 Lipoplex transfection.	146
6.7.5 Acyl chain dinner.	146
6.7.6 Segmentation of other amphipathic systems	147
6.7.7 Self assembly of a DPPC bilayer	147
6.8 References	148
7 Summary and Outlook	151
7.1 What has been done	151
7.2 What is to come.	152
Acknowledgements	155

Preface

My personal considerations on the atomistic worldview

Dear reader,

I would like to thank you for picking up this booklet and I will share with you some philosophical ideas before we dive into hard science. To me consciousness and free will are some of the most interesting topics in life, therefore I will focus on explaining why they are so tough to appreciate within the atomistic paradigm. My aim with this preface is to make it clear what (currently) is and isn't possible using this mechanical world view. These are my personal considerations and I stand behind them, however any questions or remarks on the topic are welcome.

I hope this prelude finds you in good health,

*Bart Marlon Herwig Bruininks
Groningen, October 2020*

For millennia people have been observing nature. Religion was often used in the days of old to explain natural phenomena which could not be explained otherwise. Over time an increasing understanding of the world around us replaced many Gods. Phenomena such as lightning and earthquakes are no longer assigned to them and in some cases we are even capable of preventing or causing such phenomena. Lightning is understood as a form of electrical discharge and earthquakes are well described as shock waves originating from released stress[1, 2]. Life, as complicated as it may be, has undergone a similar evolution of understanding. The current scientific belief no longer requires a soul to live, nor is intelligent design required for complexity. For life is commonly regarded as an *emergent* property of nature itself. This view treats life as a machine, which has to obey the laws of nature[3]. Therefore, to understand life, we have to understand nature.

0.1. Taking a closer look at nature

Humans are considered a part of nature, and since you are most likely a human yourself, we could not think of a better subject to pick for a short consideration of this mechanical view. Stating that humans are machines might feel a bit short sighted at first, for humans feel like they have a free will and machines clearly do not (right?).

However, a quick inspection of the two might reveal some interesting similarities. We will start with the car. One could say a car is a machine, for it does not choose to drive, it just does what humans instruct it to do. Inside the car is an engine, which is capable of transforming chemical energy (the fuel) into mechanical energy (the rotation of the drive shaft). This in turn causes the wheels to rotate which, due to friction with the ground, causes the car to move. The gas pedal as well as the steering wheel, might be inside the car, but it is clearly not up to the car to decide the speed nor direction of movement. A modern car might be able to influence such parameters as speed and direction. However we can explain that behaviour as being controlled by a logic circuit, which was designed by humans. Upon taking a closer look at the individual components and the materials they are made of, we would find that the main materials used are aluminium, magnesium, carbon fiber, copper, steel, glass, rubber, and plastics. Where aluminium, magnesium and copper are atoms themselves, the others are compositions of atoms in molecules or other forms of larger assemblies of atoms. These atoms are well known building blocks of nature and their behaviour is understood relatively well. No free will was reliably detected in any atom as of yet.

Clearly a car is a machine and humans are not. Humans contain a free will which guides their actions. Our free will is driven by thought, which is forged in our brain. Our brain is made out of brain tissues containing a variety of brain cells. These brain cells are composed of organic molecules mainly containing oxygen, carbon, hydrogen, nitrogen, calcium, and phosphorus atoms. As it turns out to be, cars and humans are both made out of atoms, the building blocks of nature. Apparently atoms have such properties, that if you place them together in a complex mixture they can start to behave in complex manners. Back to the free will. Proving that the free will does or does not exist is non-trivial at this point. Nevertheless, it might be interesting to consider the following. Like an engine or a brain cell, the free will is an emergent property of a complex assembly of atoms. This hints at the possibility that no free will can ever exist without the body and its experiences. Such a concept of the free will might be striking to some and inspire a radical change in behaviour. On the other hand, explaining behaviour from an atomistic point of view is not very practical and rarely results in an enhanced understanding of specific behaviour. The sheer amount of computations involved to look at the free will from an atomistic point of view quickly becomes incomputable.

0.1.1. Currently theoretically impossible

The behaviour of the atoms, amongst others, depends on their positions and velocities. In the case of a human being so many in fact, that computers are needed to perform the calculations. To model the behaviour of atoms with a computer we would at least have to be able to access their positions and velocities. To get a better understanding of what we want to achieve, we will perform some back-of-the-envelope calculations. We live in a three dimensional (3D) world and therefore we require three coordinates for both positions and velocities. To put things in perspective, there are

Figure 0.1 | The relative scales of life and its components. This image was reused from Environmental Biology by Matthew R. Fisher and is licensed under CC BY 4.0.

roughly 10^{13} atoms in a typical human cell¹ and around $3 \cdot 10^{13}$ cells in a human body[5] (Fig. 0.1). Therefore we would have to keep track of $1.8 \cdot 10^{27}$ values. The most powerful supercomputer to date (Summit) only has $1.4 \cdot 10^{17}$ flops (floating point operations per second)[6]. This means that inspecting all the atomic positions and velocities once, using this computer, would require 128.6 years. If that still sounds reasonable, you have to consider the fact that a simulation requires around 10^{15} iterations to simulate one second. This would take 128,600,000,000,000 years.

Evidently, trying to model a human being, at this level of detail, is out of our reach for many years to come or might never be possible at all. Understanding and modelling the free will from an atomistic point of view is extremely difficult. The sheer amount of data involved would be incomputable. Nevertheless, would it not be marvelous if the free will can be understood as to be comparable to a tree or a dishwasher from an atomistic point of view?

¹A human cell has a mass in the order of nanograms (10^{-12}). For convenience the amount of atoms per cell was taken to be the same as the amount of cells in a human body, a common rule of thumb. Nevertheless, the amount of atoms in the human body hovers around $10^{27} - 10^{28}$ [4]. Counting all cells in the human body involves too much counting and only estimates are provided based on tissue averages. The exact number of atoms is easier to estimate, for we can chemically measure the amount of each atom present in a human body.

0.1.2. Currently theoretically possible

Describing the free will from an atomistic point of view will be left for future generations and something we will not return to in this thesis, for it clearly lies out of reach (for now). However, subcomponents of humans might just be doable. A single cell consists of 10^{13} atoms and we potentially have 10^{17} flops available. Although the exact modelling of the physics of atoms is more involved than just reading their position and velocities, this at least indicates that we would be able to perform operations on such a dataset and expect results within our lifetime. At the time of writing, Molecular Dynamics (MD) simulations of atomistic resolution are limited to subsections of cells over time scales in the order of microseconds. Nevertheless the first attempts of molecular minimal cell simulations have been granted funding and results are expected in the coming years.

0.2. Outline of this thesis

This thesis can be subdivided into four main issues. Issue one is the theoretical framework expanding on how to model nature from a molecular perspective (chapter 1 and 2). Next, the end of chapter 2 and chapter 3 as a whole, focuses on how to prepare a system so simulation can be performed. Then, data with biological relevance is generated and interpreted which is presented in chapter 4 and 5. The last issue revolves around the increasing complexity of the analysis as the complexity of the systems increases (chapter 6). In the summary and outlook we reflect on the achieved and expand on what the future of modelling life with molecular detail will look like. Finally, I will thank everyone who helped shaping me and this thesis in the acknowledgments.

0.3. References

- [1] B. Franklin, *Experiments and observations on electricity* (Printed and sold by E. Cave, at St. John's Gate, 1751).
- [2] M. Wyss, *Towards a Physical Understanding of the Earthquake Frequency Distribution*, Geophysical Journal of the Royal Astronomical Society **31**, 341 (1973).
- [3] J. Gayon, *Defining life: Synthesis and conclusions*, Origins of Life and Evolution of Biospheres **40**, 231 (2010).
- [4] *Amount of atoms in a human body*, <http://book.bionumbers.org/what-is-the-elemental-composition-of-a-cell/>, accessed: 2020-10-19.
- [5] R. Sender, S. Fuchs, and R. Milo, *Revised estimates for the number of human and bacteria cells in the body*, PLOS Biology **14**, 1 (2016).
- [6] *Performance summit super computer*, <https://www.top500.org/lists/top500/2019/06/>, accessed: 2020-10-19.

1

What it means to be an atom

You've explained something if you are finished talking.

Liesbeth Couwenberg

The idea of the atom as the fundamentally smallest conceptual particle has been around for millenia and the earliest documented considerations to our knowledge stem from the ancient Greek and Indian philosophers around 500 BCE (Democritus, Kanada)[1]. In this chapter we will first introduce the concept of the atom and then use this concept to make an atomistic toy model and use it to explain parts of our perceived reality. Whenever we talk about atoms in this section we mean the philosophical concept, not the actual modern atoms presented in the periodic table (unless stated otherwise). The main thing we should keep in mind is that we assume everything is made of atoms for atoms are the fundamentally smallest particles. At the end of this chapter the reader should understand how to build a physical model based on observations. The example model is not meant to be the most accurate description of reality, instead it focuses on obtaining a qualitative model which illustrates the fundamental interplay between individual motions and system properties such as temperate and pressure.

1

1.1. The atom

An atom has no nose, nor does it have eyes, ears, fingers or a tongue. It has no brain nor intention. For now let's say it just is. It is an entity in space. Like a point or maybe a small sphere. That would probably be all we would know about a universe with only one atom. Luckily for us, our universe contains lots of atoms. This fact brings some new interesting features. For example, an atom cannot be too close to another atom (exclusion). Some atoms might attract each other, whilst others repel. These interactions act as forces on the atoms causing them to move. The manner in which atoms attract or repel each other could be regarded as a part of their tiny perception. Atoms possess enough information to react to their environment and they are continuously reconfiguring to find their best state¹. Besides pairwise interactions, atoms gain some other interesting forms of interactions when they come with more than one. They can be linked together by so-called bonds. These bonds could be imagined as a force preventing the connected atoms from moving away from each other and is dependent on the distance. If more than two atoms are connected to each other they might prefer a certain angle, while four or more atoms could prefer a specific dihedral (Fig. 1.1).

Figure 1.1 | The geometric definitions of the bonded interactions between atoms.

Such assemblies are called molecules, still potentially very small but as opposed to atoms, molecules are divisible. Every atom also has some mass, thus gravity is introduced. However, due to the extremely small mass of our systems, and the weakness of gravity it is usually disregarded. To illustrate, consider the following to understand how weak gravity is. When you lift a paperclip with a small refrigerator magnet, the magnetic force between the paperclip and the magnet is bigger than all of the gravity of the earth 'pulling' on the paperclip, otherwise the paperclip would fall.

The description of all the atoms and the manner in which they interact with each other is called a *force field*. The true nature of these force fields lies in the realm of Quantum Mechanics, which will not be addressed in this thesis². However, in practice the description of atoms and the manner in which they interact is based on both Quantum Mechanical (*ab initio*) and experimental data (*in situ*). Careful characterization of the actual atoms of life has taken place over the past decades

¹No machine can be driven by the movement of the atoms after reaching this best state.

²Quantum Mechanics is a theory which deals with the observation that making things smaller has its limits. This theory does not take atoms as its most basic component, but probability density functions. It is a more complete, but much more complex theory. In the realm of Quantum Mechanics a system of a few molecules is regarded large.

and still continues today [2–8].

To understand what is required to define a force field and to use it to model the behaviour of atoms, we will start with building our first crude model and perform some experiments. The construction of this model requires very little background knowledge and is recommended for all readers to whom molecular dynamics is a fresh concept. After a qualitative description of the model, we will move on to an exact definition which will be implemented in python3. The math and implementation section are meant for high school or university students, attempting to wrap their heads around actually building simulation software. Finally we will discuss the strengths and weaknesses of our model and suggest possible improvements to be made. At the end of this chapter the reader should feel much more comfortable about modelling the physical world and its dynamics, albeit in an armchair. For most readers the first chapter will be the only relevant chapter to read, which is absolutely fine. To others, this chapter might be unnecessary, nevertheless we hope it will result in some fresh views on an otherwise well understood topic.

It is only after careful consideration that concepts tend to become trivial.

The author

1.2. An atomistic toy model

As always one should start simple. Let's assume that atoms are spherical³ and they possess some mass. Maybe for now we could think of them as marbles. All the marbles are the same. Meaning none prefer anyone over the other. However, there is one limitation: No two marbles can be in the same place at the same time. Meaning they interact as hard objects (exclusions), such as billiard balls, chairs, etc. Since all our marbles have the same mass, such a property is meaningless to the marbles in our marble world. Therefore we will exclude mass from consideration⁴.

Figure 1.2 | No two marbles can be at the same place at the same time.

³Spherical in the sense of spherical symmetry. One could think of a circle as a 2D sphere and two dots on a line as a 1D sphere, finally a point is a 0D sphere. 4D and higher dimensional spheres are often called hyper spheres.

⁴A base property which is always constant is immeasurable and meaningless from the observer's frame of reference. E.g. time is pointless if it is always 12.00h, likewise what does distance mean if everything exists at the same place?

1

1.2.1. Start with placing some marbles

We start with putting two of our marbles at any distance on straight smooth rails and try out all possible distances of placement. Due to the local exclusion of marbles within their radius, the center points of any two touching marbles are always separated by two times the marble radius (diameter; Fig. 1.3). All distances between center points further away than the diameter are equally good – if we only consider two marbles. Once placed, we can move the marbles on the rails by repositioning them with our fingers. Continuously moving marbles away from each other causes their relative distance to keep increasing. Where if we move the marbles towards each other, they get closer and closer, until they are touching. It might be interesting to notice that when the available space is larger than the total space occupied by the marbles, we have some choice in designing our *system* (the total of the marbles, space and all other things we include in our model). If the space available matches the total amount of occupied space, there is no freedom in how to build the system. Finally if less space is available than the total occupied space for the marbles, there is no way in which the system can be built at all. Such an initial set-up is called the initial state of position. To finalize our initial state we have to add initial velocities as well. But before we do so we will describe what we mean by movement and velocity.

Figure 1.3 | All possible distances of two marbles on a line. Marbles can be treated as hard spheres which do not allow for distances smaller than two times the marble radius (its diameter). Since there are only two particles (blue, green) and there are no other interactions between our marbles, all distances greater than our diameter are equally likely. A possible distance is represented as a one on the y axis and a zero is used for an impossible distance.

1.2.2. Movement

As we have introduced position, positioning, exclusion and repositioning, it is time to focus on the movement (kinetics) of our marbles (Fig. 1.5 A). Since we are using the macroscopic concept of marbles, we would do well to inspect how marbles move in the real world. As position is only defined relative to other marbles or walls so is movement, for it relies on a change of said distance.

Figure 1.4 | Movement is relative.

Beside the concept of relativity in movement, four important aspects of marble movement should be considered: i) *A marble at rest does not start to move without an external event.* ii) *A marble which is moving stays moving, but slows down due to friction until it is at rest.* iii) *When a marble bounces at a right angle into a wall, its velocity is inverted, while its speed is maintained* (Fig. 1.5 B). Consequently, since velocity is relative, a marble shoots off at twice the speed as the wall that hits it. This under the assumption that the marble was at rest before it collided with the moving wall.⁵ Finally, iv) *like in billiards, if a marble hits another marble which is at rest, the moving marble and the marble at rest exchange velocities upon colliding* (Fig. 1.5 C). Meaning the previously moving marble lies still after the collision and the previously still marble will move at the velocity at which the marble that hit it was traveling. However, since all marbles are equal there is no way in which you can tell them apart, therefore one could also envision the instantaneous moment of collision between marbles as marbles exchanging their position and retaining their velocity. However, in the implementation we will interpret the collision between marbles as the exchange of velocities which is more in line with our macroscopic observation.

1.2.3. Heat pressure space and density

Having defined our concepts of position and movement, we are ready to tackle real world phenomena such as heat, temperature, pressure, space and density. However to do so we have to leave the macroscopic marbles behind and zoom into the atomistic scale. We have to do so as the manner in which friction, heat and temperature manifest is rather different at the size of atoms. At this scale there is no longer any air in between the marbles, for the air itself is made of marbles. Ground friction disappears for the same reason. Any friction that would occur in our system is explicit,

⁵In soccer this implies that kicking a ball could maximally result in the ball shooting off from your foot with the velocity at which you were swinging it. I.e. two times the velocity of the swinging foot from the frame of reference of the ground. Similarly, the minimum velocity at which you have to move your foot backwards to stop the ball is half the velocity of the ball.

Figure 1.5 | Positioning, movement and collisions of marbles. The positioning and movement of a marble in a one dimensional space, the clock in the topleft represents the passing of time in between the two snapshots (A). The collision between a marble and a wall (B). Marble-marble collision (C). On the right the same phenomena are displayed using space-time diagrams. The slopes in the space-time diagrams represent the velocity of the marble. In a space-time diagram the collision between a marble and a non-moving wall should always result in equal angles of incidence and reflection (like a mirror). Similarly the change in the angle of marble movement is identical for marble-marble collisions.

since the only interactions our marbles can have is collision due to exclusion with another marble or a wall. For all physical considerations the rails do not exist and they are just a visual construct to help us think in one dimension of space.

We defined movement as the displacement of marbles relative to other objects (i.e. marbles or walls). These were all properties of individual objects. If we look at the whole system we could spot some other properties such as the total amount of movement and the amount of wall collisions over time. To allow for a more intuitive reading experience we will give these system properties a name which couples well to some familiar macroscopic concepts such as temperature and pressure. Temperature can be thought of as a property which is dependent on the internal motion present in our system of atomistic marbles, therefore it can only be equal to, or greater than zero (average velocity). For example, any temperature higher than zero reflects some form of internal motion of our marbles. Meaning we know for sure that not all marbles will be lying still.

However, since marbles do not start or stop moving on their own, we have to add some initial motion to the system to have any movement at all. In the atomistic marble world, we could apply initial motion with our atomistic fingers by flicking some of the marbles. After doing so we should prevent our edge marbles from shooting off, only to be lost to the infinity of space, by holding our fingers on the rails at some distance around the marbles. To put it another way, we use our fingers to act as walls. In doing so we define a space where the marbles can be. After placing our fingers we observe how often and hard our fingers get hit. The frequency and intensity at which our fingers get hit reflects the internal pressure. In other words, the pressure reflects the effort it takes to hold our finger in place over a certain amount of time. After repeating the proposed experiment a few times we would notice that the harder we initially flick, the harder our fingers get hit by the marbles at the end of the rails. Another observation we could make, is that as we do slide our fingers to a closer distance around the center, they are hit more frequently. As it seems, there is an interesting relationship between movement, pressure and the amount of space our marble system is allowed to take. However, what is this relationship exactly?

Figure 1.6 | At the scale of atoms friction and collision are one and the same.

1.2.4. Taking a closer look

For now consider a piece of rails with only one marble, confined in its length by our fingers acting as walls. If we would flick the marble it would start colliding between our two fingers. After experimenting a bit, we would find that there are two alterations we could make to our initial state to affect the frequency or intensity

1

at which our fingers get hit (i.e. the pressure). One, by initially flicking the marble more or less, we can change the speed at which the marble moves between our fingers. Flicking the marble harder would result in our fingers getting hit more often and intense, and vice versa. Two, by allowing more space on the rails the frequency would go down due to the fact that it takes the marble longer to travel a greater distance at equal speed. Using similar reasoning, decreasing the amount of space would increase the frequency at which our fingers get hit.

A less trivial consequence of sliding our fingers, is that we might change the speed at which the marble is moving. Depending on when we move our fingers we can affect the intensity at which our fingers get hit. If we would move our fingers quickly, in between collisions with the marble, the intensity at which our fingers get hit would remain unchanged due to the sliding. However, if the marble hits our fingers while we are sliding them, we do affect the velocity of the marble. If the marble collides with our finger while we are sliding inwards, we effectively flick the marble at the velocity at which we are sliding, thus increasing the temperature. On the other hand if our fingers get hit whilst we are sliding them further apart, we reduce the velocity of the marble (as taking the ball in soccer stops the ball). A phenomena very similar to the Doppler effect in which the pitch of a sound wave increases if the sound source or reflective wall is moving towards the observer and vice versa. As the temperature of our system is the average speed of marble movement, colliding with walls which are moving away causes the system to cool down.

Finally, if we consider more than one marble we can also affect the pressure. Adding more marbles would cause our fingers to get hit more frequent, as decreasing the amount would have the opposite effect. The fact that decreasing the amount of marbles from one to zero results in no collisions at all, should obviously result in no pressure. However, how is it that increasing the amount of marbles results in more frequent collisions? This has to do with the amount of space available. We already concluded that every marble excludes some space equal to its diameter. Therefore adding an extra marble removes its diameter from the total available space to move through. Meaning less distance needs to be traveled by the marbles to result in a hit with our fingers, causing an increase in the hit frequency. The temperature, however, should go down as we add non-moving marbles to our system. For the total speed of the system remains unchanged but it needs to be divided over more marbles. It should be reasonable that the opposite would be true if we would remove some marbles from our system – if we had more than one to begin with. A nice example of this reduction of effective space by adding more marbles is Newton's Cradle, a common desk toy (Fig. 1.7). By displacing one of the outer metal balls and releasing it, the displaced ball starts to swing towards the chain of resting balls. As it hits the closest resting ball it appears as if the ball at the end of the chain instantaneously starts to move. The movement of the outer balls apparently skips some of the distance due to the presence of the center metal balls.

Armed with our fresh model of position, movement, space, temperature, pressure and the amount of marbles, we would do well to make some predictions to evaluate our atomistic marble model of the world. After which it is time to reflect on the value of our model and decide if we want to put in more effort.

Figure 1.7 | Newton's cradle.

1.2.5. Predictions and validations

As decreasing the amount of available space on the rails containing a certain amount of marbles causes our fingers to get hit more frequently and potentially even more intensely. We might predict that compressing air could result in an increase in pressure and temperature (Fig. 1.8). We have to state could, for the temperature might as well stay the same if one is able to change the shape of the container in between individual collisions. An interesting prediction from our marble world, and one we can easily validate.

The next time we use a bicycle pump we would do well to appreciate the fact that the pressure is going up in our pump because we decrease the amount of available space by compression. The tire increases in pressure because more air (marbles) is filling roughly the same amount of space. Just as our model predicted! Regarding the temperature, if we would touch the tubing where the pump connects to the valve it turns out it always gets hotter during pumping. Pumping in between collisions appears extremely hard and indeed has been proven impossible without generating heat itself[9]. In fact, the heating due to compression is the ignition system in a diesel engine which is considered extremely reliable. An interesting side observation we could make is that decompressing the tire cools the valve, although this phenomena is not explicitly addressed in our model. A lot more could be investigated: continuously beating or heating a tire should cause the inner temperature and pressure to rise to a point at which the tire explodes. Tires in winter should run flat due to the lower temperature causing a drop in pressure, we can boil a milkshake by blending it long enough etc. The consequences are numerous and the effects of them are all around us, all predicted by an imaginary world filled with tiny marbles on rails (and some fingers).

Figure 1.8 | Results from the simulations of constant and dynamic space. The conditions of constant, decreasing and increasing space were tested in silico. Our thought experiments were spot on! Again we see a strong coupling between space, pressure and temperature. The time step clearly shows periodicity in the case of constant space. The velocity of the second wall was zero, negative and positive for constant, compression and decompression respectively.

For now let us stay focussed on our successes and we will get back to some drawbacks at the end of this chapter. Our current atomistic marble model might be considered crude as it only uses simple concepts such as the movement and collision of equal sized marbles between walls. Yet we find that we have been able to simulate real world phenomena from a model at the atomistic scale. Such undertakings fall within the field of molecular dynamics and we have just succeeded in building a very crude but useful atomistic model. However, our marble on a rail model is only expressed in a mostly qualitative manner. We did not construct a complete formal definition of our model. Adding such a description would allow us to make quantitative predictions. Predictions such as the exact increase in temperature of the gas in the bicycle pump due to the act of compressing the air inside. This mathematical implementation requires a high school level understanding of math, although most of it should be understandable to anyone with a little bit of help.

1.2.6. Formalizing the marbles on rails model

To formalize our model we have to express it into a mathematical form. This formalization can then be translated to machine instructions, which would allow us to run our model on a computer. Potentially saving us a lot of time consuming manual calculations. To keep track of what needs to be done it would be wise to make a short list of all the components in our model. Our model contains: space, time, marbles, positions, velocities, and walls. For our marbles it is important to define how they move, collide and interact with the walls. This would include what makes marbles start or stop moving and how to collide with themselves or with the walls. As it turns out all the other properties (e.g. pressure and temperature) discussed previously will be dependent on the elementary components mentioned above.

Position

A marble can be placed at any position in the space enclosed by two walls, as long as its center is at least its diameter away from any other marble's center or its radius from the wall (Fig. 1.5). A position is always indicated with respect to something

else and is therefore relative. The vertical bars around expressions indicate that the result should always be expressed as positive (i.e. -1 and 1 both result in 1).

$$position(m) = meters \quad (1.1)$$

$$radius(m) = |position_{boundary}(m) - position_{center}(m)| \quad (1.2)$$

$$diameter_{marble}(m) = 2 \cdot radius(m) \quad (1.3)$$

$$space_{total}(m) = |position_{wall\ A} - position_{wall\ B}| \quad (1.4)$$

$$space_{effective}(m) = space_{total}(m) - diameter_{marble}(m) \cdot amount_{marbles} \quad (1.5)$$

Movement

Speed is the magnitude of displacement over a time interval (i.e. change in position). Velocity is the speed multiplied with its direction. Left is negative and right is positive one. Speed and velocity rely on position and therefore all of these properties are relative.

$$time(s) = seconds \quad (1.6)$$

$$velocity(m/s) = \frac{position_{final}(m) - position_{initial}(m)}{time_{final}(s) - time_{initial}(s)} = \frac{\Delta position(m)}{\Delta time(s)} \quad (1.7)$$

$$speed(m/s) = |velocity(m/s)| \quad (1.8)$$

Collisions and the amount of marbles

marble-marble In our marble model, we envisioned marble movement to follow the rules of our macroscopic reality (i.e. the world at human scale). From some simple observations we deduced the following truths. When a marble at rest is hit by another marble, they exchange their velocity. Since movement is relative between marbles, any collision can be expressed as a collision between a marble at rest and a marble moving, by subtracting the speed of the first marble from both marbles. Therefore we can solve any collision using the following expressions setting the relative velocity of A to zero. A and B represent the two colliding marbles where the 1 and 2 represent their velocity before and after collision respectively.

$$velocity_{A1\ relative} = velocity_{A1} - velocity_{A1} = 0 \quad (1.9)$$

$$velocity_{B1\ relative} = velocity_{B1} - velocity_{A1} \quad (1.10)$$

1

$$velocity_{A2 \text{ relative}} = velocity_{B1 \text{ relative}} \quad (1.11)$$

$$velocity_{B2 \text{ relative}} = velocity_{A1 \text{ relative}} \quad (1.12)$$

marble-wall A collision of a marble with a non-moving wall results in the inversion of the velocity. Again, movement is relative and we can also solve moving walls with this equation. As the relative speed decreases when the wall is moving away from the marble, the speed of the marble will be lower after the collision. If the wall would move towards the marble, the final speed of the marble is higher. For non-moving walls the final speed is equal to the initial speed. Due to the fact that a marble cannot go through a wall but collides in the opposite directions, marbles can never escape the defined amount of space. Therefore the amount of marbles is constant.

$$velocity_{A2} = -velocity_{A1} \quad (1.13)$$

$$amount_{marbles} = constant \quad (1.14)$$

Temperature, space and pressure

In our description of temperature we stated that temperature is equal or greater than zero and is related to the average movement in our system. To satisfy these rules we could define temperature as the average squared velocity of the marbles in the system. The square is often used in mathematics to make sure that the final value is always positive. As the temperature relates to the average movement of our marbles which we expressed in velocity, we can get rid of the directional character of velocity by squaring it. In this case the squaring conceptually serves the same role as taking the absolute value (speed). However, due to historical reasons the square approach was chosen to use for the definition of temperature.

$$temperature(m^2/s^2) = \sum_{i=1}^{amount_{marbles}} \frac{velocity_i^2(m^2/s^2)}{amount_{marbles}} \quad (1.15)$$

Pressure is defined as the product of the hit frequency and hit intensity of the outer marble with our fingers. The hit frequency is dependent on the effective space of the system and the temperature, while the average hit intensity is only dependent on the temperature.

$$frequency_{impact}(1/s) = \frac{\sqrt{temperature}(m/s)}{space_{effective}(m)} \quad (1.16)$$

$$pressure(m/s^2) = \sqrt{temperature}(m/s) \cdot frequency_{impact}(1/s) \quad (1.17)$$

$$pressure(m/s^2) = \frac{temperature(m^2/s^2)}{space_{total}(m) - diameter_{marble}(m) \cdot amount_{marbles}} \quad (1.18)$$

Conditional consequences

A marble-marble collision can only result in exchanging velocities. A marble-wall collision with a non moving wall results in the inversion of the velocity. Thus, the total and average marble movement must remain constant over time under such conditions. The amount of space is also constant as long as the walls do not move.

$$temperature = constant \quad (1.19)$$

$$space_{total} = constant \quad (1.20)$$

To conclude

Having properly expressed position, movement, velocity, space, temperature, pressure and the amount of marbles, we are finished with the definitions of all discussed concepts.

bonus material However, the huge constraint of equal sized marbles and the separate rule for marble-wall collisions might leave too many open questions for some. Therefore a system including the concept of unequal mass will be quickly defined below for those who cannot resist. This part requires knowledge which is not covered and is meant for the critics reading this chapter. The proper derivation requires concepts such as the preservation of moment and energy, which are not discussed, nevertheless we can briefly evaluate the result. In short, one could consider mass as a weight factor. The magnitude at which a wall's or marble's velocity is altered upon collision is dependent on the relative weight and velocity of the involved objects.

$$velocity_{A2}(m/s) = 2 \cdot \frac{mass_B(kg)}{mass_A(kg) + mass_B(kg)} \cdot (velocity_{B1}(m/s) - velocity_{A1}(m/s)) + velocity_{A1}(m/s) \quad (1.21)$$

$$velocity_{B2}(m/s) = 2 \cdot \frac{mass_A(kg)}{mass_A(kg) + mass_B(kg)} \cdot (velocity_{A1}(m/s) - velocity_{B1}(m/s)) + velocity_{B1}(m/s) \quad (1.22)$$

The apparent advantage of adding mass is that the distinction between a wall and a marble disappears, a wall is just a very heavy marble. From these equations it follows that a marble will practically invert its velocity upon collision with a wall, as long as the wall is standing still and the wall is much heavier than the marble. However, since no wall is infinitely heavy, it will start moving a tiny amount and the marble will lose a tiny amount of movement. Effectively causing the system to start expanding and to cool down. Since the walls will only get hit from the inside, they will keep increasing their velocity until all marbles are moving away from the center point at a velocity lower than the wall, preventing any further collisions with

it. Even though the velocity at which the system expands is fixed from this point, the system will keep expanding forever. Such drift could be prevented by making the walls infinitely heavy. However, allowing the walls to be infinitely heavy would defy Newton's Second law which states that any interaction between two objects has to be symmetrical. Meaning the change on one object due to the other, is equal and opposite to the change on the other object due to the one.

Thus adding mass would bring us no closer to the truth from this perspective. A proper solution for the wall problem is to remove the concept of walls completely and work in a periodic space. However, the concept of pressure becomes a lot less intuitive under such conditions, as well as envisioning the motion. Finally the concept of mass is not mandatory for the concept of motion per se. It is for these reasons that we did not include mass in our model and made the walls inert.

1.2.7. Implementation

Using the proper definitions of our model, we can implement our model into code in three manners. First, we could use an analytic approach in which we always solve all possible collisions and sort them in time. This is followed by moving all marbles up to the first collision event⁶. The simulation is performed by recalculating all collision paths and executing the first occurring collision in an iterative manner. By doing so we could simulate the whole system under all conditions at machine precision accuracy, which is pretty neat. For a one dimensional system with hard interactions such an approach is actually not bad at all⁷. This algorithm was implemented in python3 and was added under an open software license in the supplementary material at the end of this chapter. The graphs and videos were rendered using this simulation software.

The second manner in which we could perform the simulation, is by looking at the systems in small time steps. If our timestep times the highest velocity is always smaller than the radius of the marbles, we can detect collision in the form of overlap. If the displacement of a marble due to its velocity results in overlap with either another marble or a wall, it is treated as a collision taking the time step into account. This algorithm obviously starts to fail as the maximum velocity or time step increases. However, if the interactions are not as hard as we defined them in our model (proper exclusion), but are of a more soft nature the analytical solution is no longer trivial. As it becomes unsolvable for most cases involving more than two interacting objects[10]. Therefore the numeric approach is the approach most often used in condensed matter physics, such as the physics of life.

Finally we could solve any 1D marble system by hand using a ruler, graphing paper and some colored pencils. First, for a system of marbles having a certain position, velocity and radius, we have to shift our marbles from real space to reduced space (Fig. 1.9 C -> A). To do so we make the distance between our marbles with no radius equal to the distance that was in between our marbles when they had a radius measured from their outer boundaries. Secondly, we draw our marble velocity

⁶This would also work for marbles with different masses.

⁷After writing this chapter and the code we discovered a rather trivial method to solve this system with a ruler and some colored pencils, which indicates that this algorithm is actually not very efficient. Nevertheless it works and the third method is presented below.

Figure 1.9 | Constructing complex marble movement by hand. A marble density pattern over time for marbles with no radius (reduced space, A). Annotated marble trajectories with no radius, each line is given a color at the start and this color is propagated through the pattern by never crossing another line (reduced space, B). To give the marbles a radius we can pull apart the annotated trajectories, the amount of displacement represents two times the radius times the order of the marbles (counted from left to right i.e. 1, 2 and 3 respectively) (real space, C). A system from a resting frame of reference can be placed in a moving frame of reference by shearing the diagram in the spatial dimension (D). A system with a changing amount of marbles (E, F). A compressing system (G).

vectors in a space-time diagram and extend them as if they were light rays bouncing between two mirrors (Fig. 1.9 A). Since marbles are presumed to collide and not change places, we can now color all marble traces which do not intersect (Fig. 1.9 B). To get the real marble collision trajectories we have to move from reduced space to real space again by displacing the traces by the same amount we initially used to get from the real space to the reduced space (Fig. 1.9 C)⁸. To change our frame of reference we can apply a shear transformation to our diagram in the spatial dimension

⁸Constructing the collision pattern in reduced space also works for marbles with non-equal radii.

1

(Fig. 1.9 D). After careful inspection of figure 1.9 D we can conclude that a change in reference does not affect pressure and temperature, although it does affect individual marble velocities. This method of marble collision construction can also be used to describe system which have a changing amount of marbles (Fig. 1.9 E, F). It makes sense that this should work, for the past is not dependent on the future. An interesting observation is that that the overall patterns in the space-time diagram do not change upon adding or removing a marble, but the individual marble trajectories do. Finally we can solve for compressing or decompressing systems (Fig. 1.9 G) by reflecting our lines on the wall as if it was a mirror and then add two times the wall velocity to the colliding marble. This one is the hardest to draw on paper for you cannot simply draw reflective lines. The increase or decrease of marble velocity and pressure can be observed by marble traces becoming more or less flat respectively.

1.2.8. Weaknesses of our current model

The one dimensional atomistic model of marbles appears to be adequate for qualitatively⁹ describing concepts such as pressure and temperature, but it seems to have some downsides. To address some of the shortcomings we will take a look at the soft interactions, and the concept of molecules and chemistry.

Soft interactions

In our model we explicitly stated there is only one form of interaction which is governed by the marble/wall exclusion principle: 'No two objects can occupy the same space.' To know how such an event is prevented we investigated two rules of collision. One is the exchange of velocity/position in the event of a marble-marble collision and the other one is the inversion of the marble velocity upon collision with a wall. In such a description of interaction you either interact completely and collide, or you do not interact at all. Another aspect of all these interactions is that they repel. There is no way in which we can have any attractive interaction in our current model. However, gravity appears to be both attractive and 'soft', soft meaning it can act weaker over a distance. For gravity prevents you from falling off the earth, even when you are a considerable distance away from it. It appears we would have to add a whole new set of rules to define such an interaction. The collection of such rules is called the force field and it defines all possible interactions in the model. In our simple model the only interactions in our force field are the simple collisions based on position and velocity, yet to add gravity we have to add another motion law. This law should act on our marbles not only depending on position and velocity, but also their mass. Where mass can be defined as the total amount of marbles grouped together in a lump, moving collectively.

Gravity is not the only interaction which appears to be soft. When two magnets are clicked together and separated, you can feel the pull even if the magnets are partially separated. Another form of soft interaction is the induced magnetic effect due to electricity passing through a conductive coil (an electromagnet). Again we

⁹Our model produces quantitative results, but it only correlates qualitatively with reality. Meaning that we might predict a tire heats up due to compression, which turns out to be true, but we do not accurately predict the exact amount of increase in temperature.

can observe a magnetic effect which appears to act soft over distance, not discrete like our marbles at all.

Molecules and Chemistry

The concept of molecules was briefly discussed when we looked at what it means to be an atom. In our current model we only handled the non-bonded interactions by means of the exclusion rule. However, we stated that atoms can also have bonded interactions, effectively fusing multiple atoms together. To allow for the phenomena of bonded interactions we would have to add a special potential between two marbles to prevent them from moving apart too much. Additionally, in our one dimensional model we would never be able to describe an angle or a dihedral. Such concepts would require at least two and three dimensions of space respectively. Finally, in real life molecules can react with one another and change their chemical behaviour due to those reactions. Obviously such complexity is not covered for our model does not include molecules in the first place.

1.2.9. To conclude

By building a one dimensional dynamic model of marbles we were capable of deriving a powerful real world relationship between temperature, pressure, space and the amount of atoms. The activity of building such a model and using it to simulate the world is what it entails to perform Molecular Dynamics simulations. However, to model real life or its components more accurately we require soft interactions both repulsive and attractive, as well as molecules with angles and dihedrals. Therefore a one dimensional model without attraction will never be enough to simulate life (completely).

There are many options available to improve our model in such a manner that it indeed can handle concepts such as attraction, gravity and electromagnetism. However, it is mainly the lack of attraction which is radiating in absence. Without attraction we can never model anything which has any viscosity and almost everything in our real world seems to be at least slightly viscous (like honey and water to a lesser extent). Therefore we think the best way forward with our model is to include attraction. This can even be done separately from 'softness' for we could add another rule which states that whenever a marble's velocity is lower than the combined attraction between the colliding marbles they remain stuck together and move collectively. However to solve for collision with a collective lump we have to model how collisions work between objects with a different mass. Softness could then be added as the cherry on the 1D marble pie.

We also did not allow marbles to be different from each other, while in nature there are many atoms and the manner in which they interact and form molecules can widely vary. Therefore, if we were to be serious about modelling life from an atomistic point of view we would require a model with at least three dimensions of space and the capability of having a wide variety of marbles and interactions. In the following chapter such a complex model (Martini) will be introduced. In essence Martini is very much like our one dimensional model but the devil is in the detail.

Finally we hope that the iterative nature of science became clear to all, we started

1

off with a guess (1D marbles), formalized our idea and compared it to reality. What are the fruits of our labor? On the one hand we always find that there are things missing and we are not done, yet in our attempts to model nature by marbles we learned a great deal about motion, temperature, pressure and space! To all (high school) students, friends, parents and other families we would like to thank you for your interest in this topic and we hope dynamic models are a bit less of a mystery from now on. If you have any questions or remarks, please feel free to contact the author of this thesis.

1.3. References

- [1] T. Slootweg, *Hegel's Philosophy of the Historical Religions*, Critical Studies in German Idealism (Brill, 2012).
- [2] J. Huang, S. Rauscher, G. Nawrocki, T. Ran, M. Feig, B. L. de Groot, H. Grubmüller, and A. D. MacKerell, *CHARMM36m: an improved force field for folded and intrinsically disordered proteins*, *Nature Methods* **14**, 71 (2016).
- [3] M. M. Reif, P. H. Hünenberger, and C. Oostenbrink, *New interaction parameters for charged amino acid side chains in the GROMOS force field*, *Journal of Chemical Theory and Computation* **8**, 3705 (2012).
- [4] K. Roos, C. Wu, W. Damm, M. Reboul, J. M. Stevenson, C. Lu, M. K. Dahlgren, S. Mondal, W. Chen, L. Wang, R. Abel, R. A. Friesner, and E. D. Harder, *OPLS3e: Extending force field coverage for drug-like small molecules*, *Journal of Chemical Theory and Computation* **15**, 1863 (2019).
- [5] J. A. Maier, C. Martinez, K. Kasavajhala, L. Wickstrom, K. E. Hauser, and C. Simmerling, *ff14sb: Improving the accuracy of protein side chain and backbone parameters from ff99sb*, *Journal of Chemical Theory and Computation* **11**, 3696 (2015).
- [6] K. N. Kirschner, A. B. Yongye, S. M. Tschampel, J. González-Outeiriño, C. R. Daniels, B. L. Foley, and R. J. Woods, *GLYCAM06: A generalizable biomolecular force field. carbohydrates*, *Journal of Computational Chemistry* **29**, 622 (2007).
- [7] A. Krämer, F. C. Pickard, J. Huang, R. M. Venable, A. C. Simmonett, D. Reith, K. N. Kirschner, R. W. Pastor, and B. R. Brooks, *Interactions of water and alkanes: Modifying additive force fields to account for polarization effects*, *Journal of Chemical Theory and Computation* **15**, 3854 (2019).
- [8] S. Riniker, *Fixed-charge atomistic force fields for molecular dynamics simulations in the condensed phase: An overview*, *Journal of Chemical Information and Modeling* **58**, 565 (2018).
- [9] L. Szilard, *On the decrease of entropy in a thermodynamic system by the intervention of intelligent beings*. *Behavioral science* **9** **4**, 301 (1964).
- [10] J. Barrow-Green, *Henripoincaré, memoir on the three-body problem (1890)*, in *Landmark Writings in Western Mathematics 1640-1940* (Elsevier, 2005) pp. 627–638.

2

A Practical View of the Martini Force Field

Simulating is believing.

Siewert-Jan Marrink

Martini is a coarse-grained (CG) force field suitable for molecular dynamics (MD) simulations of (bio)molecular systems. It is based on the mapping of two to four heavy atoms to one CG particle. The effective interactions between the CG particles are parametrized to reproduce partitioning free energies of small chemical compounds between polar and apolar phases. In this chapter, a summary of the key elements of this CG force field is presented, followed by an example of practical application: a lipoplex-membrane fusion experiment. Formulated as hands-on practice, this chapter contains guidelines to build CG models of important biological systems, such as asymmetric bilayers and double-stranded DNA. Finally, a series of notes containing useful information, limitations, and tips are described in the last section.

Keywords: Coarse-grained models, Martini force field, Molecular dynamics simulations, Biomolecular systems

This chapter has been published in Biomolecular Simulations: Methods and Protocols, Methods in Molecular Biology (2020)[1].

2.1. Introduction

The initial Martini coarse-grained (CG) force field was developed in 2003 to study lipid membrane properties[2–4]. It allowed to investigate the behavior of large lipid aggregates at spatial and time scales unachievable to atomistic MD simulations, while retaining enough resolution and chemical specificity to give a microscopic and dynamic picture still unavailable in experiments. The Martini force field was shown to be capable to address a wide range of lipid-based processes such as vesicle self-assembly, vesicle fusion, lamellar to inverted hexagonal phase transition, and the formation of the gel- and liquid-order phases[2–7]. Over the years, the applicability of the force field has expanded to most common biomolecules such as proteins[8, 9], sugars[10, 11], nucleotides[12, 13], and some important cofactors[14], as well as many nonbiological molecules including synthetic polymers[15–21] and nanoparticles[18–20, 22]. Examples of Martini CG models are shown in Fig. 2.1 A. A complete list can be found under “downloads” at cgmartini.nl. Noteworthy is the high compatibility of the individual models with each other. This allows for the modeling of complex biological environments such as the plasma membrane[23] (shown in Fig. 2.1 b) and photosystem II in a thylakoid membrane[24].

This high compatibility is achieved by a clear modular mapping and parameterization scheme based on building blocks, called beads. Martini is a CG force field, which, in general, maps four non hydrogen atoms to a single CG bead. During the mapping, chemical groups such as carboxylates or esters are represented by a single CG bead. This approach makes it easy to build new models based on the already available ones. The CG beads come in four chemical classes (or “flavors”): charged (Q), polar (P), nonpolar(N), and apolar (C). The Q and N classes each have four subtypes that are linked to their capability of participating in hydrogen bonding: donor and acceptor (da), donor (d), acceptor (a), and none (0). The main difference between these subtypes is their interaction strength with each other, allowing for a qualitative representation of hydrogen bonding. The P and C beads each have five subtypes, which represent a gradient from weak to strong polar or apolar properties, respectively. In total, this gives rise to 18 different bead chemical types. For computational efficiency, the mass of all standard beads is set to 72 amu, which equals the mass of four water molecules (represented by a P4 bead type in Martini).

Martini employs bonded and non-bonded potential forms commonly used in atomistic force fields, which make the model easy to be implemented in modern molecular dynamic programs as GROMACS[4, 6], GROMOS[26], and NAMD[27, 28]. Although this choice of potential forms is not the most accurate one for coarse-grained models (see Notes 2.4.1 and 2.4.2)[25], it enables Martini to take benefit of all the advances in high-performance parallel algorithms and enhanced sampling techniques developed in the past years. For the non-bonded interactions, 12–6 Lennard-Jones and Coulomb potentials are used (as shown in Eq. 2.1). In practice, these potentials are shifted and truncated for computational speedup. In the current implementation[6, 29], these potentials are both shifted such that the potentials reach 0 kJ/mol for any distance greater than 1.1 nm, the cutoff distance. In case of the LJ potential, ten levels of interaction are defined, differing in the LJ well depth (epsilon ranging from 5.6 to 2.0 kJ/mol), but with the same bead size (a sigma of

0.47 nm is used for the standard beads, except for interaction level IX, which has an increased sigma of 0.62 nm). For all possible pairs of CG bead types, one of those ten interaction levels has been assigned. These levels have been chosen based on the experimental water-oil partitioning of small molecules that are represented by each of the beads. Only the Q-beads bear an explicit charge and additionally interact via the Coulomb potential with a relative dielectric constant $\epsilon_{rel} = 15$ for explicit screening. Together with the use of a shift function, this effectively results in a distance dependent screening. To allow for the mapping of aliphatic and aromatic ring structures (such as cyclohexane or benzene), a smaller bead (denoted with prefix "S") was introduced, mapping two or three nonhydrogen atoms to a single CG bead. The S beads have a reduced sigma of 0.43 nm and a scaled interaction strength corresponding to 75 of their standard bead counterpart. In the current model only S-S interactions make use of the reduced interaction scheme, that is, interactions with

normal (N) particles are treated as N-N interactions. For the full parameterization scheme and interaction table, we would like to refer to the original paper of Marrink et al.[6] After the release of Martini 2.0 in 2007, an extra bead size has been added to the collection. The tiny (T) bead was introduced for the mapping of nucleotides and follows three/two-to-one mapping. Such a tiny bead is needed for correct stacking distance of the nucleobases in the double-stranded helix. The interaction strength of the T beads is not reduced, but their sigma is 0.32 nm. T beads interact with S and N beads as S-S and N-N, respectively (see Note 2.4.3)[12]. Besides the extra bead size, many new beads were introduced to satisfy the needs of specific models, as nucleic acids[12, 13], polymers [15–17, 21], nanoparticles[22], and some sugars[11, 30]. Polarizable water models [31, 32] have been designed for modeling of systems where implicit screening of electrostatic interactions caused by reorienting water dipoles is necessary (as discussed in Note 2.4.4).

$$V^{non-bonded} = \underbrace{\sum_{ij} [(\frac{\sigma_{ij}}{r_{ij}})^{12} - (\frac{\sigma_{ij}}{r_{ij}})^6]}_{V^{LJ}} + \underbrace{\sum_{ij} \frac{q_i q_j}{4\pi\epsilon_0\epsilon_{rel}r_{ij}}}_{V^{Coul}} \quad (2.1)$$

Where the non-bonded interactions follow a “top-down” modeling approach (making use of experimental partitioning data), the bonded interactions are usually extracted via a “bottom-up” approach, based on reference atomistic data. This is achieved by mapping the heavy atoms of an atomistic simulation to a pseudo CG trajectory. The CG beads are usually placed at the centers of mass of the atoms they represent. From these pseudo CG trajectories, the bonded parameters can be extracted and compared to the CG model under development. By changing the bonded parameters, one should try to maximize the overlap of the conformational distributions of the pseudo CG and real CG models. This is an iterative process that should be repeated for achieving the best results. However, the philosophy of the Martini force field is to use(mainly) simple bonded potentials (as shown in Eq. 2.2); therefore, perfect overlap is not always achieved. Also, be aware that the bonded parameters might influence the partitioning of your molecule or even other macroscopic properties (e.g., area per lipid for bilayers and radius of gyration for polymers). Therefore, validation of your CG model against experimental data, after satisfactory bonded parameters have been achieved, is considered good practice. An in-depth tutorial of parameterizing a new molecule can be found under “Tutorials” at the Martini web page (cgmartini.nl), which is summarized in Fig. 2.1 c. Automatic parameterization approaches could be an easier but probably a less accurate option to generate Martini models[33]. However, they could provide prospects for high-throughput simulation methodologies[34]. Note that, to keep the secondary structure of proteins and nucleotides close to the target state (e.g., crystal structure), Martini makes use of additional harmonic bonds that define an elastic network (see Note 2.4.5).

$$V^{bonded} = \underbrace{\sum_{bonds} K_r(r - r_0)^2}_{V^{bond}} + \underbrace{\sum_{angles} K_\theta(\cos(\theta) - \cos(\theta_0))^2}_{V^{angle}} + \underbrace{\sum_{dihedrals} K_\phi(1 + \cos(n\phi - \delta))}_{V^{dihedral}} \quad (2.2)$$

In the next subsection of this chapter, we will describe a hands-on tutorial, which uses one of our current projects (lipoplex-membrane fusion) as a guide. This practice addresses useful examples of how to build Martini CG models of macromolecules (double-stranded DNA), (solvated) liquid crystals, and complex asymmetric membranes. At the final part, all these CG models are put together for particle-membrane fusion simulations.

2.2. Hands-On: cationic lipid-DNA lipoplexes for gene transfer

This section will be a guide for setting up a CG simulation of a lipoplex-membrane fusion experiment making use of the Martini2.0 force field with the DNA extension[12]. We will start with a small introduction regarding lipoplexes and their biological relevance. However, the main objective of this section is to show the construction of such a complex system using the Martini approach, from A to Z.

Lipoplexes are complexes of genetic material and lipids used for transfection in gene therapy. Due to the high negative charge of nucleotide polymers (DNA, RNA), they do not readily cross the hydrophobic core of biological membranes via a passive mechanism. Another downside of using naked DNA for therapeutic purposes is its low half-life time in the bloodstream[35]. In lipoplexes, the packing architecture is such that, upon fusion with a membrane or vesicle, its content is mainly released on the side of the membrane opposing the leaflet of initial fusion[36]. This allows the escape of the genetic material from the endosome in which the lipoplex is trapped after being taken up by the cell via endocytosis. Transfection of cells utilizing depathogenized viral vectors currently has a much higher transfection efficacy than the lipid-based vectors or any other non-viral-based method[37]. However, depathogenized viral vectors still trigger the immune system in humans, causing their application in medical gene therapy to be limited[38]. Another drawback of viral vectors compared to their non-viral counterparts is their high preparation cost[38]. Therefore, increasing the efficacy of non-viral vectors for transfection is required for the development of gene therapy as a safe and affordable medical treatment.

In this hands-on section, we will build a lipoplex using one of the earlier lipoplex compositions used[39], in which the DNA is trapped in an inverted hexagonal (H_{II}) lipid phase. First, the lipoplex itself will be built using 1,2-dioleoyl-sn-glycerol-phosphoethanolamine (DOPE) as the helper lipid and 1,2-dioleoyl-3-trimethylammonio-propane (DOTAP) as the cationic lipid in a 4:1 ratio. In this example, we will use double-stranded (ds) DNA oligomers, with a length of 24 base pairs (bp), as the gene carrier. Replacing the DNA with RNA should not change the general procedure described in this section, though this was not specifically tested. Lastly, the constructed lipoplex will be solvated and fused with an asymmetric bilayer mimicking the endosomal membrane.

During this section, lines preceded with a "\$" are to be executed in the terminal. This hands-on was designed to be used in combination with GROMACS 2016.3[40]. Visual Molecular Dynamics (VMD) version 1.9.3[41] was used for visual inspection, and Python 2.7 was the default python compiler used for all python scripts, unless

stated otherwise.

2

2.2.1. Building a liquid lipoplex crystal

For building the inverted hexagonal phase (H_{II}), we will make use of the procedures described in [42, 43]. We start with preparing a Martini CG model for DNA, which is to be rested on top of a bilayer. Then, we duplicate this DNA-membrane stack and perform energy minimization, equilibration, and finally a production run to form the crystal lattice.

Preparing the DNA

First, download the .pdb file from the martini website. This structure corresponds to a sequence of DNA with 24 bases in each strand ([CGCGAATTCGCG]2).

```
$ wget http://cgmartini.nl/images/parameters/dna/24bp_AA2CG_stiff
  .tar.gz
$ tar -xvf 24bp_AA2CG_stiff.tar.gz
```

To transform an atomistic DNA structure into a CG structure, we will use `martinize-dna` (cgmartini.nl, under Downloads, Force field parameters, DNA)[12, 13]. Before running `martinize-dna`, we should remove the ion and water molecules from the .pdb. Be aware that if you have used a DNA structure builder, to obtain a different length or sequence of DNA (e.g., using `scfbio-iitd.res.in`), the final atom and residue names have to follow the default pdb nomenclature.

The "24bp_cleaned.gro" file is already made compatible with the `martinize-dna` script and we can perform the all-atom (AA) to CG transformation (Fig. 2.2 a). The output includes a "CG.pdb" and the "CG.itp" files. We will use a stiff elastic network to imply a strict helical structure to our DNA (see Note 2.4.5). Later on, this could be replaced by a softer elastic network if desired.

```
$ python martinize-dna.py -dnatype ds-stiff -f 24bp_cleaned.gro -
  x 24bp_CG.pdb
```

Always read the output and check, for example, if the number of chains or base pairs specified matches that what you expect. In our case, this should be two chains (A and B), each containing 24 nucleotides. We need to rotate the DNA such that it will lie parallel to the membrane. We will use GROMACS to do so.

```
$ gmx editconf -f 24bp_CG.pdb -rotate 0 90 0 -o 24bp_CG_rotated.
  gro
```

Creating the DNA bilayer stack

Now, we need to generate a simulation box containing our DNA CG model and a symmetric bilayer with the desired lipid composition. We will aim at a 4:1 ratio of DOPE and DOTAP, respectively. To do so, we will use `insane`[44]. `Insane` is a python program, developed in-house, which generates an initial CG configuration using a grid-based approach. This procedure makes `insane` one of the fastest initial state builders for membranes with or without incorporated protein(s). The latest stable version of `insane` can be downloaded from our web page (cgmartini.nl, under

Figure 2.2 | Building a lipoplex with Martini. (a) The CG mapping of a 24 bp dsDNA; (b1, b2) shows two views of the initial bilayer with a double-stranded DNA; (c) This system is replicated to form a lamellar configuration; (d) After running 45 ns of MD simulations, the lamellar phase is converted to an H_{II} phase (liquid lipoplex crystal); (e) The liquid lipoplex crystal is solvated and coated with lipids; (f) An example of a lipoplex-membrane fusion experiment.

downloads/tools). We will generate a small piece of membrane with DNA, which will be used later on to generate the H_{II} phase with dsDNA inside its channels.

```
$ python insane.py -l DOPE:4 -l DOTAP:1 -aname DOTAP -ahead "C"
  -allink "G G" -altail "CDCC CDCC" -x 11 -y 6.5 -z 7 -f 24
  bp_CG_rotated.gro -dm 3.5 -o bilayer_1DNA.gro -sol W -salt
  0.150 -sold 0.87
```

As DOTAP is not a default lipid, its topology needs to be completely described in insane using the flags "-aname" (name of the new lipid), "-ahead" (groups in the lipid head, where "C" defines the head as choline), "-allink" (groups in the lipid linkers, with "G G" defining two ester groups), and "-altail" (define the lipid tails, where each "C" indicates four carbon atoms in a linear saturated chain, while "D" indicates four linear carbon atoms containing a single unsaturation). This should generate a box containing roughly 187 DOPE, 44 DOTAP, plus the DNA at physiological salt concentration (as shown in Fig. 2.2 B1, B2). The target ratio of atomistic water with respect to the amount of lipids is around 8:1[42, 43]. For Martini water, with one bead representing four water molecules, the target ratio is thus 2:1. Do remember that the charge of the DNA is not yet neutralized and this will be -46. Therefore, overshooting your target amount of water by 46 is recommended, for we will transform those waters into sodium beads. To generate a bigger complex, we will copy the DNA-membrane configuration along its axis perpendicular to the channel normal (as shown in Fig. 2.2 c).

```
$ gmx genconf -f bilayer_1DNA.gro -o 2bilayer_4DNA.gro -nbox 122
```

Preparing the topology files

Next, we need to generate a top file which matches the "2bilayer_4DNA.gro" composition and order and makes use of the correct topology files. To achieve a charge neutral system, we will reduce the number of waters in the topology by 46 and add 46 sodium ions right underneath (this will transform 46 water molecules into sodium ions). The final topology should resemble the example topology below

```
#include "martini-dna-150909/martini_v2.1-dna.itp"
#include "martini-dna-150909/martini_v2.0_ions.itp"
#include "martini_v2.0_DOTAP.itp"
#include "martini_v2.0_DOPE_02.itp"
#define RUBBER_BANDS
#include "Nucleic_A+Nucleic_B.itp"

[ system ]
; name
Martini system containing 4 dsDNA and 2 bilayers
[ molecules ]
; name      number
Nucleic_A+Nucleic_B 1
DOPE          89
DOTAP         22
DOPE          89
```

```

DOTAP                22
W                    106
NA                    49
CL                    47
...another 3 times for the 2bilayer4_DNA.gro

```

Running EM/EQ and production

Now, we have all the components together, and we can start running an energy minimization, equilibration, and production run. We will not go into much details, but the default settings can be found at cgmartini.nl (under Downloads, Force field parameters, Input parameters). To perform energy minimization, we need to set the integrator to "steep" and 1000 steps should suffice.

```

$ gmx grompp -f em.mdp -c 2bilayer_4DNA.gro -p topol.top -o em.
  tpr -maxwarn 1
$ gmx mdrun -deffnm em -v

```

The "-maxwarn" flag is to allow overwriting name mismatches between the .gro and .top file, caused by the DOTAP and added sodium ions. After running the minimization, we need to create an index file ("index.ndx") to use for our temperature coupling scheme. We will create a group for all lipids, the DNA, and the water plus ions. For the sake of this tutorial, those groups are called "Lipids", "DNA" and "W_IONS" respectively. The index groups can be easily generated from the "em.gro", using "gmx make_ndx".

```

$ gmx make_ndx -f em.gro -o index.ndx

```

Before we run the equilibration, we need to fix the DNA in the x dimension. This will help with a smooth conversion from the periodic crystal to the solvated naked lipoplex. To do so, we need to add a few lines of code to the bottom of our "Nucleic_A+Nucleic_B.itp".

```

#ifdef CONSTRAINED_X
[ position_restraints ]
;ai funct fcx fcy fcz
1 1 500 0 0; restrains to a plane (y-z-plane)
305 1 500 0 0; restrains to a plane (y-z-plane)
#endif

```

For equilibration, 250,000 steps at a 2 fs time step should suffice. We will use anisotropic pressure coupling, and the berendsen barostat for improved stability. Do not forget to set the integrator back to "md" and add "define -DCONSTRAINED_X". In the "mdrun" command we add an "rdd" of 2 to prevent instabilities due to our long elastic bonds in the DNA.

```

$ gmx grompp -f eq.mdp -c em.gro -p topol.top -n index.ndx -o eq.
  tpr
$ gmx mdrun -deffnm eq -v -rdd 2.0

```

For the production run, we will use roughly the same settings as those for equilibration. Important is that the pressure coupling will be changed from "berendsen"

to "parrinello-rahman", and the time step for the integrator should be larger. For systems containing DNA, a time step of 10 fs is the maximum (see Note 2.4.6).

```
$ gmx grompp -f md.mdp -c eq.gro -p topol.top -n index.ndx -o md.tpr
$ gmx mdrun -deffnm md -v -rdd 2.0 -nsteps -1
```

Keep running the simulation until inspection with VMD shows full formation of the H_{II} phase.

While writing this hands-on section, it took roughly 2,268,000 steps (45 ns CG time, which should correspond to 200 ns realtime, as explained in Note 2.4.7) to complete the formation of the H_{II} phase (as shown in Fig. 2.2 d). The final frame can be extracted using VMD, and for the sake of this tutorial, we assume it was named "md.gro". We are now finished with the construction of the liquid lipoplex crystal. Do keep in mind that this crystal was made to be solvated, and therefore, its PBC conditions were not optimized for a perfect hexagonal unit cell.

2.2.2. Solvating a liquid lipoplex crystal

For the second part of this tutorial, we will use the liquid lipoplex crystal we generated in part 2.1. The goal is to extract the crystal in such a manner that the channel geometry is not disturbed (naked lipoplex). We will also have to add an extra layer of lipids around our extracted lipoplex to act as a coat (coated lipoplex, as shown in Fig. 2.2 e). Once we have added the coating lipids, we can solvate the whole system. From there on, we energy minimize, equilibrate, and run a production run in a similar manner as that described in part 2.1.

Extracting the periodic crystal

The liquid lipoplex crystal generated before contains four channels with four strands of dsDNA in total. This could be considered the smallest possible crystal under cubic periodic boundary conditions. To generate a larger crystal we use "gmx genconf" to copy this box in the desired dimensions. Even though we will stick to the 2x2 geometry for the solvated lipoplex, we will duplicate the box in its y and z dimensions. By doing so, we can make extraction of a 2x2 lipoplex much easier. This might not seem logical as of yet, but bear with us, and you will see that this is indeed the case. To prevent weird indexes after copying the box, we will first make the .gro file "whole".

```
$ gmx trjconv -f 2bilayer_4DNA.gro -pbc whole -o 2bilayer_4DNA_whole.gro
$ gmx genconf -f 2bilayer_4DNA_whole.gro -o 4bilayer_16DNA.gro -nbox 1 2 2
```

We will use VMD to select the inner four strands of DNA. Then, we will make an area selection around these strands of DNA to include all their lipids, water, and ions.

```
$ vmd 4bilayer_16DNA.gro "same resid as within 20 of index your_central_4_DNA_strands"
```

Export the selected structure and name it "naked_lipoplex.gro".

Coating the naked lipoplex

After extracting the naked lipoplex, we will have to add coating lipids. This can be achieved in many ways. One of the options is to increase the selection range in the VMD command used above in combination with excluding any DNA beads. However, the lipids you will have selected in doing so are in a structure that is rather stable. This would make equilibration of the coat of the lipoplex, potentially, a long and expensive process. Therefore, we will make use of PACKMOL, another tool for initial state building[45]. PACKMOL differs from insane, for it uses a packing optimization in defined regions of space. This makes PACKMOL much slower than insane for building simple membranes, but it allows for the addition of molecules to an already complex geometry. PACKMOL can be downloaded free of charge at the web page of the University of Campinas (www.ime.unicamp.br/martinez/packmol/).

We need to coat the lipoplex with a nice monolayer, and we will make a rough initial approximation by regarding each face of the naked lipoplex as an independent plane. Then we will use the average area per lipid (APL) to calculate the amount of lipids we would need to cover the total area of our lipoplex. To calculate the APL of our mixture, we have to set up a symmetrical bilayer containing our lipid concentrations. A bilayer of 10x10 nm² will be large enough to get an accurate APL for the lipids used in this tutorial. This bilayer can be constructed following the same protocol as that described in part 2.1. To calculate the APL, we need to simulate the solvated bilayer (under semi-isotropic pressure coupling) up to the point that the box dimensions are stable for a while. Then we can extract the box dimensions over time using "gmx energy".

```
$ gmx energy -f md.edr
```

Select either the x or y dimension as the preferred output and calculate the average value over the period where it is stable (e.g., the last 10%).

$$APL_{DOPE,DOTAP} = \frac{2x^2}{\text{Number of lipids}} \approx 0.65 \text{ nm}^2 \quad (2.3)$$

To roughly calculate the area the outer monolayer (the coat) has to cover, you can use VMD. Pressing "2" in the visualization window will allow you to select two particles and measure the distance between them. We will add 2 nm to each side to accommodate for the fact that the outer leaflet has an increased distance to span with respect to the inner leaflet (which we are measuring).

$$\begin{aligned} Lipids_{added} &= \frac{2[(x+2)(y+2)] + [(x+2)(z+2)] + [(y+2)(z+2)]}{APL_{DOPE,DOTAP}} \\ &\approx 1800 \rightarrow 1440 \text{ DOPE}, 360 \text{ DOTAP} \quad (2.4) \end{aligned}$$

After estimating the amount of lipids we need, we will use PACKMOL to configure them around our lipoplex. Extensive tutorials for using PACKMOL can be found at their web page (m3g.iqm.unicamp.br/packmol/). As PACKMOL makes use of .pdb files, we will have to convert our individual files to .pdb.


```
$ gmx editconf -f your_file.gro -o your_file.pdb
```

Use VMD to obtain a .pdb file for each of the lipids using the export coordinates function, with a single lipid selected. After generating the correct PACKMOL input file we can run it.

```
$ packmol < coating_lipoplex.inp
```

Running PACKMOL can take up quite some time and sometimes no correct packing can be found, even upon running for multiple hours. Try playing around with the excluded and included volume which are defined in the input file (outside and inside box), until a satisfactory packing has been achieved. The output structure is presumed to be named "coated_lipoplex.pdb". Convert the .pdb to a .gro in the same manner as was demonstrated in part 1.

Tidying up the GRO and topology files

To tidy things up, you need to sort the "coated_lipoplex.gro" and make sure that the topology has the same order and number of molecules as your sorted .gro. After sorting and updating our topology, we will solvate the lipoplex using insane.

```
$ python insane.py -f coated_lipoplex.pdb -o solvated_lipoplex.gro -sol W -salt 0.15 -pbc cubic
```

Add the amount of waters and ions insane added to your sorted topology. During the PACKMOL step, we also added more DOTAP, which causes our system to have a nonzero net charge. Therefore, we will replace some of the added water in the insane step with CL to obtain a net charge of zero. To prevent freezing of water (see Note 2.4.1), we will also replace 10% of the waters in our last entry with antifreeze water (WF).

Running EM/EQ and production

To finish, we need to energy minimize and equilibrate our system before we can start the production run. To do so, we can use the same procedure as before, but we will now use the "-DCONSTRAINED_XYZ" flag (constructed in the same manner as before, but now in XYZ) until large deformations are resolved, and we will use isotropic pressure coupling. It will take roughly 10 μ s for the solvated lipoplex to equilibrate its outer coat. You can use "gmx gyrate" to inspect if the lipoplex' shape has stabilized. If the outer coat is too loose, or tight, try adding more or less coating lipids with PACKMOL, or change the included and excluded volume. We are now done with solvating a liquid lipoplex crystal.

2.2.3. Lipoplex fusion with an asymmetric complex membrane

For the lipoplex fusion with an asymmetric complex bilayer, we will use the solvated and equilibrated lipoplex generated in part 2.2 of this hands-on section, in combination with a generated asymmetric bilayer.

Creating an asymmetric bilayer

By now, we are able to generate complex bilayers; however, we never attempted to generate an asymmetric one. Here, we want to model an endosomal type membrane containing a 1:1 DPPC/DOPC mixture in the upper leaflet and a 2:2:1 DPPC/DOPC/DOPS mixture in the bottom one. To do so, we will use *insane* again. However, first we generate each leaflet of our asymmetric bilayer as a symmetrical bilayer. Thus, for each asymmetric bilayer, you need to run two symmetric simulations. From those simulations, we can obtain the complex APL for each of the leaflets. We can use the same protocol as described in part 2.1. After obtaining the complex APL for each of the leaflets, we will use *insane* to generate the complex bilayer using the obtained APL of each leaflet. This will make sure that the final tension in the two leaflets is equal. An example is given below:

```
$ python insane.py -u DPPC:1 -u DOPC:1 -ua APL_{DPPC-DOPC} -l
  DPPC:2 -l DOPC:2 -l DOPS:1 -a APL_{DPPC-DOPC-DOPS} -x 10 -y
  10 -z 10 -sol W -salt 0.150 -o complex_asymmetrical_bilayer.
  gro
```

The topology files for a wide range of lipids can be found at our web page (cg-martini.nl, under Downloads). Make the corresponding topology using the output of *insane* and equilibrate the membrane. This will take roughly 500 ns, though the amount of time equilibration will take increases with the complexity of the membrane. Equilibration of the membrane can be done as before in parts 2.1 and 2.2; do make sure that you use semi-isotropic pressure coupling and that your membrane lies in the XY plane.

Combining the asymmetric bilayer and the lipoplex

After obtaining an equilibrated membrane, we will remove the waters and ions. We now have an equilibrated membrane in vacuum and a solvated lipoplex. To be able to combine the lipoplex and the membrane, we will also extract the solvated lipoplex from its environment. We will use VMD for this extraction.

```
$ vmd solvated_lipoplex_equilibrated.gro
```

To extract the lipoplex from its environment, we will use the following selection and export it, as described in part 2.1. ``pbwithin 15 of (resname DOPE DOTAP or name "SC.*" "BB.*")``. We will save it as `"extracted_lipoplex.gro"`.

To combine our extracted lipoplex with the equilibrated asymmetric complex membrane, we create a combined .gro file. To do so, we copy the content of the membrane into the new .gro file (`"lipoplex-membrane.gro"`). Remove the final line (the box description). Add the content of the `"extracted_lipoplex.gro"` but remove the header line, atom count, and box description.

If the lipoplex and membrane overlap, you can use `"gmx editconf -translate x y z"` to shift your lipoplex. Solvate the system as before using *insane*. Generate a topology file that matches the .gro one. Energy minimize and equilibrate (as before in part 1) using "semi-isotropic" pressure coupling. The equilibrated system should resemble Fig. 2.2 F. To perform an unbiased fusion experiment, you have to simulate for a long time to observe adhesion, stalk formation, and transfection. However, such a

simulation would take up a large amount of computational time; therefore, you could add biasing forces to drive membrane adhesion and initiate stalk formation. From there on, you could remove all biasing forces to observe the fusion process after initial stalk formation [2].

Alternative methods and tools

We suggest to the reader as possible follow-up steps for this hands-on section, the building of even more complex CG systems. For example, a lipoplex fusion experiment within a vesicle (a process that could mimic the DNA transfection event in the early endosome). However, the current implementation of insane does not allow us to build such lipid structures yet. A user-friendly option for this problem could be the usage of a graphical user interface as CHARMM-GUI Martini Maker[46, 47]. In the current implementation, this program can build Martini models of micelles, nanodiscs, bilayers, and vesicles. We hope that the tools and tips presented above will help you on your way with your own implementations of the Martini force field. If you would like to learn more about Martini, you can visit our web page for other tutorials (cgmartini.nl). For any questions regarding the implementation of the Martini force field for your project(s), we would like to direct you to the forum at our web page.

2.3. Outlook

Since its initial publications, the Martini force field has been developed and tested in a broad range of applications, from simple lipid bilayers to complex fusion processes as detailed here for the lipoplex hands-on. Despite the huge success of the model, certain problems have been reported as excessive protein and sugar aggregation (see Note 2.4.4), and water freezing (also described in Note 2.4.1). Along with the modeling demand of new and challenging systems, these limitations pushed the MD group of Groningen to improve the CG beads — the fundamental building blocks of Martini — until now largely untouched since version 2. One of the main features of the forthcoming version, entitled Martini 3, is the reparameterization of small (S) and tiny (T) beads, designed to be fully balanced with the normal (N) size beads. New chemical-type beads were also tuned to model systems not included in the current version. For instance, we will have more beads with hydrogen-bonding capabilities (including all polar and nonpolar beads) and charged beads dedicated for modeling divalent ions. Water has also its own special bead, parameterized to improve its miscibility with other beads and also avoid freezing problems. The interaction matrix was modified, including more interaction levels and smoother transitions between the beads. In this aspect, special attention was taken regarding charged beads, with trends in solvent polarization and ion- π interactions implicitly included in the Lennard-Jones potential with neutral beads. Together with other new features of the model, preliminary tests indicate a significant improvement of proteins, sugars, and nucleic acids in Martini 3. Besides, exciting new systems seem to nicely behave, including MD simulations of ionic liquids, and coacervates, as well as protein-ligand binding. We conclude that a promising new era of Martini CG simulations is coming.

2.4. Notes

This last section of the chapter contains a series of notes, which include useful information, limitations, and tips for problems that can arise using the Martini force field.

2.4.1. Limited stability of fluid phase and water freezing problem

This is a known consequence of the use of the LJ 12–6 potential for the non-bonded interactions. The thermodynamic behavior of solid-fluid and gas-fluid interfaces should therefore be interpreted with care, at least at the quantitative level. In applications where such interfaces are formed (especially water-vapor), these limitations have to be kept in mind. In biomolecular simulations, a related problem is the potential freezing of the Martini water model. The LJ parameters for water (5.0 kJ/mol and 0.47 nm) put the model into the solid-state region of the LJ phase diagram. However, the use of a shifted and truncated potential reduces the long-range attractive part and the CG water is more fluid compared to the standard LJ particle. While the freezing temperature is higher than it should be (around 290 K[4–6, 48]), in most applications, freezing is not observed as long as no nucleation site is formed. At lower temperatures, rapid freezing is a potential problem in systems where a nucleation site is already present (such as an ordered bilayer surface) or when periodicity enhances the long-range ordering. In these cases, a simple pragmatic solution is the addition of 10% antifreeze particles into the bulk water[6].

2.4.2. Entropy-enthalpy compensation

Martini parameterization is based on partition free energies. The inherent entropy loss on coarse-graining is necessarily compensated for by a reduced enthalpy term[26]. The enthalpy/entropy balance of many processes may therefore be biased when modeled at the CG level and affect its temperature dependence, although not necessarily weakening it. For instance, the temperature-dependent hydration free energy for linear alkanes was found to be more pronounced in the CG representation compared to an AA representation[26]. As is true for any force field, applications outside the temperature range used for parameterization (270–330 K) have to be considered with care. Although absolute entropies are clearly underestimated due to the loss of atomistic degrees of freedom, entropy differences can still be accurate[49].

2.4.3. "Sticky problem" in sugars and proteins

In the past years, the Martini force field has showed some specific limitations involving excessive interactions between certain classes of molecules. For example, protein-protein interactions in water solution seem to be overestimated[50]. To a lesser extent, this effect was also demonstrated for some transmembrane proteins[51]. Recently, mono, oligo- and polysaccharides were found to aggregate in simulations even at moderate concentrations, below their solubility limit[52]. These similar problems (called together here as "sticky problems") could be attenuated for pragmatic solutions, as down-scaling of the Lennard-Jones parameters between the solutes[50–52].

Another option is to increase the interactions with the solvent, which was successfully applied to study protein-crowded environments[53]. The usage of S-beads for the modeling of carbohydrate rings seems to reduce their aggregation propensity, as shown in the reparametrization of gangliosides[54]. All these possible procedures are not ideal solutions, as they are applied without a deeper understanding of the reasons behind the sticky problem. Besides, scaling factors are specific for classes of molecules and could potentially change important properties of the systems, as the correct binding mode of proteins dimers[51]. New rules for the usage of S- and T-beads together with reparameterization of Martini (e.g., including specific cross-interactions between standard and S/T bead sizes) showed to be crucial to reduce the excessive interactions (results not published yet). In the near future, all these features will be released as new version of the Martini force field (coined as version 3.0).

2.4.4. Electrostatic interactions and implicit screening

Another difficulty encountered in our CG model, and perhaps in most coarse-graining approaches, is the correct modeling of the partitioning of polar and charged compounds into a low dielectric medium. Because of the implicit screening, the interaction strength of polar substances is underestimated in nonpolarizable solvents. Applications involving the formation of polar/charged complexes in a nonpolar environment are especially prone to be affected. The inability to form a transmembrane water pore upon dragging a lipid across the membrane is an example[6, 55]. The development of a Martini water model that includes orientational polarization by the means of a dipole represented by two drude charges attached to each water bead allows to correct for some of these effects[31]. Apart from the implicit screening in the CG model, the neglect of long-range electrostatic forces poses a further limitation. Pairwise interactions beyond 1.1 nm (between 2 and 3 CG beads away) are not taken into account. In principle, long-range electrostatic interactions could be added to the CG model, in ways similar to those used in atomistic simulations[32]. In particular, PME in combination with the polarizable Martini water model is often used.

2.4.5. Fixed structure for proteins and nucleic acids

In applications of peptides, proteins, and nucleic acids, one has to be aware that structure transformations are not modeled in the current parameterization. For proteins, the secondary structure(SS) is essentially fixed by the use of bond angle and dihedral angle potential energy functions. The backbone bead type is also a function of the SS, to take into account the fact that when involved in interactions stabilizing a given element, the back-bone is less prompted to engage in other interactions. The backbone interaction strength is therefore decreased when involved in a SS element. This approach allows discrimination between various secondary structure elements but prevents realistic transitions between them. Processes in which folding and unfolding are playing a substantial role are therefore not suitable for modeling with the current Martini force field. However, movements of SS elements with respect to each other are possible and were shown to be quite realistic, for instance,

in modeling the gating of a membrane-embedded mechanosensitive channel[56]. In cases where the specificity of the local deformations of the protein backbone is important, the use of other approaches are necessary, as combining Martini with an elastic network[57, 58] or with structure-based CG models[59]. In the case of peripheral membrane proteins, further corrections in the side-chain dihedral angles could also be necessary[60]. Martini DNA and RNA can be used to model both single and double-stranded structures. For the single-stranded settings, the structure is considered flexible, while double-stranded could be modeled with two different elastic networks: a soft model which has a cutoff of 1.2 nm and a force constant of 13 kJ/mol/nm², and the stiff model which has a cutoff of 1.0 nm and a 500 kJ/mol/nm² force constant[12, 13].

2.4.6. Time step

Martini has been parameterized using time steps in the range of 10–40 fs. Whether you can use 40 fs or have to settle for a somewhat smaller time step depends on your system, and on your attitude toward coarse-grained modeling, as explained below. First, the Martini force field is not an atomistically detailed force field. Many assumptions underlie the model, the major one being the neglect of some of the atomistic degrees of freedom. As a result, the interactions between particles are effective ones and the energy landscape is highly simplified. This simplified energy landscape allows for a greatly increased sampling speed at the cost of a loss of detail. This makes CG models in general so powerful. The emphasis, therefore, should not be to sample the energy landscape as accurately as possible, but rather, as effectively as possible. This is in contrast to traditional all-atom models, for which the energy landscape is more realistic and an accurate integration scheme is more important. In practice, the inherent “fuzziness” of the Martini model makes the presence of small energy sinks or sources a less critical problem than in accurate atomistic simulations. Second and most importantly, structural properties are rather robust with respect to the time step; for a time step up to 40 fs, there are no noticeable effects on structural properties of the systems investigated. Moreover, thermodynamic properties such as the free energy of solvation also appear insensitive to the size of the time step. Thus, if the goal is to generate representative ensembles quickly, large time steps seem acceptable. Whereas one can debate the first argument (i.e., the “idealist” vs. “pragmatic” view of the power of CG simulations), the second argument (i.e., the insensitivity of both structural and thermodynamic properties to the magnitude of the time step) implies that a reduction of the time step to 10 fs or below, as has been suggested[61], is a waste of computer time[48]. Nevertheless, time steps of 40 fs and beyond may be pushing the limits too far for certain systems. For some systems, as nucleic acids, time steps higher than 10 fs promote simulation instability[12, 13]. We therefore recommend a timestep of 10–20 fs to be on the safe side. Of course, one should always check whether or not results are biased by the choices made. Given that the largest simplifications are made at the level of the interaction potentials, this can best be done by comparing to results obtained using more detailed models.

2.4.7. Effective timescale

The CG dynamics are faster than the AA dynamics, because the CG interactions are much smoother compared to atomistic interactions. The effective friction caused by the fine-grained degrees of freedom is missing. Based on comparison of diffusion constants for a range of systems (including simple solvents and lipids) in the CG model versus experimental data, the effective time sampled using CG interactions is three- to eightfold larger. When interpreting the simulation results with the CG model, a standard conversion factor of 4 has been used, which is the effective speed-up factor in the diffusion dynamics of CG water compared to real water. The same order of acceleration of the overall dynamics is also observed for a number of other processes, including the permeation rate of water across a membrane[4], the sampling of the local configurational space of a lipid[62], the aggregation rate of lipids into bilayers[4], and the self-diffusion of lipids[4, 6], transmembrane peptides[63], and proteins[64]. However, the speed-up factor can be quite different in other systems or for other processes, and in general no simple conversion of the time axis can be performed. Particularly for protein and nucleic acid systems, no extensive testing of the actual speed-up due to the CG dynamics has been performed, although protein translational and rotational diffusion was found to be in good agreement with experimental data in simulations of CG rhodopsin[64]. In general, the timescale of the simulations has to be interpreted with care.

2.5. Acknowledgments

The authors would like to thank the many people who have directly and indirectly contributed to the development of the Martini force field. In particular Alex de Vries, Helgi I. Ingolfsson, Manuel N. Melo, Tsjerk Wassenaar, Xavier Periole and all the past and present members of the MD group in Groningen, as well as the many users abroad, are acknowledged for their dynamism and enthusiasm in using, criticizing, and improving Martini.

2.6. References

- [1] B. M. H. Bruininks, P. C. T. de Souza, and S. J. Marrink, *A practical view of the martini force field*, in *Methods in Molecular Biology* (Springer New York, 2019) pp. 105–127.
- [2] S. J. Marrink and A. E. Mark, *The mechanism of vesicle fusion as revealed by molecular dynamics simulations*, *Journal of the American Chemical Society* **125**, 11144 (2003), pMID: 16220905, <https://doi.org/10.1021/ja036138+>.
- [3] S. J. Marrink and A. E. Mark, *Molecular dynamics simulation of the formation, structure, and dynamics of small phospholipid vesicles*, *Journal of the American Chemical Society* **125**, 15233 (2003).
- [4] S. J. Marrink, A. H. de Vries, and A. E. Mark, *Coarse grained model for semi-quantitative lipid simulations*, *The Journal of Physical Chemistry B* **108**, 750 (2004).
- [5] S. J. Marrink, J. Risselada, and A. E. Mark, *Simulation of gel phase formation and melting in lipid bilayers using a coarse grained model*, *Chemistry and Physics of Lipids* **135**, 223 (2005).
- [6] S. J. Marrink, H. J. Risselada, S. Yefimov, D. P. Tieleman, and A. H. de Vries, *The MARTINI force field: coarse grained model for biomolecular simulations*, *The Journal of Physical Chemistry B* **111**, 7812 (2007).
- [7] H. J. Risselada and S. J. Marrink, *The molecular face of lipid rafts in model membranes*, *Proceedings of the National Academy of Sciences* **105**, 17367 (2008).
- [8] L. Monticelli, S. K. Kandasamy, X. Periole, R. G. Larson, D. P. Tieleman, and S. J. Marrink, *The MARTINI coarse-grained force field: Extension to proteins*, *Journal of Chemical Theory and Computation* **4**, 819 (2008).
- [9] D. H. de Jong, G. Singh, W. F. D. Bennett, C. Arnarez, T. A. Wassenaar, L. V. Schäfer, X. Periole, D. P. Tieleman, and S. J. Marrink, *Improved parameters for the martini coarse-grained protein force field*, *Journal of Chemical Theory and Computation* **9**, 687 (2012).
- [10] C. A. López, A. J. Rzepiela, A. H. de Vries, L. Dijkhuizen, P. H. Hünenberger, and S. J. Marrink, *Martini coarse-grained force field: Extension to carbohydrates*, *Journal of Chemical Theory and Computation* **5**, 3195 (2009).
- [11] C. A. López, G. Bellesia, A. Redondo, P. Langan, S. P. S. Chundawat, B. E. Dale, S. J. Marrink, and S. Gnanakaran, *MARTINI coarse-grained model for crystalline cellulose microfibrils*, *The Journal of Physical Chemistry B* **119**, 465 (2015).
- [12] J. J. Uusitalo, H. I. Ingólfsson, P. Akhshi, D. P. Tieleman, and S. J. Marrink, *Martini coarse-grained force field: Extension to DNA*, *Journal of Chemical Theory and Computation* **11**, 3932 (2015).

- [13] J. J. Uusitalo, H. I. Ingólfsson, S. J. Marrink, and I. Faustino, *Martini coarse-grained force field: Extension to RNA*, *Biophysical Journal* **113**, 246 (2017).
- [14] D. H. de Jong, N. Liguori, T. van den Berg, C. Arnarez, X. Periole, and S. J. Marrink, *Atomistic and coarse grain topologies for the cofactors associated with the photosystem II core complex*, *The Journal of Physical Chemistry B* **119**, 7791 (2015).
- [15] E. Panizon, D. Bochicchio, L. Monticelli, and G. Rossi, *MARTINI coarse-grained models of polyethylene and polypropylene*, *The Journal of Physical Chemistry B* **119**, 8209 (2015).
- [16] G. Rossi, P. F. J. Fuchs, J. Barnoud, and L. Monticelli, *A coarse-grained MARTINI model of polyethylene glycol and of polyoxyethylene alkyl ether surfactants*, *The Journal of Physical Chemistry B* **116**, 14353 (2012).
- [17] G. Rossi, L. Monticelli, S. R. Puisto, I. Vattulainen, and T. Ala-Nissila, *Coarse-graining polymers with the MARTINI force-field: polystyrene as a benchmark case*, *Soft Matter* **7**, 698 (2011).
- [18] R. Alessandri, J. J. Uusitalo, A. H. de Vries, R. W. A. Havenith, and S. J. Marrink, *Bulk heterojunction morphologies with atomistic resolution from coarse-grain solvent evaporation simulations*, *Journal of the American Chemical Society* **139**, 3697 (2017).
- [19] J. Liu, L. Qiu, R. Alessandri, X. Qiu, G. Portale, J. Dong, W. Talsma, G. Ye, A. A. Sengrhan, P. C. T. de Souza, M. A. Loi, R. C. Chiechi, S. J. Marrink, J. C. Hummelen, and L. J. A. Koster, *Enhancing molecular n-type doping of donor-acceptor copolymers by tailoring side chains*, *Advanced Materials* **30**, 1704630 (2018).
- [20] L. Qiu, J. Liu, R. Alessandri, X. Qiu, M. Koopmans, R. W. A. Havenith, S. J. Marrink, R. C. Chiechi, L. J. A. Koster, and J. C. Hummelen, *Enhancing doping efficiency by improving host-dopant miscibility for fullerene-based n-type thermoelectrics*, *J. Mater. Chem. A* **5**, 21234 (2017).
- [21] F. Grunewald, G. Rossi, A. H. de Vries, S. J. Marrink, and L. Monticelli, *Transferable MARTINI model of poly(ethylene oxide)*, *The Journal of Physical Chemistry B* **122**, 7436 (2018).
- [22] L. Monticelli, *On atomistic and coarse-grained models for c60 fullerene*, *Journal of Chemical Theory and Computation* **8**, 1370 (2012).
- [23] H. I. Ingólfsson, P. Tieleman, and S. Marrink, *Lipid organization of the plasma membrane*, *Biophysical Journal* **108**, 358a (2015).
- [24] F. J. V. Eerden, M. N. Melo, P. W. J. M. Frederix, X. Periole, and S. J. Marrink, *Exchange pathways of plastoquinone and plastoquinol in the photosystem II complex*, *Nature Communications* **8** (2017), 10.1038/ncomms15214.

- [25] X. Periole and S. J. Marrink, *The martini coarse-grained force field*, in *Methods in Molecular Biology* (Humana Press, 2012) pp. 533–565.
- [26] R. Baron, D. Trzesniak, A. H. de Vries, A. Elsener, S. J. Marrink, and W. F. van Gunsteren, *Comparison of thermodynamic properties of coarse-grained and atomic-level simulation models*, *ChemPhysChem* **8**, 452 (2007).
- [27] A. Y. Shih, A. Arkhipov, P. L. Freddolino, and K. Schulten, *Coarse grained protein-lipid model with application to lipoprotein particles†*, *The Journal of Physical Chemistry B* **110**, 3674 (2006).
- [28] A. Y. Shih, P. L. Freddolino, A. Arkhipov, and K. Schulten, *Assembly of lipoprotein particles revealed by coarse-grained molecular dynamics simulations*, *Journal of Structural Biology* **157**, 579 (2007).
- [29] D. H. de Jong, S. Baoukina, H. I. Ingólfsson, and S. J. Marrink, *Martini straight: Boosting performance using a shorter cutoff and GPUs*, *Computer Physics Communications* **199**, 1 (2016).
- [30] C. A. López, A. H. de Vries, and S. J. Marrink, *Computational microscopy of cyclodextrin mediated cholesterol extraction from lipid model membranes*, *Scientific Reports* **3** (2013), 10.1038/srep02071.
- [31] S. O. Yesylevskyy, L. V. Schäfer, D. Sengupta, and S. J. Marrink, *Polarizable water model for the coarse-grained MARTINI force field*, *PLoS Computational Biology* **6**, e1000810 (2010).
- [32] J. Michalowsky, L. V. Schäfer, C. Holm, and J. Smiatek, *A refined polarizable water model for the coarse-grained MARTINI force field with long-range electrostatic interactions*, *The Journal of Chemical Physics* **146**, 054501 (2017).
- [33] T. Bereau and K. Kremer, *Automated parametrization of the coarse-grained martini force field for small organic molecules*, *Journal of Chemical Theory and Computation* **11**, 2783 (2015).
- [34] R. Menichetti, K. H. Kanekal, K. Kremer, and T. Bereau, *In silico screening of drug-membrane thermodynamics reveals linear relations between bulk partitioning and the potential of mean force*, *The Journal of Chemical Physics* **147**, 125101 (2017).
- [35] K. Kawabata, Y. Takakura, and M. Hashida, *Pharmaceutical Research* **12**, 825 (1995).
- [36] Y. Zhao and L. Huang, *Lipid nanoparticles for gene delivery*, in *Nonviral Vectors for Gene Therapy - Lipid- and Polymer-based Gene Transfer* (Elsevier, 2014) pp. 13–36.
- [37] M. Ramamoorth, *Non viral vectors in gene therapy- an overview*, *JOURNAL OF CLINICAL AND DIAGNOSTIC RESEARCH* (2015), 10.7860/jc-dr/2015/10443.5394.

- [38] S. Chira, C. S. Jackson, I. Oprea, F. Ozturk, M. S. Pepper, I. Diaconu, C. Braicu, L.-Z. Raduly, G. A. Calin, and I. Berindan-Neagoe, *Progresses towards safe and efficient gene therapy vectors*, *Oncotarget* **6**, 30675 (2015).
- [39] L. Ciani, S. Ristori, A. Salvati, L. Calamai, and G. Martini, *DOTAP/DOPE and DC-chol/DOPE lipoplexes for gene delivery: zeta potential measurements and electron spin resonance spectra*, *Biochimica et Biophysica Acta (BBA) - Biomembranes* **1664**, 70 (2004).
- [40] M. J. Abraham, T. Murtola, R. Schulz, S. Páll, J. C. Smith, B. Hess, and E. Lindahl, *GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers*, *SoftwareX* **1-2**, 19 (2015).
- [41] W. Humphrey, A. Dalke, and K. Schulten, *VMD: Visual molecular dynamics*, *Journal of Molecular Graphics* **14**, 33 (1996).
- [42] S. J. Marrink and A. E. Mark, *Molecular view of hexagonal phase formation in phospholipid membranes*, *Biophysical Journal* **87**, 3894 (2004).
- [43] J. Corsi, R. W. Hawtin, O. Ces, G. S. Attard, and S. Khalid, *DNA lipoplexes: Formation of the inverse hexagonal phase observed by coarse-grained molecular dynamics simulation*, *Langmuir* **26**, 12119 (2010).
- [44] T. A. Wassenaar, H. I. Ingólfsson, R. A. Böckmann, D. P. Tieleman, and S. J. Marrink, *Computational lipidomics with insane: A versatile tool for generating custom membranes for molecular simulations*, *Journal of Chemical Theory and Computation* **11**, 2144 (2015).
- [45] L. Martínez, R. Andrade, E. G. Birgin, and J. M. Martínez, *PACKMOL: A package for building initial configurations for molecular dynamics simulations*, *Journal of Computational Chemistry* **30**, 2157 (2009).
- [46] Y. Qi, X. Cheng, and W. Im, *CHARMM-GUI martini maker for coarse-grained simulations*, *Biophysical Journal* **108**, 161a (2015).
- [47] P.-C. Hsu, B. M. H. Bruininks, D. Jefferies, P. C. T. de Souza, J. Lee, D. S. Patel, S. J. Marrink, Y. Qi, S. Khalid, and W. Im, *CHARMM-GUI martini maker for modeling and simulation of complex bacterial membranes with lipopolysaccharides*, *Journal of Computational Chemistry* **38**, 2354 (2017).
- [48] S. J. Marrink, X. Periole, D. P. Tieleman, and A. H. de Vries, *Comment on "on using a too large integration time step in molecular dynamics simulations of coarse-grained molecular models" by m. winger, d. trzesniak, r. baron and w. f. van gunsteren*, *phys. chem. chem. phys.*, 2009, 11, 1934, *Phys. Chem. Chem. Phys.* **12**, 2254 (2010).
- [49] Y. Yano and K. Matsuzaki, *Measurement of thermodynamic parameters for hydrophobic mismatch 1: self-association of a transmembrane helix[†]*, *Biochemistry* **45**, 3370 (2006).

- [50] A. C. Stark, C. T. Andrews, and A. H. Elcock, *Toward optimized potential functions for protein–protein interactions in aqueous solutions: Osmotic second virial coefficient calculations using the MARTINI coarse-grained force field*, *Journal of Chemical Theory and Computation* **9**, 4176 (2013).
- [51] M. Javanainen, H. Martinez-Seara, and I. Vattulainen, *Excessive aggregation of membrane proteins in the martini model*, *PLOS ONE* **12**, e0187936 (2017).
- [52] P. S. Schmalhorst, F. Deluweit, R. Scherrers, C.-P. Heisenberg, and M. Sikora, *Overcoming the limitations of the MARTINI force field in simulations of polysaccharides*, *Journal of Chemical Theory and Computation* **13**, 5039 (2017).
- [53] B. Liu, C. Åberg, F. J. van Eerden, S. J. Marrink, B. Poolman, and A. J. Boersma, *Design and properties of genetically encoded probes for sensing macromolecular crowding*, *Biophysical Journal* **112**, 1929 (2017).
- [54] R.-X. Gu, H. I. Ingólfsson, A. H. de Vries, S. J. Marrink, and D. P. Tieleman, *Ganglioside-lipid and ganglioside-protein interactions revealed by coarse-grained and atomistic molecular dynamics simulations*, *The Journal of Physical Chemistry B* **121**, 3262 (2016).
- [55] W. D. Bennett and D. P. Tieleman, *Water defect and pore formation in atomistic and coarse-grained lipid membranes: Pushing the limits of coarse graining*, *Journal of Chemical Theory and Computation* **7**, 2981 (2011).
- [56] M. N. Melo, C. Arnarez, H. Sikkema, N. Kumar, M. Walko, H. J. C. Berendsen, A. Kocer, S. J. Marrink, and H. I. Ingólfsson, *High-throughput simulations reveal membrane-mediated effects of alcohols on MscL gating*, *Journal of the American Chemical Society* **139**, 2664 (2017).
- [57] X. Periole, M. Cavalli, S. J. Marrink, and M. A. Ceruso, *Combining an elastic network with a coarse-grained molecular force field: Structure, dynamics, and intermolecular recognition*, *Journal of Chemical Theory and Computation* **5**, 2531 (2009).
- [58] I. Siuda and L. Thøgersen, *Conformational flexibility of the leucine binding protein examined by protein domain coarse-grained molecular dynamics*, *Journal of Molecular Modeling* **19**, 4931 (2013).
- [59] A. B. Poma, M. Cieplak, and P. E. Theodorakis, *Combining the MARTINI and structure-based coarse-grained approaches for the molecular dynamics studies of conformational transitions in proteins*, *Journal of Chemical Theory and Computation* **13**, 1366 (2017).
- [60] F. A. Herzog, L. Braun, I. Schoen, and V. Vogel, *Improved side chain dynamics in MARTINI simulations of protein–lipid interfaces*, *Journal of Chemical Theory and Computation* **12**, 2446 (2016).

- [61] M. Winger, D. Trzesniak, R. Baron, and W. F. van Gunsteren, *On using a too large integration time step in molecular dynamics simulations of coarse-grained molecular models*, *Physical Chemistry Chemical Physics* (2009), 10.1039/b818713d.
- [62] R. Baron, A. H. de Vries, P. H. Hünenberger, and W. F. van Gunsteren, *Configurational entropies of lipids in pure and mixed bilayers from atomic-level and coarse-grained molecular dynamics simulations*, *The Journal of Physical Chemistry B* **110**, 15602 (2006).
- [63] S. Ramadurai, A. Holt, L. V. Schäfer, V. V. Krasnikov, D. T. Rijkers, S. J. Marrink, J. A. Killian, and B. Poolman, *Influence of hydrophobic mismatch and amino acid composition on the lateral diffusion of transmembrane peptides*, *Biophysical Journal* **99**, 1447 (2010).
- [64] X. Periole, T. Huber, S. J. Marrink, and T. P. Sakmar, *G protein-coupled receptors self-assemble in dynamics simulations of model bilayers*, *Journal of the American Chemical Society* **129**, 10126 (2007).

3

CHARMM-GUI Martini Maker for modeling and simulation of complex bacterial membranes with lipopolysaccharides

I truly believe truth does not exist.

Albert Bruininks

A complex cell envelope, composed of a mixture of lipid types including complex lipopolysaccharides, protects bacteria from the external environment. Clearly, the proteins embedded within the various components of the cell envelope have an intricate relationship with their local environment. Therefore, to obtain meaningful results, molecular simulations need to mimic as far as possible this chemically heterogeneous system. However, setting up such systems for computational studies is far from trivial, and consequently the vast majority of simulations of outer membrane proteins still rely on oversimplified phospholipid membrane models. This work presents an update of CHARMM-GUI Martini Maker for coarse-grained modeling and simulation of complex bacterial membranes with lipopolysaccharides. The qualities of the outer membrane systems generated by Martini Maker are validated by simulating them in bilayer, vesicle, nanodisc, and micelle environments (with and without outer membrane proteins) using the Martini force field. We expect this new feature

This chapter has been published in the Journal of Computational Chemistry (2017)[1].

in Martini Maker to be a useful tool for modeling large, complicated bacterial outer membrane systems in a user-friendly manner.

Keywords: Coarse-grained simulation, Outer membrane vesicles, Bilayers, Nanodiscs, Micelles

3.1. Introduction

Computational studies and in particular molecular dynamics (MD) simulations are now a firmly established technique for the study of biological membranes. At the all-atom or united-atom level, molecular simulations have been successfully used to study various membrane properties, the permeation of small molecules directly through model membranes, transport of ions and small molecules through specific channels, and the dynamics of membrane protein structures solved in different environments[2–5].

Over the past decade, the biological simulation community has seen a widespread embracing of coarse-grained (CG) models and methods, which have widened the scope of simulations by allowing access to longer time and length scales. The Martini force field developed by Marrink and co-workers[6, 7] is perhaps the most widely used model for CG simulations of biological membranes. Some examples of recent successes are combination of the force field with the metadynamics method of enhanced sampling to study the energetics of conformational rearrangements in the epidermal growth factor receptor[8]; high throughput simulations showing the effects of alcohol on a mechanosensitive protein[9]; studies of local phase transitions in bacterial membranes induced by an antimicrobial peptide[4], and unraveling of the plastoquinone exchange pathways of the photosystem II complex[10]. A key advantage of CG models is the ability to self-assemble the lipid component whether this be a flat bilayer, a micelle, or a small spherical vesicle in the presence or absence of the proteins of interest, thereby eliminating the initial ‘guesswork’ of determining how proteins are oriented and packed by the lipids and/or detergents in each environment. However, such a spontaneous self-assembly becomes rather more complex and questionable when the membranes are asymmetric or composed of lipids with large, complex carbohydrate components such as lipopolysaccharides (LPS) as in the case of bacterial membranes[11].

Bacterial membranes are complex in terms of their chemical composition. In particular, the cell envelope surrounding the cytoplasm of Gram-negative bacteria is composed of the inner membrane, the periplasm, and the outer membrane (OM). The OM is a unique asymmetrical bilayer composed of LPS molecules in the outer leaflet and a mixture of zwitterionic and anionic phospholipids in the inner leaflet[12, 13]. The inner membrane is more or less symmetric, and both leaflets closely resemble the inner leaflet of the OM in terms of the phospholipid composition[14]. Given that modification of these membranes is one of the ways that bacteria achieve resistance to our current arsenal of antibiotics[15, 16], studying the relationship between the membrane/membrane proteins and drugs is imperative for the rational design of novel antibiotics[5, 17–19]. However, simulation studies of biologically relevant bacterial OM models are complicated by the chemical complexity of the membrane constituents, in particular, the LPS molecules[14, 20–23]. Detailed atomistic simulations of the OM have only recently become widespread[24–28], and consequently the first CG models of these membranes are only just being reported in the last couple of years[29, 30].

CHARMM-GUI¹ is a web-based graphical user interface (GUI) to prepare complex

¹<http://www.charmm-gui.or>

molecular simulation systems and input files to facilitate the usage of common and advanced simulation techniques[31]. Recently, taking advantage of the frameworks in all-atom CHARMM-GUI modules[32–34], CHARMM-GUI Martini Maker² was developed to prepare various CG simulation systems in solution, bilayer, micelle, nanodisc, and vesicle environments using the Martini force field[35]. Although more than 200 lipid types are available in Martini Maker, incorporating the CG LPS molecules in the system building process poses new challenges and thus requires specific methods for system size determination and counterion placement. To address the practical difficulties of setting up simulation systems with LPS molecules, we have updated Martini Maker to automate the building process of LPS-containing complex bilayer, micelle, nanodisc, and vesicle systems, as well as systems with randomly distributed LPS molecules. In this work, the update in Martini Maker is described and its robustness is tested by building and simulating various LPS-containing systems to illustrate the simulation contents that one can perform with Martini Maker.

3.2. Methods

3.2.1. CHARMM-GUI implementation

Martini models of two different LPS, Ra LPS (RAMP) and Re LPS (REMP), were added in CHARMM-GUI Martini Maker (Fig. 3.1). The LPS models follow a 4-to-1 mapping scheme of the Martini force field and the parameters were optimized based on united-atom LPS simulations to improve accuracy[36, 37].

The overall building procedures of all LPS-containing Martini Maker modules (Bilayer/Nanodisc/Vesicle/Micelle/Random Builders) are identical to the original implementation[35]. Briefly, in STEP 1, a user-specified protein structure is read-in through PDB Reader. In STEP 2, the protein orientation is changed based on the user-specific input; by definition, the Z axis is the membrane normal and $Z = 0$ is the membrane center. In STEP 3, the system size is determined, and the pseudo spheres are placed for assigning lipid head group positions. Note that this is the first step when the membrane-only generation option is selected. In STEP 4, the system components (lipids, water, and ions) are generated. Finally, all the components are assembled in STEP 5. During STEP 5, the CHARMM structure (PSF) and coordinate (CRD/PDB) files of each component generated in STEP 4 are merged into single PSF and CRD/PDB files, and water beads in close proximity to the solutes are removed.

Some LPS-specific changes were introduced in the system size calculation and ion placement steps. As described above, the system size was previously determined in STEP 3. However, as the LPS molecule has a long carbohydrate chain, a portion of the LPS molecule can be stretched out beyond the system box determined in STEP 3 based on phospholipids. To resolve this issue, if the system contains LPS, the system size is recalculated by taking the LPS height into account in STEP 4, and the updated system size information is used for further steps (building water box and placing ions).

As divalent cations play an important role in stabilizing the bacterial OM by interacting with the LPS[23, 38–41], the ion placement procedure in STEP 4 was modified

²<http://www.charmm-gui.org/input/martini>

Figure 3.1 | Atomistic and coarse LPS. Structural comparison between (A) *E. coli* R3-core type LPS and corresponding Martini (B) Ra and (C) Re LPS models. The carbohydrate names are: Kdo for 2-keto-3-deoxyoctulosonate, Hep for L-glycero-D-manno heptose, Glc for D-glucose, Gal for D-galactose, and GalNAc for N-acetyl-D-galactosamine. The lipid A molecule in this study consists of two D-glucosamine (GlcN) residues joined by a β -(1-6)-linkage, two monophosphoester groups at O1 and O4', and six amide/ester-linked fatty acids. Lipid A head and tail groups are colored in red and light blue, respectively. Two Kdo (Re glycans) are colored in dark gray, and the rest (Ra) glycans are colored in light gray. Phosphates and Kdo carboxyl groups are colored in tan and dark blue, respectively.

to use Ca^{2+} as the counterions for LPS lipid A and core oligosaccharides. By default, CHARMM-GUI adds Ca^{2+} ions to neutralize lipid A, but for the LPS core, CHARMM-GUI provides an option to select an ion type (Na^+ or Ca^{2+}).

3.2.2. Martini force fields used in this study

The standard lipid parameters for palmitoyl-oleoyl-phosphatidylethanolamine (POPE) and palmitoyl-oleoyl-phosphatidylglycerol (POPG) were taken from Wassenaar et al.[42] and those for CDL2 (a cardiolipin with a net charge of $-2e$) from Dahlberg and Maliniak[43]. The parameters for Na^+ and Cl^- were taken from Marrink et al.[6, 7] For Ca^{2+} , a well-tested Martini model is not available yet. Here, Ca^{2+} was simply modeled as Na^+ with $+2e$, as this has been used before in other published studies[44]. As in most applications, the standard water model was taken from Marrink et al.[45] The parameters for LPS are those defined in Hsu et al.[36] Note that a few bonds

with large force constants in the original LPS models were replaced with constraints to improve stability and allow larger integration time steps[7]. For the proteins, the Martini 2.1 protein force field was used[46] in combination with an elastic network[47]. The common settings associated with the Martini model were used to perform the simulations, including a 12 Å cutoff for the non-bonded interactions using shifted potentials[48]. In this study, unless specified explicitly, all NPT (constant particle number, pressure and temperature) simulations were performed at 310 K, atmospheric pressure of 1 bar, and physiological salt concentration (150 mM NaCl for bulk solution with additional Na⁺ or Ca²⁺ ions to neutralize the LPS core and Lipid A, respectively). The systems generated by Martini Maker using default options (unless specified explicitly) were energy-minimized and equilibrated using the default settings of the README output file and the GROMACS 5.1 molecular dynamics package[49].

3.2.3. Bilayer systems

Three CG OM systems were constructed to test Bilayer Builder in Martini Maker; Table S3.1 provides a summary of the system information. The membrane composition mimics the *E. coli* OM and contains Ra LPS in the outer leaflet and a composition of POPE, POPG, and CDL2 at a ratio of 18:1:1 at the inner leaflet[18, 42, 43, 50]. Two of the OM systems contained an inserted β -barrel outer membrane protein (OMP), the monomeric OmpA (PDB:1QJP)[51] or the trimeric OmpF (PDB:3POX)[52]; see Fig. 3.2 for their structures. Both are major OMPs in *E. coli* as identified by proteomics studies[53, 54]. The proteins were inserted into the OM using the orientation predefined in the OPM database[55]. The OM-only and the OM-OmpA systems were set to be $100 \times 100 \text{ Å}^2$ in XY, the OM-OmpF system was bigger ($150 \times 150 \text{ Å}^2$) due to the relatively large size of the OmpF trimer. All systems initially had a layer of 45 Å of Martini water between the periodic images of the OM. After equilibration, each system was simulated for 5- to 9- μ s production with a time step of 20 fs.

In this work, the membrane thickness was defined as the distance between the average Z positions of phosphate atoms in each leaflet and calculated using MDAnalysis[56]. For Ra LPS (containing phosphates in Hep sugars), only the Lipid A phosphates were used for the thickness calculation. The area per lipid of each lipid type was determined by two-dimensional Voronoi analysis using the phosphate group of each lipid and an in-house adaptation of the pyvoro library (created by Joe Jordan) and MDAnalysis; for LPS, PO1 and PO2 in lipid A were used. All protein beads within a distance of 10 Å from any selected phosphate groups of neighboring lipids were used for the protein-lipid interface. Only the last microsecond of the production run was used to measure this value. In all cases the leaflets were analyzed separately. All means and standard deviations were calculated using GROMACS 5.1 gmx analyze[49]. Visual inspection, image rendering, and calculation of the root-mean-square deviation (RMSD) of the protein from their initial structure were performed with VMD[57], and in case of the RMSD, the full production run of 5- μ s was used.

3.2.4. Nanodisc systems

Two symmetric discoidal LPS nanodiscs with neutralizing Na⁺ or Ca²⁺ ions were constructed with Nanodisc Builder in Martini Maker; see Table S3.2 for the system infor-

Figure 3.2 | Outer membrane protein A and F. Top view structures of outer membrane proteins OmpA (PDB ID: 1QJP) and trimeric OmpF porins (PDB ID: 3POX). Four and Eight loops of OmpA and OmpF are also depicted in different colors. Red and blue spheres in the OmpF porins are acidic and basic residues in the constriction zone.

mation. In each system, the discoidal LPS bilayers were encased by two membrane scaffold protein MSP1D1. Given the nanodisc area by MSP1D1 (a diameter of 95 Å), the first system was made with 37 Re LPS molecules per leaflet, and the second system was built with 33 Ra LPS molecules per leaflet. After the equilibration, we performed 1-μs production runs with a time step of 20 fs. The membrane thickness analysis was done on a per-phosphate basis through the use of two-dimensional Voronoi diagrams and Delaunay triangulations as a function of distance from the nanodisc center.

3

3.2.5. Vesicle systems

Two outer membrane vesicle (OMV)-only and two OMV-OmpF systems of different diameters (150 Å or 100 Å) were constructed with Vesicle Builder in Martini Maker; see Table S3.3 for the system information. The outer leaflet of the vesicles contained either Re or Ra LPS molecules, while the inner leaflets contained POPE, POPG, and CDL2 lipids at a ratio of 18:1:1. Following energy minimization and NPT equilibration, each vesicle system was simulated for several μs with a time step of 20 fs (Table S3.3). Before the production run, the vesicle was built with two pores on the surface to allow free movement of water particles to equilibrate the interior and exterior water[58]. The vesicle properties were determined using the last 500 ns of simulation time. The bilayer thickness was interpreted using radial distribution function (RDF) of the selected phosphate groups in the inner and outer leaflets of the OMV with respect to the center of mass of vesicle, and the area per lipid was estimated by calculating Voronoi cells for each of these isolated lipid phosphate groups[59]. Vesicle radii were estimated using an in-house script by calculating the distance between the center of mass of the OM and the vesicle center[60]. The membrane density profiles were measured as the position of the lipid phosphate groups along the bilayer normal using a bin width of 1 Å.

3.2.6. Micelle and random systems

To demonstrate the robustness and flexibility of Micelle and Random Builders with LPS CG models, we constructed and simulated various systems with different numbers of Ra or Re LPS molecules and with/without OmpF; see Table S3.4 for the system information. The LPS-only systems contained 5, 10, 15, and 20 LPS molecules in a micelle form or randomly distributed in 0.15 M NaCl solution with neutralizing Ca²⁺ ions added near to the Lipid A phosphate groups. The OmpF micelle and random systems were composed of a monomer of OmpF and 40 Ra LPS molecules at the same salt concentration. After the conventional CHARMM-GUI protocol for minimization and equilibration of the systems, we performed 10 μs production runs with a time step of 20 fs. As simulations with 20 Ra LPS (starting from a preformed micelle or a random distribution) showed some discrepancies regarding the shape of aggregates formed, we also performed additional simulations: i) simulations without Ca²⁺ ions, but replaced by Na⁺; ii) simulations with polarizable Martini water model[61]; iii) simulations at higher temperature (400 K) and NVT (constant particle number, volume, and temperature) ensemble (with the volume equal to the average values obtained in the NPT simulations at 310 K and 1 bar). All average properties were

calculated over the last 2 μs of the simulations.

3.3. Results and discussion

Martini (Re and Ra) LPS models (Fig. 3.1) have been added to the following Martini Maker modules: Bilayer, Nanodisc, Vesicle, Micelle, and Random Builder. In this section, the robustness and applications of these modules are tested and illustrated by a wide content of practical simulation studies and their analyses.

3.3.1. Bilayer systems

Martini Maker has shown its capability to construct complex plasma membranes (with or without a membrane protein) using Bilayer Builder[35]. Here, to illustrate its ability to build complex, asymmetric OMs, three representations of the *E. coli* OMs were constructed, constituted of pure Ra LPS in the outer leaflet and a mixture of POPE, POPG, and CDL2 in the inner leaflet (see Methods). The OM-only, OM-OmpA, and OM-OmpF systems (Fig. 3.3 A, B) contained a total of 216, 205, and 402 lipids fully hydrated in 150 mM NaCl solution. The negative charges of Ra LPS were neutralized by a mixture of Ca^{2+} and Na^+ . All bilayers were rather rigid in the outer leaflet, with little to no diffusion of Ra LPS, in the course of the entire simulation. This caused the inserted protein to stay trapped between its initial neighboring Ra LPS molecules throughout the production run. In the lower leaflet, lipids diffuse freely and do not phase separate. None of the proteins left the OM during any of the runs.

Figure 3.3 | Outer membrane with and without protein. (A, B) The side and top view of the final snapshot of 5 μs simulations (at 310 K and 1 bar) of different Ra LPS systems. The unit cells are depicted by a blue dotted wireframe. (C) The OM thickness varied only little over the span of the simulation for each system (the red line shows the running average; $n=20$).

The objective of the LPS Bilayer Builder extension was to generate an initial structure that is close to an equilibrated state of an LPS-containing bilayer. Therefore, we investigated three bilayer properties. First, the OM thickness was defined as the distance between the phosphate groups of the inner leaflet and the phosphate groups of the lipid A in the outer leaflet. The average bilayer thickness and standard error over the last μs of each system are 37.61 ± 0.15 (OM-only), 37.25 ± 0.20 (OM-OmpA), and 37.18 ± 0.12 Å (OM-OmpF). For all three systems, the change in bilayer thickness over 5 μs is very small (less than 1 %). Interestingly, even though the change is less than 0.5 Å compared to the OM-only simulation, the insertion of OmpA and OmpF appears to slightly reduce the overall bilayer thickness. This is probably due to hydrophobic mismatch at the protein-lipid interface.

The second property we tested is the area per lipid (APL). The change in APL is slightly positive (<5%) for all lipids in all systems over 5 μs . POPG and CDL2 show the highest change probably due to their low copy number (Fig. S3.1). A table containing the average APL over the last μs for each lipid in each system can be found in Table S3.5. The slight increase in APL corresponds well with the observed slight decrease in the OM thickness. As for the OM thickness, the initial APL appears to be close to its equilibrium value.

The third property is the protein integrity. Since the Martini ElnDyn elastic network used for OmpA and OmpF should preserve the global protein structure, the protein integrity was examined by calculating the backbone RMSD with respect to the CG crystal structure. The calculated RMSD for OmpA (1.60 ± 0.05 Å) and OmpF (1.88 ± 0.07 Å) was in the same order of magnitude reported in the original CG Martini ElnDyn publication[47]. Therefore, the insertion process of the protein in Bilayer Builder does not alter the protein to any relevant extent.

All of the bilayers tested are stable. The membrane thickness and APL after 5 μs are close to their initial values, which normally indicates that the generated initial configuration is close to an equilibrium state of an LPS-containing bilayer. However, due to the extremely slow dynamics of Ra LPS and the measured small drift in the bilayer thickness and APL, it is difficult to prove that we are indeed close to equilibrium. This is a general difficulty of working with Ra LPS or any model with extremely slow dynamics and has little to do with the capacities of Bilayer Builder. In addition, through Bilayer Builder, users can change the number of lipids or each lipid's initial area in each leaflet, so that one can examine various properties of asymmetric bilayers of their own interest.

3.3.2. Nanodisc systems

Discoidal lipid/protein particles, termed nanodiscs, are synthetic model membrane systems which are useful in the study of membrane proteins and native membrane environments[62, 63]. Each nanodisc generally contains two amphipathic membrane scaffold proteins (MSPs) that encase and thereby support a cylindrical lipid bilayer. Nanodiscs are increasingly being used as platforms for investigating integral membrane proteins such as bacteriorhodopsin, cytochrome P450, and G-protein coupled receptors[64]. While the application of these discoidal lipid/protein particles is largely limited to simple phospholipids, the incorporation of LPS molecules into nanodiscs

would enable the fabrication and investigation of more realistic Gram-negative OM environments. To provide insight into the dynamic properties of LPS nanodiscs, and to test the capabilities of Nanodisc Builder in Martini Maker, we constructed nanodiscs of Re or Ra LPS that were encased in two MPS1D1 proteins (see Methods).

The RMSD of the MSPs reached asymptotic values within the simulation time, indicating that the nanodisc properties had converged (Fig. S3.2). The equilibrium RMSD values are similar in the Ra LPS nanodisc systems; 4.3 ± 0.2 and 4.3 ± 0.3 Å (for each MSP with neutralizing Ca^{2+} ions) and 4.6 ± 0.2 and 4.1 ± 0.2 Å (with neutralizing Na^+ ions) over the last 100 ns of simulation time. For comparison, the RMSD values in the Re LPS nanodisc systems are 4.7 ± 0.3 and 4.2 ± 0.4 Å (Ca^{2+} ions) and 5.2 ± 0.4 and 5.0 ± 0.3 Å (Na^+ ions). The data reveal a slight reduction in the MSP stability when they encase LPS molecules of smaller head groups, and likewise reveal a slight decrease in protein stability when divalent cations are not present to stabilize the repulsive electrostatic interactions between LPS phosphate and carboxylate groups.

Fig. 3.4 shows the bilayer thickness as a function of radial distance (R) from the nanodisc center for each nanodisc system. While the Ra LPS nanodisc bilayers generally preserve their thickness as a function of R, the Re LPS nanodisc bilayers taper towards the nanodisc edge. This deformation of lamellar structure is rationalized in terms of the incomplete coverage afforded by the MSPs that encase LPS acyl tails, and the relatively small size of the Re LPS lipid head groups. The Re LPS lipid head groups are able to perforate small gaps in the amphipathic protein belts leading to deformation of bilayer structure and reduction of the membrane thickness towards the disc edge (Fig. 3.4 E, F). This insight has important implications for experiments involving LPS nanodiscs, as it is evident that realistic Gram-negative OM environments are more easily achieved using LPS variants with bulkier core oligosaccharide sections, which are less able to perforate small holes in the encasing protein belts. Note that the symmetric LPS nanodisc bilayers are thinner than the planar OM thickness (Fig. 3.3) because the OM contains phospholipids with longer acyl chains in the inner leaflet. The average thickness of the nanodisc bilayers within $R = 20$ Å are 28.46 ± 0.74 Å (Ra LPS with Ca^{2+}), 28.32 ± 0.82 Å (Ra LPS with Na^+), 34.65 ± 0.34 Å (Re LPS with Ca^{2+}), and 35.89 ± 1.32 Å (Re LPS with Na^+).

3.3.3. Vesicle systems

Martini Maker has previously been used to manufacture multifaceted membrane systems that incorporate different membrane proteins[35]. Here, we built outer membrane vesicles (OMVs) that were made of lipids alone (OMV-only), and complex vesicles that incorporate lipids and embedded OmpF (OMV-OmpF). Vesicle radii of 100 and 150 Å were studied for each OMV composition (Fig. 3.5 A, B). The outer leaflet of the vesicles contained Re or Ra LPS lipids, and the inner leaflet contained a combination of POPE, POPG, and CDL2 (see Methods). To assess simulation convergence, we monitored the radius of each OMV (R_{OMV}), which was defined as the radial distance between the OMV center and the center of mass of the encapsulating membranes. The R_{OMV} from the last 500 ns simulations are 142.36 ± 0.31 (150 Å Re OMV-only), 141.96 ± 0.03 (150 Å Re OMV-OmpF), 140.07 ± 0.02 (150 Å Ra OMV-only), and

Figure 3.4 | LPS nanodiscs. Nanodisc bilayer thickness as a function of radial distance from the nanodisc center for the Ra LPS systems with (A) Na^+ or (B) Ca^{2+} neutralizing ions and for the Re LPS systems with (C) Na^+ or (D) Ca^{2+} neutralizing ions. The thickness values are calculated for the last 100 ns of the simulations. The error bars represent one standard deviation. (E) Side view of the symmetric Re LPS nanodisc after 1 μs of simulation time. The component Re LPS molecules are colored white, and the encasing MSP1D1 helices are represented as red and green isosurfaces, extracted from a volumetric Gaussian density map. (F) The symmetric Ra LPS nanodisc with the MSPs omitted to accentuate the Re LPS molecules that have perforated gaps in the encasing MSP helices. These perforating Re LPS lipids are colored yellow, grey, mauve and purple.

$96.93 \pm 0.04 \text{ \AA}$ (100 $\text{\AA}$ Ra OMV-OmpF), respectively (Fig. S3.3).

To investigate if the embedded protein influenced vesicle morphology, we measured the positions of the lipid phosphate groups relative to R_{OMV} (Fig. 3.5 C, D). The incorporation of the OmpF protein does not significantly affect the distribution of LPS phosphate groups in either Re or Ra OMV. The OMV thickness was measured using the RDF of the outer and inner leaflets with respect to the center of mass of the vesicles. The OMV thicknesses from the last 500 ns are 35.88 ± 2.51 (150 $\text{\AA}$ Re OMV-only), 36.18 ± 2.15 (150 $\text{\AA}$ Re OMV-OmpF), 34.97 ± 4.16 (150 $\text{\AA}$ Ra OMV-only), and $34.45 \pm 6.26 \text{ \AA}$ (100 $\text{\AA}$ Ra OMV-OmpF), respectively. The average LPS APL in the OMV systems over the last 500 ns simulations were 179.2 ± 0.014 (150 $\text{\AA}$ Re OMV-only), 179.4 ± 0.015 (150 $\text{\AA}$ Re OMV-OmpF), 200.5 ± 0.018 (150 $\text{\AA}$ Ra OMV-only), and $200.8 \pm 0.029 \text{ \AA}^2$ (100 $\text{\AA}$ Ra OMV-OmpF), respectively, while the APL for the phospholipids in the OMV systems can be found in Table S3.6. The OMV thickness was smaller than the bilayer system (Fig. 3.3) and the area per lipid of LPS was larger than the bilayer system (Table S3.5), indicating that our OMVs were less packed. Given the complexity of the system building and simulation, this could be acceptable, but also suggest that users may need to try a few different set of the number of LPS in the outer leaflet, as it would be difficult for us to come up with a set of general parameters applicable to various systems.

Figure 3.5 | LPS vesicles with and without protein. (A, B) OMVs structures and (C, D) Re and Ra LPS phosphate particle positions. The color scheme in (A) and (B) is the same as in Fig. 3.3, and water beads are omitted for clarity. The phosphate particle positions are shown relative to the center of mass of the vesicle bilayers, and the average R_{OMV} is at position 0. The black line shows data for (C) 150 Å Re OMV-only and (D) 150 Å Ra OMV-only, while the magenta line shows data for (C) 150 Å Re OMV-OmpF and (D) 100 Å Ra OMV-OmpF. Note that there are two phosphate density profiles in (D) Ra OMVs in the outer leaflet, one from lipid A and the other from the core region.

3.3.4. Micelle and random systems

Regarding their immune response, LPS aggregates are considered to be biologically active as monomers[65, 66]. However, little is known about their supramolecular organization. Experimental evidences indicate that LPS molecules can form micelles, lamellar, tubular or even cubic assemblies[66–69]. In this section, we tested Random Builder and Micelle Builder as tools to get the first estimates of how the LPS Martini CG models predict the type of aggregates formed by small numbers of Re and Ra LPS molecules. MD simulations of 5 to 20 LPS molecules at 310 K and 1 bar were performed in order to predict their self-assemblies (see Methods). The results are presented in Fig. 3.6, giving emphasis to the systems containing 20 Ra LPS molecules. Other results, including those obtained with Re LPS molecules, are detailed in Fig. S3.4–S3.6.

The 10 μ s simulations of 20 Ra LPS molecules randomly placed in a water cubic box at 310 K did not yield any form of micelle structure or small bilayer patches. Instead, after 6 μ s, Ra LPS aggregates appear to be trapped in stable elongated structures with an average radius of gyration (R_g) of 30.73 ± 0.03 Å (Fig. 3.6 A, D).

This unusual aggregate indicates more favorable interactions between the oligosaccharide cores than the aliphatic tails (as indicated by the tail-tail/core-core contacts ratio in Fig. 6E). Different counterions (Na^+ or Ca^{2+}) or improvements in the electrostatic interactions (through the usage of polarizable models) were also not able to produce micelle or bilayer-like structures (Fig. 3.6 F). Only an auxiliary 5 μ s MD simulation performed at higher temperature (400 K but the same density of systems simulated at 310 K) was converged to bilayer-like aggregates with an average Rg of 25.60 ± 0.01 Å (Fig. 3.6 A), showing a highly compact sugar head, as already exemplified in the previous bilayer sections.

The usage of Micelle Builder clearly showed the importance of the initial configuration for MD simulations with LPS molecules. The pre-assembled micelle structures at physiological conditions (310 K and 1 bar) and with less simulation time (Fig. 3.6 B) allowed the formation of aggregates similar to those predicted by MD simulations

of random Ra LPS initial configuration at high temperatures. About 1 μ s was sufficient for the convergence of the bilayer-like structure. For MD simulations performed with less than 15 LPS molecules, the aggregates obtained from the simulations of Ra and Re LPS systems from both Micelle and Random Builders are structurally very similar with similar Rg, as shown in Fig. 3.6 G (and also in snapshots of Figures S3.5 and S3.6). For the creation of aggregates with 15 to 44 LPS molecules, we do not recommend the usage of Random Builder. In these cases, Micelle builder proved to be a more suitable tool, including its application for MD simulations of LPS-protein aggregates, as exemplified by the pre-assembled Ra LPS–OmpF micelle (Fig. 3.6 C). In pure LPS systems containing 44 or more LPS molecules, Vesicle Builder or Bilayers Builder should be a reasonable choice, depending on the application. It should be stressed that the LPS aggregates presented here are not necessarily the supramolecular structures expected for specific LPS:water ratio, but an easy route to generate aggregates in specific sizes. Further MD simulations are necessary to understand if the current CG model is able to predict the most thermodynamically stable structure in specific LPS concentration.

3.4. Conclusions

In conclusion, we have demonstrated the utility of the CHARMM GUI Martini-Maker for the automated construction of systems that are of relevance to bacterial membranes containing LPS that are much more complex than the typical phospholipids. While phospholipids are often used to mimic bacterial membranes in computational studies, for molecular dynamics simulations of bacterial membranes to be biologically relevant, models should incorporate the diverse chemical moieties that are present in the native bacterial membranes, as well as accessing timescales that are longer than those practically possible using atomistic-level models. This argues for CG systems that can be easily set up. To this end, web-based user-friendly Martini-Maker is developed to construct flat bilayers, vesicles, micelles and nanodiscs with minimal user inputs. We have shown that the resulting structures are robust and similar compared to their atomistic counterparts, although there is room for improvement (i) by optimizing the Martini parameters for Ca^{2+} and by reducing the overall stickiness of sugar-based molecules as was recently performed for gangliosides[70] and (ii) the Martini-Maker system building parameters to better estimate the number of lipid molecules in Vesicle Builder. Martini-Maker is expected to replace the laborious process of manual system setup of these complex LPS-containing systems, with an easy to use, freely available, online procedure that requires minimal manual intervention. Finally, the current work provides a framework to include all LPS molecules (including O-antigen polysaccharides) available in LPS Modeler in CHARMM-GUI³[71].

3.5. Acknowledgments

The work was supported in part by grants from NSF MCB-1727508, NIH GM087519 and GM103695, XSEDE MCB-070009 (to WI) and an ERC Advanced grant "COMP-

³<http://www.charmm-gui.org/input/lps>

MICR-CROW-MEM" (to SJM).

3.6. Supplementary Information

Table S3.1 | The composition of the bilayer systems.

Systems	Protein	Lipid types (# Lipids)		Ion types (# Ions)			# Water	System size (xyz; Å)	Simulation time (μs)
		Outer	Inner	Na ⁺	Cl ⁻	Ca ²⁺			
OM-only	–	56 RaLPS	144 POPE 8 POPG 8 CDL2	395	35	112	4390	100,100,100	9
OM-OmpA	OmpA	56 RaLPS	144 POPE 8 POPG 8 CDL2	390	29	112	4585	100,100,100	5
OM-OmpF	OmpF	102 Ra LPS	270 POPE 15 POPG 15 CDL2	753	60	204	9455	150,150,100	5.5

3

Table S3.2 | The composition of the nanodisc systems.

Systems	Protein	Lipid types (# Lipids)	Ion types (# Ions)			# Water	System size (xyz; Å)	Simulation time (μs)
		Outer / Innter	Na ⁺	Cl ⁻	Ca ²⁺			
Re LPS + MSP	MSP1D1	37 / 37 Re LPS	227	227	227	17321	140, 140, 140	1
	MSP1D1	37 / 37 Re LPS	681	227	0	17321	140, 140, 140	
Ra LPS + MSP	MSP1D1	33 / 33 Ra LPS	297	297	335	22466	150, 150, 150	
	MSP1D1	33 / 33 Ra LPS	967	297	0	22466	150, 150, 150	

Table S3.3 | The composition of the vesicle systems.

Systems	Protein	Lipid types (# Lipids)		Ion types (# Ions)			# Water	System size (xyz; Å)	Simulation time (μs)
		Outer	Inner	Na ⁺	Cl ⁻	Ca ²⁺			
150-Å Re OMV-only	-	2076 Re LPS	3006 POPE 167 POPG 167 CDL2	11689	7036	4152	627652	444, 444, 444	9
150-Å Re OMV-OmpF	OmpF	2076 Re LPS	3006 POPE 167 POPG 167 CDL2	5436	9051	8304	623277	444, 444, 444	5
150-Å Ra OMV-only	-	1844 Ra LPS	3006 POPE 167 POPG 167 CDL2	21382	9817	3688	831366	487, 487, 487	5.5
100-Å Ra OMV-OmpF	OmpF	919 Ra LPS	1273 POPE 70 POPG 70 CDL2	2492	7760	7352	421440	387, 387, 387	10

Figure S3.1 | APL profiles for the bilayer systems. Time-series of area per lipid (APL) of the OM systems. A larger deviation is observed for OM-OmpF due to instability in the analysis program with regard to Voronoi tessellation on frames with multiple beads on the similar x,y coordinates (z is dumped). A skip of 10 was used to reduce the chance of hitting such a frame. Even in this case the average could be defined with a good certainty.

Table S3.4 | The composition of micelle and random systems using standard (W) or polarizable (PW) water models.

Systems	Protein	# Lipids	Ion types (# Ions)			# Water	System size (Å)	Simulation time (μs)
			Na ⁺	Cl ⁻	Ca ²⁺			
Micelle + Ra LPS	—	5	198	168	10	15305 W	124	10
	—	10	304	244	20	22378 W	141	10
	—	15	357	267	30	24477 W	145	10
	—	20	395	275	40	25157 W	147	10
	—	20	475	275	—	25117 W	147	10
	—	20	421	301	40	27383 PW	149	10
	OmpF	40	835	583	80	54062 W	191	10
Micelle + Re LPS	—	5	84	74	10	6692 W	95	10
	—	10	124	104	20	9512 W	106	10
	—	15	135	105	30	9573 W	107	10
	—	20	162	122	40	11156 W	113	10
Random + Ra LPS	—	5	278	248	10	22309 W	140	10
	—	10	308	248	20	21951 W	140	10
	—	15	338	248	30	21500 W	140	10
	—	20	368	248	40	21106 W	140	10
	—	20	448	248	—	21066 W	140	10
	—	20	368	248	40	20993 PW	140	10
	OmpF	40	773	521	80	44135 W	180	10
Random + Re LPS	—	5	131	121	10	10753 W	110	10
	—	10	141	121	20	10471 W	110	10
	—	15	151	121	30	10248 W	110	10
	—	20	161	121	40	9869 W	110	10

Table S3.5 | APL and thickness for the bilayer systems. The average and standard error of mean (SEM) of the APL and bilayer thickness over the last microsecond of the 5 μ s Bilayer Builder production runs. Even though the SEM is pretty low for all systems and lipids, the insertion of a protein causes a systematic error in the Voronoi procedure at the protein lipid interphase.

OM-only	Parameter	RAMP ($\text{\AA}^2$)	POPE ($\text{\AA}^2$)	POPG ($\text{\AA}^2$)	CDL2 ($\text{\AA}^2$)	Bilayer Thickness ($\text{\AA}$)
	Mean	173.33	58.27	60.69	103.87	37.61
	SEM	0.22	0.31	0.11	0.53	0.15
OM-OmpA	Parameter	RAMP ($\text{\AA}^2$)	POPE ($\text{\AA}^2$)	POPG ($\text{\AA}^2$)	CDL2 ($\text{\AA}^2$)	Bilayer Thickness ($\text{\AA}$)
	Mean	170.61	57.72	59.95	103.18	37.25
	SEM	1.31	0.14	0.72	0.44	0.20
OM-OmpF	Parameter	RAMP ($\text{\AA}^2$)	POPE ($\text{\AA}^2$)	POPG ($\text{\AA}^2$)	CDL2 ($\text{\AA}^2$)	Bilayer Thickness ($\text{\AA}$)
	Mean	176.93	59.50	62.59	104.82	37.18
	SEM	0.17	0.34	0.64	2.0	0.12

Table S3.6 | The average and the standard error of mean (SEM) of OMV area per lipid over the last 500 ns of total simulation time.

150- $\text{\AA}$ Re OMV-only	Parameter	REMP ($\text{\AA}^2$)	POPE ($\text{\AA}^2$)	POPG ($\text{\AA}^2$)	CDL2 ($\text{\AA}^2$)
	Mean	179.2	61.7	63.4	66.9
	SEM	0.014	0.01	0.012	0.01
150- $\text{\AA}$ Re OMV-OmpF	Parameter	REMP ($\text{\AA}^2$)	POPE ($\text{\AA}^2$)	POPG ($\text{\AA}^2$)	CDL2 ($\text{\AA}^2$)
	Mean	179.4	62.1	64.3	65.6
	SEM	0.015	0.002	0.021	0.011
150- $\text{\AA}$ Ra OMV-only	Parameter	RAMP ($\text{\AA}^2$)	POPE ($\text{\AA}^2$)	POPG ($\text{\AA}^2$)	CDL2 ($\text{\AA}^2$)
	Mean	200.5	60.7	62.3	68.7
	SEM	0.018	0.001	0.012	0.010
100- $\text{\AA}$ Ra OMV-OmpF	Parameter	RAMP ($\text{\AA}^2$)	POPE ($\text{\AA}^2$)	POPG ($\text{\AA}^2$)	CDL2 ($\text{\AA}^2$)
	Mean	200.8	61.1	59.7	66.1
	SEM	0.029	0.004	0.017	0.016

Figure S3.2 | RMSD of nanodisc systems. Time-series of the root-mean-square deviation(RMSD) of MSP1D1 proteins in the Ra LPS nanodiscs neutralized with (A) Na⁺ or (B) Ca²⁺ ions and Re LPS nanodiscs neutralized with (C) Na⁺ or (D) Ca²⁺ ions.

Figure S3.3 | Radii for vesicle systems. Time-series of the OMV radii calculated as the radial distance between the OMV center of and the center of mass of the encapsulating membranes for (A) the total simulation time and (B) the last 500 ns simulation time

Figure S3.4 | Time-series of the radius of gyration of the LPS molecules in various micelle and random systems.

Figure S3.5 | Initial structure and the last snapshot of MD simulations performed with Ra LPS micelle and random systems (5, 10, and 15 LPS molecules).

Figure S3.6 | Initial structure and the last snapshot of MD simulations performed with Re LPS micelle and random systems (5, 10, 15, and 20 LPS molecules.

3.7. References

- [1] P.-C. Hsu, B. M. H. Bruininks, D. Jefferies, P. C. T. de Souza, J. Lee, D. S. Patel, S. J. Marrink, Y. Qi, S. Khalid, and W. Im, *CHARMM-GUI martini maker for modeling and simulation of complex bacterial membranes with lipopolysaccharides*, *Journal of Computational Chemistry* **38**, 2354 (2017).
- [2] T. Mori, N. Miyashita, W. Im, M. Feig, and Y. Sugita, *Molecular dynamics simulations of biological membranes and membrane proteins using enhanced conformational sampling algorithms*, *Biochimica et Biophysica Acta (BBA) - Biomembranes* **1858**, 1635 (2016).
- [3] H. I. Ingólfsson, C. Arnarez, X. Periole, and S. J. Marrink, *Computational 'microscopy' of cellular membranes*, *Journal of Cell Science* **129**, 257 (2016).
- [4] D. Jefferies, P.-C. Hsu, and S. Khalid, *Through the lipopolysaccharide glass: A potent antimicrobial peptide induces phase changes in membranes*, *Biochemistry* **56**, 1672 (2017).
- [5] K. R. Pothula, C. J. Solano, and U. Kleinekathöfer, *Simulations of outer membrane channels and their permeability*, *Biochimica et Biophysica Acta (BBA) - Biomembranes* **1858**, 1760 (2016).
- [6] S. J. Marrink and D. P. Tieleman, *Perspective on the martini model*, *Chemical Society Reviews* **42**, 6801 (2013).
- [7] S. J. Marrink, H. J. Risselada, S. Yefimov, D. P. Tieleman, and A. H. de Vries, *The MARTINI force field: coarse grained model for biomolecular simulations*, *The Journal of Physical Chemistry B* **111**, 7812 (2007).
- [8] M. Lelimosin, V. Limongelli, and M. S. P. Sansom, *Conformational changes in the epidermal growth factor receptor: Role of the transmembrane domain investigated by coarse-grained MetaDynamics free energy calculations*, *Journal of the American Chemical Society* **138**, 10611 (2016).
- [9] M. N. Melo, C. Arnarez, H. Sikkema, N. Kumar, M. Walko, H. J. C. Berendsen, A. Kocer, S. J. Marrink, and H. I. Ingólfsson, *High-throughput simulations reveal membrane-mediated effects of alcohols on MscL gating*, *Journal of the American Chemical Society* **139**, 2664 (2017).
- [10] F. J. V. Eerden, M. N. Melo, P. W. J. M. Frederix, X. Periole, and S. J. Marrink, *Exchange pathways of plastoquinone and plastoquinol in the photosystem II complex*, *Nature Communications* **8** (2017), 10.1038/ncomms15214.
- [11] Y. Knirel, *Bacterial lipopolysaccharides : structure, chemical synthesis, biogenesis and interaction with host cells* (Springer-Verlag, Wien, 2011).
- [12] M. P. Bos, V. Robert, and J. Tommassen, *Biogenesis of the gram-negative bacterial outer membrane*, *Annual Review of Microbiology* **61**, 191 (2007).

- [13] N. Ruiz, D. Kahne, and T. J. Silhavy, *Advances in understanding bacterial outer-membrane biogenesis*, *Nature Reviews Microbiology* **4**, 57 (2006).
- [14] J. Parkin, M. Chavent, and S. Khalid, *Molecular simulations of gram-negative bacterial membranes: A vignette of some recent successes*, *Biophysical Journal* **109**, 461 (2015).
- [15] H. Nikaido, *Role of permeability barriers in resistance to β -lactam antibiotics*, *Pharmacology & Therapeutics* **27**, 197 (1985).
- [16] H. Nikaido, *Molecular basis of bacterial outer membrane permeability revisited*, *Microbiology and Molecular Biology Reviews* **67**, 593 (2003).
- [17] D. S. Patel, S. Re, E. L. Wu, Y. Qi, P. E. Klebba, G. Widmalm, M. S. Yeom, Y. Sugita, and W. Im, *Dynamics and interactions of OmpF and LPS: Influence on pore accessibility and ion permeability*, *Biophysical Journal* **110**, 930 (2016).
- [18] T. J. Piggot, D. A. Holdbrook, and S. Khalid, *Electroporation of the e. coli and s. aureus membranes: Molecular dynamics simulations of complex bacterial membranes*, *The Journal of Physical Chemistry B* **115**, 13381 (2011).
- [19] E. L. Wu, P. J. Fleming, M. S. Yeom, G. Widmalm, J. B. Klauda, K. G. Fleming, and W. Im, *E. coli outer membrane and interactions with OmpLA*, *Biophysical Journal* **106**, 2493 (2014).
- [20] S. Khalid, N. A. Berglund, D. A. Holdbrook, Y. M. Leung, and J. Parkin, *The membranes of gram-negative bacteria: progress in molecular modelling and simulation*, *Biochemical Society Transactions* **43**, 162 (2015).
- [21] D. S. Patel, Y. Qi, and W. Im, *Modeling and simulation of bacterial outer membranes and interactions with membrane proteins*, *Current Opinion in Structural Biology* **43**, 131 (2017).
- [22] A. Pavlova, H. Hwang, K. Lundquist, C. Balusek, and J. C. Gumbart, *Living on the edge: Simulations of bacterial outer-membrane proteins*, *Biochimica et Biophysica Acta (BBA) - Biomembranes* **1858**, 1753 (2016).
- [23] S. Kim, D. S. Patel, S. Park, J. Slusky, J. B. Klauda, G. Widmalm, and W. Im, *Bilayer properties of lipid a from various gram-negative bacteria*, *Biophysical Journal* **111**, 1750 (2016).
- [24] P. J. Fleming, D. S. Patel, E. L. Wu, Y. Qi, M. S. Yeom, M. C. Sousa, K. G. Fleming, and W. Im, *BamA POTRA domain interacts with a native lipid membrane surface*, *Biophysical Journal* **110**, 2698 (2016).
- [25] D. A. Holdbrook, T. J. Piggot, M. S. Sansom, and S. Khalid, *Stability and membrane interactions of an autotransport protein: MD simulations of the hia translocator domain in a complex membrane environment*, *Biochimica et Biophysica Acta (BBA) - Biomembranes* **1828**, 715 (2013).

- [26] J. Lee, D. S. Patel, I. Kucharska, L. K. Tamm, and W. Im, *Refinement of OprH-LPS interactions by molecular simulations*, *Biophysical Journal* **112**, 346 (2017).
- [27] T. J. Piggot, D. A. Holdbrook, and S. Khalid, *Conformational dynamics and membrane interactions of the e. coli outer membrane protein FecA: A molecular dynamics simulation study*, *Biochimica et Biophysica Acta (BBA) - Biomembranes* **1828**, 284 (2013).
- [28] S. Jo, E. L. Wu, D. Stuhlsatz, J. B. Klauda, A. D. MacKerell, G. Widmalm, and W. Im, *Lipopolysaccharide membrane building and simulation*, in *Methods in Molecular Biology* (Springer New York, 2015) pp. 391–406.
- [29] H. Ma, F. J. Irudayanathan, W. Jiang, and S. Nangia, *Simulating gram-negative bacterial outer membrane: A coarse grain model*, *The Journal of Physical Chemistry B* **119**, 14668 (2015).
- [30] B. V. Oosten and T. A. Harroun, *A MARTINI extension for pseudomonas aeruginosa PAO1 lipopolysaccharide*, *Journal of Molecular Graphics and Modelling* **63**, 125 (2016).
- [31] S. Jo, T. Kim, V. G. Iyer, and W. Im, *CHARMM-GUI: A web-based graphical user interface for CHARMM*, *Journal of Computational Chemistry* **29**, 1859 (2008).
- [32] E. L. Wu, X. Cheng, S. Jo, H. Rui, K. C. Song, E. M. Dávila-Contreras, Y. Qi, J. Lee, V. Monje-Galvan, R. M. Venable, J. B. Klauda, and W. Im, *CHARMM-GUI Membrane buildertoward realistic biological membrane simulations*, *Journal of Computational Chemistry* **35**, 1997 (2014).
- [33] S. Jo, J. B. Lim, J. B. Klauda, and W. Im, *CHARMM-GUI membrane builder for mixed bilayers and its application to yeast membranes*, *Biophysical Journal* **97**, 50 (2009).
- [34] X. Cheng, S. Jo, H. S. Lee, J. B. Klauda, and W. Im, *CHARMM-GUI micelle builder for pure/mixed micelle and protein/micelle complex systems*, *Journal of Chemical Information and Modeling* **53**, 2171 (2013).
- [35] Y. Qi, H. I. Ingólfsson, X. Cheng, J. Lee, S. J. Marrink, and W. Im, *CHARMM-GUI martini maker for coarse-grained simulations with the martini force field*, *Journal of Chemical Theory and Computation* **11**, 4486 (2015).
- [36] P.-C. Hsu, D. Jefferies, and S. Khalid, *Molecular dynamics simulations predict the pathways via which pristine fullerenes penetrate bacterial membranes*, *The Journal of Physical Chemistry B* **120**, 11170 (2016).
- [37] J. A. Graham, J. W. Essex, and S. Khalid, *PyCGTOOL: Automated generation of coarse-grained molecular dynamics models from atomistic trajectories*, *Journal of Chemical Information and Modeling* **57**, 650 (2017).

- [38] R. T. Coughlin, S. Tonsager, and E. J. McGroarty, *Quantitation of metal cations bound to membranes and extracted lipopolysaccharide of escherichia coli*, *Biochemistry* **22**, 2002 (1983).
- [39] M. Schindler and M. J. Osborn, *Interaction of divalent cations and polymyxin b with lipopolysaccharide*, *Biochemistry* **18**, 4425 (1979).
- [40] L. van Alphen, A. Verkleij, J. Leunissen-Bijvelt, and B. Lugtenberg, *Architecture of the outer membrane of escherichia coli III. protein-lipopolysaccharide complexes in intramembraneous particles*, *Journal of Bacteriology* **134**, 1089 (1978).
- [41] E. L. Wu, O. Engström, S. Jo, D. Stuhlsatz, M. S. Yeom, J. B. Klauda, G. Widmalm, and W. Im, *Molecular dynamics and NMR spectroscopy studies of e. coli lipopolysaccharide structure and dynamics*, *Biophysical Journal* **105**, 1444 (2013).
- [42] T. A. Wassenaar, H. I. Ingólfsson, R. A. Böckmann, D. P. Tieleman, and S. J. Marrink, *Computational lipidomics with insane: A versatile tool for generating custom membranes for molecular simulations*, *Journal of Chemical Theory and Computation* **11**, 2144 (2015).
- [43] M. Dahlberg and A. Maliniak, *Mechanical properties of coarse-grained bilayers formed by cardiolipin and zwitterionic lipids*, *Journal of Chemical Theory and Computation* **6**, 1638 (2010).
- [44] M. Pannuzzo, D. H. D. Jong, A. Raudino, and S. J. Marrink, *Simulation of polyethylene glycol and calcium-mediated membrane fusion*, *The Journal of Chemical Physics* **140**, 124905 (2014).
- [45] S. J. Marrink, A. H. de Vries, and A. E. Mark, *Coarse grained model for semi-quantitative lipid simulations*, *The Journal of Physical Chemistry B* **108**, 750 (2004).
- [46] L. Monticelli, S. K. Kandasamy, X. Periole, R. G. Larson, D. P. Tieleman, and S. J. Marrink, *The MARTINI coarse-grained force field: Extension to proteins*, *Journal of Chemical Theory and Computation* **4**, 819 (2008).
- [47] X. Periole, M. Cavalli, S. J. Marrink, and M. A. Ceruso, *Combining an elastic network with a coarse-grained molecular force field: Structure, dynamics, and intermolecular recognition*, *Journal of Chemical Theory and Computation* **5**, 2531 (2009).
- [48] D. H. de Jong, S. Baoukina, H. I. Ingólfsson, and S. J. Marrink, *Martini straight: Boosting performance using a shorter cutoff and GPUs*, *Computer Physics Communications* **199**, 1 (2016).
- [49] M. J. Abraham, T. Murtola, R. Schulz, S. Páll, J. C. Smith, B. Hess, and E. Lindahl, *GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers*, *SoftwareX* **1-2**, 19 (2015).

- [50] E. Lugtenberg and R. Peters, *Distribution of lipids in cytoplasmic and outer membranes of escherichia coli k12*, Biochimica et Biophysica Acta (BBA) - Lipids and Lipid Metabolism **441**, 38 (1976).
- [51] A. Pautsch and G. E. Schulz, *High-resolution structure of the OmpA membrane domain*, Journal of Molecular Biology **298**, 273 (2000).
- [52] R. G. Efremov and L. A. Sazanov, *Structure of escherichia coli OmpF porin from lipidic mesophase*, Journal of Structural Biology **178**, 311 (2012).
- [53] M. P. Molloy, B. R. Herbert, M. B. Slade, T. Rabilloud, A. S. Nouwens, K. L. Williams, and A. A. Gooley, *Proteomic analysis of the escherichia coli outer membrane*, European Journal of Biochemistry **267**, 2871 (2000).
- [54] J. Murrell, *Physiology of the bacterial cell — a molecular approach*, Trends in Genetics **7**, 341 (1991).
- [55] M. A. Lomize, A. L. Lomize, I. D. Pogozheva, and H. I. Mosberg, *OPM: Orientations of proteins in membranes database*, Bioinformatics **22**, 623 (2006).
- [56] N. Michaud-Agrawal, E. J. Denning, T. B. Woolf, and O. Beckstein, *MDAnalysis: A toolkit for the analysis of molecular dynamics simulations*, Journal of Computational Chemistry **32**, 2319 (2011).
- [57] W. Humphrey, A. Dalke, and K. Schulten, *VMD: Visual molecular dynamics*, Journal of Molecular Graphics **14**, 33 (1996).
- [58] H. J. Risselada, A. E. Mark, and S. J. Marrink, *Application of mean field boundary potentials in simulations of lipid vesicles*, The Journal of Physical Chemistry B **112**, 7438 (2008).
- [59] S. Buchoux, *FATSLIM: a fast and robust software to analyze MD simulations of membranes*, Bioinformatics **33**, 133 (2016).
- [60] D. A. Holdbrook, R. G. Huber, T. J. Piggot, P. J. Bond, and S. Khalid, *Dynamics of crowded vesicles: Local and global responses to membrane composition*, PLOS ONE **11**, e0156963 (2016).
- [61] S. O. Yesylevskyy, L. V. Schäfer, D. Sengupta, and S. J. Marrink, *Polarizable water model for the coarse-grained MARTINI force field*, PLoS Computational Biology **6**, e1000810 (2010).
- [62] S. Bibow, Y. Polyhach, C. Eichmann, C. N. Chi, J. Kowal, S. Albiez, R. A. McLeod, H. Stahlberg, G. Jeschke, P. Güntert, and R. Riek, *Solution structure of discoidal high-density lipoprotein particles with a shortened apolipoprotein a-i*, Nature Structural & Molecular Biology **24**, 187 (2016).
- [63] I. G. Denisov and S. G. Sligar, *Nanodiscs in membrane biochemistry and biophysics*, Chemical Reviews **117**, 4669 (2017).

- [64] I. G. Denisov and S. G. Sligar, *Nanodiscs for structural and functional studies of membrane proteins*, *Nature Structural & Molecular Biology* **23**, 481 (2016).
- [65] M. Mueller, B. Lindner, S. Kusumoto, K. Fukase, A. B. Schromm, and U. Seydel, *Aggregates are the biologically active units of endotoxin*, *Journal of Biological Chemistry* **279**, 26307 (2004).
- [66] H. Sasaki and S. H. White, *Aggregation behavior of an ultra-pure lipopolysaccharide that stimulates TLR-4 receptors*, *Biophysical Journal* **95**, 986 (2008).
- [67] K. BRANDENBURG and U. SEYDEL, *Investigation into the fluidity of lipopolysaccharide and free lipid a membrane systems by fourier-transform infrared spectroscopy and differential scanning calorimetry*, *European Journal of Biochemistry* **191**, 229 (1990).
- [68] N. C. Santos, A. C. Silva, M. A. R. B. Castanho, J. Martins-Silva, and C. Saldanha, *Evaluation of lipopolysaccharide aggregation by light scattering spectroscopy*, *ChemBioChem* **4**, 96 (2003).
- [69] S. G. Wilkinson, *Bacterial lipopolysaccharides – themes and variations*, *Progress in Lipid Research* **35**, 283 (1996).
- [70] R.-X. Gu, H. I. Ingólfsson, A. H. de Vries, S. J. Marrink, and D. P. Tieleman, *Ganglioside-lipid and ganglioside-protein interactions revealed by coarse-grained and atomistic molecular dynamics simulations*, *The Journal of Physical Chemistry B* **121**, 3262 (2016).
- [71] S. Jo, X. Cheng, J. Lee, S. Kim, S.-J. Park, D. S. Patel, A. H. Beaven, K. I. Lee, H. Rui, S. Park, H. S. Lee, B. Roux, A. D. MacKerell, J. B. Klauda, Y. Qi, and W. Im, *CHARMM-GUI 10 years for biomolecular modeling and simulation*, *Journal of Computational Chemistry* **38**, 1114 (2016).

4

A molecular view on the escape of lipoplexed DNA from the endosome

Problems are solved by the people closest to a solution.

Paulo C.T. de Souza

The use of non-viral vectors for in vivo gene therapy could drastically increase safety, whilst reducing the cost of preparing the vectors. A promising approach to non-viral vectors makes use of DNA/cationic liposome complexes (lipoplexes) to deliver the genetic material. Here we use coarse-grained molecular dynamics simulations to investigate the molecular mechanism underlying efficient DNA transfer from lipoplexes. Our computational fusion experiments of lipoplexes with endosomal membrane models show two distinct modes of transfection: parallel and perpendicular. In the parallel fusion pathway, DNA aligns with the membrane surface, showing very quick release of genetic material shortly after the initial fusion pore is formed. The perpendicular pathway also leads to transfection, but release is slower. We further show that the composition and size of the lipoplex, as well as the lipid composition of the endosomal membrane, have a significant impact on fusion efficiency in our models.

4.1. Introduction

Gene therapy is a promising technique with a wide applicability. The first clinical trials with gene therapy started in the early 90's, and the first approved therapy being introduced in Europe in 2012[2–4]. Even though the concept of gene therapy has been around for a while, the problem remains to target and enter the right cells, without being toxic to the rest of the organism.

Most higher organisms have evolved quite stringent measures to block uptake of DNA from their surroundings, preventing excessive genetic instability. Naked DNA gets quickly degraded in our body by exonucleases, whereas using viruses as vectors can lead to a strong acquired immune response[5]. Therefore, 'new' non-viral based vectors are being developed. Non-viral vectors have two advantages, first, they do not trigger a specific immune response, and second, they are potentially much cheaper than viral vectors. Most non-viral vectors use cationic lipids or polymers for complexation with the negatively charged DNA, concealing the genetic material from degradation. The major downside to many of these non-viral based methods, is that their transfection rates in vivo are rather low[6–8]. Moreover, lipoplexes are potentially toxic especially if they are highly positively charged[9].

In this work we focus on cationic lipid-DNA based vectors, called lipoplexes. Lipoplexes consist of (cationic) lipids whose role is twofold. One, they shield the genetic material from degradation, and two, the lipids support transfection[10–12]. Upon exposure of the vector to cells, the cells incorporate the lipoplex by means of endocytosis, causing the complex to subsequently reside in the early and late endosome, and finally to be degraded in the lysosome. It is therefore important for the genetic material to escape the endosome before it gets degraded[13–15]. This escape from the endosome is one of the least understood and inefficient steps in current lipoplex or polymer mediated gene therapy and improving endosomal escape might improve transfection efficiency drastically[16].

Due to the microscopic scale and dynamic nature of lipoplex-membrane fusion, it has proven very difficult by experimental means to relate physico-chemical properties of the vector to transfection efficiency. Molecular dynamics (MD) simulations provide an alternative tool to study molecular processes in atomic, or near atomic detail[17]. In particular coarse-grained (CG) models such as Martini[18, 19] have proven popular, trading some of the atomic detail for a computational speedup and enabling direct simulation of membrane fusion[20–22]. Here we use the Martini model to simulate the fusion between nanoscale lipoplexes and endosomal model membranes and vesicles. We are able to resolve the molecular details of the fusion process, including the full release of the lipoplex cargo, short fragments of double-stranded DNA (dsDNA), across the endosomal membrane.

4.2. Results and discussion

4.2.1. Construction and validation of the lipoplex model

To investigate the escape of genetic material from a lipoplex inside a model endosome, we first constructed and validated a small lipoplex (18 nm in diameter, 4 fragments of 24 bp dsDNA; Fig. 2.2). Our lipoplex formulation consists of 1,2-dioleoyl-

3-trimethylammonium-propane (DOTAP) as the cationic lipid to bind the dsDNA, and 1,2-dioleoyl-sn-glycero-3-phosphoethanolamine (DOPE) as the helper lipid, at a 1:4 DOTAP/DOPE ratio. This complex is known to form the inverted hexagonal phase (H_{II}) in complex with dsDNA at this ratio[10, 23, 24]. Starting from a multilamellar initial configuration, with the dsDNA sandwiched in between the lamella, we observe spontaneous formation of the H_{II} phase (Fig. 2.2 1), consistent with the experimental behavior. After removing periodic boundary conditions and coating the aggregate with an additional layer of DOTAP and DOPE, we arrive at our small lipoplex model which proves stable during a 10 μ s simulation (Fig. 4.1 I; Fig. 2.2; Fig. S4.1). To validate our model, we compared the dsDNA spacing inside the solvated lipoplex with the available SAXS data from literature. This showed that the liquid crystal lattice of the inverted hexagonal phase in our solvated CG lipoplex was very close to the experimental data (6.0 ± 0.1 nm compared to 6.1 nm for SAXS)[23].

For the endosome we made a crude symmetrical membrane model containing a mixture of PC and PS lipids at a 4:1 ratio. Clearly, real endosomal membranes are much more complex concerning lipid composition[25]. However, to a large extent, the exact composition is unknown (especially concerning asymmetry) and varies between endosomal stage and cell type. Given the large complexity of the simulated systems, we decided to keep the endosomal membrane composition as simple as possible, with PC as main phospholipid. PS was added as it was hypothesized before that anionic lipids could play an important role in the transfection mechanism[26]. Both PC and PS lipids had a mixture of diC14:0 (dimyristoyl), diC14:1,9 c (dimyristoleoyl) and C14:0-C14:1,9 c (myristoyl-myristoleoyl) tails at a 1:1:2 ratio (Fig. 4.2). The relatively short lipid tails were chosen because pore formation for longer tailed lipids have been shown to be too energetically unfavourable for the CG Martini lipids compared to their all-atom counterparts[27, 28].

4.2.2. Two distinct mechanisms leading to gene transfection

We first performed completely unbiased fusion experiments by placing the solvated lipoplex in solution above a hydrated bilayer. To monitor fusion we used: (i) lipid mixing, (ii) DNA-water contacts, and (iii) visual inspection. At the 10 μ s time scale using unbiased simulations, we did not obtain sufficient dehydration of the lipoplex membrane interface for fusion to proceed (data not shown). Dehydration of the membrane-membrane interface is a known kinetic barrier for membrane fusion[30], therefore we changed the initial set-up of the fusion experiment and started from a dehydrated state, with one of the corners of the lipoplex pointing toward the endosomal membrane. The dehydrated systems either remained in their dehydrated state, resulting eventually in the formation of a fusion stalk, or the membrane would rehydrate – from which dehydration was never observed. Since the formation of the fusion stalk has been described at many levels of detail[31, 32], and is widely accepted as an essential step in any fusion pathway, we decided to start our further transfection experiments from the stalk state (4.1 II), allowing for a more systematic approach.

We then performed five replica simulations, each ten microseconds long, starting

Figure 4.1 | Schematic view of the lipoplex-membrane transfection pathways. Unadhered lipoplex containing four dsDNAs sits above the endosomal bilayer (I). After initial stalk formation (II), a wide hemifusion diaphragm is formed (III). A pore is formed in one of the channels containing the DNA (IVa, IVb). The angle of the DNA with respect to the average bilayer normal can be either perpendicular or parallel, resulting in subsequently zipper (IVa) or ejection (IVb) like release of the DNA (V). The pathways indicated are based on 25 independent fusion experiments, each 10 μ s long, leading to 34 dsDNA transfection events. The time scales indicated are typical for successful transfection events.

MEMBRANE/LIPOPLEX	LIPIDS HEADS	LIPIDS TAILS
Short Lipids	PC:4, PS:1	DY:1, LY:2, DL:1
More Unsaturated	PC:4, PS:1	DLi:1, DO:2, DP:1
Most Unsaturated*	PC:4, PS:1	DA:1, DO:2, DP:1
DPTAP	PE:4, TAP:1	DO:4, DP:1
DOTAP	PE:4, TAP:1	DO:5
DLITAP	PE:4, TAP:1	DO:4, DLi:1

*Phase separated (Fig. 2-figure supplement 1)

Figure 4.2 | Membrane and lipoplex compositions. The membrane compositions for the fusion experiments, and the lipoplex formulations are indicated in molar ratios. All PC/PS heads were combined with all tails, resulting in six different lipids per membrane composition (first 3). For the lipoplex formulations all PE was linked to DO and all TAP to either DP/DO/DLi (last 3). The nomenclature follows the default Martini lipids abbreviations as described by T.A. Wassenaar et al.[29]. An example of each tail and headgroup is displayed under the table (the tails are in the order of occurrence in their name).

from the stalk state. In each case, the system evolved to fusion of the lipoplex with the endosomal membrane, with the cargo being delivered across the target membrane (Fig. 4.1; Video 1[1]). The initial stalk rapidly decays (on a time scale of 10–100 ns), expanding radially, leading to a so-called hemifusion diaphragm (HD) state[33] in which the inner lipoplex coat and the external endosomal membrane are in direct contact, exchanging lipids (Fig. 4.1 III; Figure 3—figure supplements 1–5[1])¹. Transfection subsequently proceeds via two different pathways, depending on the orientation of dsDNA either parallel or perpendicular with respect to the membrane normal (Fig. 4.1 IV; Figure 3—figure supplements 1–5[1]). For perpendicular transfection to the bilayer normal, an initial pore is formed at the rim of the hemifusion diaphragm, this causes rapid unzipping of lipids away from one dsDNA fragment. This unzipping results in a pore as large as the dsDNA fragment effectively

¹The obtained graphs were added to the original publication but were meant for digital display. Therefore they are not included in this thesis.

transfecting the DNA by moving the lipids (Fig. 4.1 IVa), indicated by a rapid increase in hydration of the dsDNA (Figure 3—figure supplements 1–5[1]). For parallel transfection, a pore is formed at the place where the nearest dsDNA fragment is pointing at the HD. Pore formation is then followed by ‘ejection’ of this dsDNA fragment, resembling the unloading of a syringe, by moving both the dsDNA and lipids (Fig. 4.1 IVb). Visual inspection of the unloading process (Video 1[1]) reveals that the dsDNA moves out of the channel concomitantly to the merging of the surrounding lipids with the endosomal membrane. This suggests that the release of curvature stress of the lipoplex (as the lipoplex channel lipid composition gets mixed with the endosomal lipids, the H_{II} phase becomes destabilized) acts as the driving force pushing the DNA into the cytosol.

4

In both pathways, after transfection of one dsDNA fragment, fusion can continue with another pore formation step to release the next fragment. At this point, either the lipoplex is already fully destabilized and the remaining cargo gets quickly transfected as well, or fusion halts in a prolonged hemifused state and some of the dsDNA remains trapped. In total 16 of the possible 20 dsDNA fragments (four fragments, five replicates) were successfully transfected in this setup. If complete fusion was achieved (Fig. 4.1 V), all dsDNA were at the opposing side of the bilayer. No leakage of dsDNA at the endosomal lumen was ever observed. Once transfected, although most of the dsDNA fragments still associated with the membrane (potentially stabilized by some of the cationic lipids from the lipoplex that come along), they were no longer enveloped by lipoplex components. The decomplexation of the dsDNA correlates well with the behaviour of lipoplex fusion *in vitro* by Rehman et al.[13], where they describe that the DNA after transfection is no longer associated with lipoplex components.

To quantify the overall transfection events further, we monitored the relative angle of the dsDNA fragments with respect to the membrane normal (Figure 3—figure supplements 1–5[1]). In all cases, the first transfected fragment of dsDNA was either perpendicular to the bilayer normal, or close to the starting orientation (90° and 45° respectively). Remarkably, in some of the simulations the lipoplex reoriented itself such that the following transfection event(s) took place with the dsDNA in the parallel orientation, that is switching between the perpendicular and parallel pathway. This remarkable rearrangement might find its origin in the fact that the lipoplex core is suspended similar to a ball bearing, characterized by a low friction of the DNA/lipid core with respect to the coating layer. The removal of lipoplex material during fusion, which typically takes place on one side of the hemifusion diaphragm, might result in a torque causing the observed reorientation.

4.2.3. DxTAP unsaturation is mandatory for efficient transfection

Having established two major fusion pathways leading to successful DNA transfection, next we consider the role of lipid composition in this process. Experimental studies reveal that lipid tail saturation and length, as well as the chemical composition of the headgroup of the lipoplex play an important role[34–36]. In this paragraph we aim to unravel the molecular influence of lipid tail saturation on lipoplex fusion

and transfection efficiency. To this end, we constructed two additional versions of our lipoplex, replacing the mono-unsaturated DOTAP either by its fully saturated counterpart DPTAP (i.e., replacing the oleoyl tails by palmitoyl tails), or by a double unsaturated analogue DLI_{TAP} (linoleoyl tails), whilst maintaining the same 1:4 Dx-TAP:DOPE ratio and target endosomal membrane composition (Fig. 4.2, short lipids). The fusion experiments were set up in the same manner as before with five repeats per condition, each spanning ten microseconds.

For the DPTAP based lipoplex we never observed transfection, even though the stalk was maintained. The internal structure of the lipoplex was found to be somewhat unstable, with some of the channels fusing with each other and losing their classical H_{II} hexagonal packing. Nevertheless, no DNA was lost from the complex. The orientation of the dsDNA was mainly parallel with respect to the bilayer, except for short periods of internal reconfiguration upon loss of the hexagonal unit cell (Figure 3—figure supplements 1–5[1]). The DLI_{TAP} lipoplex, on the other hand, underwent fusion and showed cargo transfection in four out of five replicate simulations. The total amount of dsDNA fragments translocated amounted to 10 out of 20, somewhat less compared to the DOTAP lipoplex (Fig. 4.3). Similar to DOTAP, both parallel and perpendicular fusion pathways were observed, as quantified by the dsDNA orientation analysis (Figure 3—figure supplements 1–5[1]). Based on these results, it appears that unsaturated lipids complexing the dsDNA are required for successful fusion and transfection. This agrees with fusion experiments between bilayers/vesicles, from which it has been suggested that more negatively curved lipids and or lipids with a decreased bending modulus favor formation of the fusion stalk and subsequent hemifusion diaphragm[20, 32, 37–39]. Since the bending modulus tends to decrease with the amount of unsaturations in the tails[40, 41], one could naively anticipate an even further increased fusion efficiency for DLI_{TAP} compared to DOTAP, contrary to our findings. In our understanding, this can be explained by the fact that unsaturation in the lipid tails does not only stabilize the stalk and other fusion intermediates, but also the H_{II} phase itself (i.e. stabilize the lipoplex, preventing the release of cargo).

4.2.4. Target membrane composition severely affects fusion rate

Our results point to a clear role for the composition of the lipoplex, raising the question whether the target membrane composition has a similar effect. To this end, we investigated the role of lipid composition of the endosomal membrane on the fusion efficiency of our lipoplex models. The precise lipid composition of the endosomal membrane is not known, and varies between cell types and level of maturation. Focusing on the role of lipid tails, we considered two additional endosomal membrane models, including some longer tailed lipids (diC16:0, dipalmitoyl, DP) and lipids with an increasing amount of unsaturation (diC18:2, dilinoleoyl, DLI and diC20:4, diarachidonyl, DA; Fig. 4.2). We selected the most fusogenic lipoplex containing the DOTAP lipids, and performed again five repeats lasting ten microseconds each. Compared to the endosomal membrane model consisting of shorter lipids, described above, the fusion efficiency drops. In case of the endosomal membrane containing DLI lipids,

Figure 4.3 | Effect of lipid composition on transfection efficiency. The number of successful transfections and amount of transfected DNA considering different lipid compositions. Each composition had 5 repeats with four dsDNA fragments each. Therefore the maximum number of transfections is five and the maximum amount of transfected fragments is 20 per condition. Left of the dotted line are the fusion results for varying lipoplex formulations on a short lipid membrane. For these formulations the lipoplex formulation was 4:1 DOPE/DxTAP. On the right of the dotted line the results for varying endosomal membrane compositions with constant lipoplex formulation (4:1 DOPE/DOTAP) are displayed. The alternative endosomal membrane models are enriched in poly-unsaturated DA or di-unsaturated DLI lipids (Fig. 4.2).

no fusion events were observed at all, and the initial stalk state remained stable. For the endosomal membrane containing DA lipids, successful fusion and translocation of dsDNA was observed, but at a lower efficiency (Figure 3). When transfection occurred, the mechanism was similar as described before as judged from the analysis of lipid mixing and dsDNA orientation (Figure 3—figure supplements 1–5[1]). A possible explanation for the reduced fusion rates with the alternative endosomal membrane models is the increased stability of the bilayer by the longer lipid tails.

4.2.5. Larger lipoplexes are more stable

After performing the fusion experiments with the small lipoplex (4×24 bp) and membrane, we expanded to a larger lipoplex (12×48 bp) and target membrane to investigate the effect of size on the fusion process (Fig. 4.4 and Fig. S4.4). For the membrane conditions we selected the most reactive lipoplex formulation (DOTAP) and membrane composition (Fig. 4.2, short lipids). The large solvated lipoplex was, like the smaller lipoplex, stable over a period of 10 μ s and adopted an H_{II} crystal lattice (Fig. 4.4 A). A difference compared to the smaller lipoplex, is that the large lipoplex contained a mixture of open and closed channels, with two of the twelve channels being continuous with the solvent on one end and one of them being open on both ends. The large lipoplex also contained additional channels connecting the

H_{II} -channels, somewhat resembling a cubic phase. These connecting channels were large enough for ions, water and lipids to flow through, but the dsDNA remained strictly in the H_{II} packing. We found that these connecting channels can have different origins. Some channels can occur when the equilibration of the H_{II} phase is too short (50 ns); slightly elongating the preparation time removed such channels in the final periodic lipoplex (Fig. S4.4 C). However, increasing lipoplex hydration (lipids:watersAA 1:12) resulted in stabilization of the connecting channels (Fig. S4.4 C). In addition, connecting channels in the coated lipoplex appeared to be stable for at least 10 μ s, and the coating process itself introduced new channels. Therefore our data suggest that these channels could be kinetically trapped structures, but also represent thermodynamic equilibrium structures depending on the exact state conditions. This is not unexpected if you consider the phase diagrams of lipid mixtures, which show a rich variety of inverted, cubic, and sponge phases that can interconvert as a function of temperature, hydration and other conditions.

Figure 4.4 | Transfection of large lipoplexes. The large lipoplex (A) showed the same stable H_{II} structure as the small lipoplex with dsDNA inside the aqueous channels. Small additional connecting channels are also present, as indicated. Transfection was performed on top of a large endosomal model bilayer patch (B) and from within a model endosomal vesicle (C). In contrast to the small lipoplex, fusion did not spontaneously occur and had to be initiated using a biasing potential on one of the channels. After release of the biasing potential the dsDNA inside the pulled channel, and all dsDNA connected through connecting channels, transfected (3 DNA fragments transfected in both lamellar and vesicular system upon pulling of the same initial channel).

Fusion was, as for the small complexes, initiated from a preformed stalk with five repeats each spanning ten microseconds. In contrast to the transfection behaviour on small scale, the larger lipoplex did not show any transfection events during this time period. We did observe a reorientation of the lipoplex from an initial parallel state to a state in which the dsDNA is oriented perpendicular to the membrane sur-

face, whilst increasing the contact area (Fig. S4.5). The lack of transfection for large lipoplexes raised the question whether this inhibition was of thermodynamic or kinetic nature. Therefore we performed five extra repeats of 10 μ s, adding another biasing potential after formation of the stable hemifused complex to initiate the release of the first dsDNA fragment (see Materials and methods). The biasing potential was released after the first fusion event, that is opening of the channel. This resulted in transfection of the dsDNA inside the channel which was opened and all the dsDNA fragments in channels which shared a connection with it (Fig. 4.4 B). In total, 15 fragments were transfected (3 DNA fragments for each of the five repeats). In general, decomplexation of the dsDNA was much slower compared to the small lipoplex case, and mainly proceeded by the unzipping pathway (Figure 1IVa). Thus it appears that cargo delivery can also take place with larger lipoplexes, but that kinetic trapping is more likely to occur in either a surface contact or hemifused state.

To explain the difference in reactivity of the large lipoplex with respect to the smaller one, it seems plausible that the curvature of the lipoplex plays an important role in fusion efficiency. The same is observed in case of fusion between small liposomes and membranes[42]. However, there is a second issue that should be considered, namely the overall stability of the lipoplex. This stability is affected by the geometric discrepancy between the optimal configuration for the lipoplex coating and the inner core. Whereas the optimal configuration of the coating would be a sphere, satisfying minimal curvature constraints, the optimal configuration of the core is an H_{II} phase which does not have uniform curvature on its circumference. From these two requirements it follows that there will be an interfacial tension between the coating and the core, in addition to the interfacial tension of the coating with the surrounding solvent. If we compare the geometry of the small lipoplex to that of the large one there is a clear difference in the outer angles of the core. Considering a 2D projection along the channel axis, the small lipoplex has two 60° and two 120° angles. The large lipoplex has six 120° angles. The bimodal angle distribution of the small lipoplex is further away from the preferred constant curvature of the coating than the larger one, resulting in a higher interfacial tension per lipid. Together, the lipoplex curvature stress and the interfacial tension between the core and coating layer explain the observed trend in reduced fusion efficiency upon increase in lipoplex size.

4.2.6. Escape from the endosome

Finally, to study lipoplex fusion with endosomal membranes in a more realistic setting, we embedded a large lipoplex inside a small vesicle representing the endosome (Fig. 4.4 C). In principle, the curvature present in the endosomal membrane could further modulate the fusion pathway and efficiency. The lipid composition of the vesicular endosome consisted again of the short lipid variant (Fig. 4.2). The vesicle measured 50 nm in diameter and the total system size exceeded 3 million CG beads (representing roughly 30 million atoms). To allow simulation of this large system size, the duration of the production run was reduced to 5 μ s and only a single simulation was performed. As for the large bilayer system, no spontaneous fusion occurred, and a dsDNA fragment was pulled for initiation of transfection. Once the channel opened,

dsDNA was transfected, followed by another two fragments that shared the same aqueous space via the connecting channels (Fig. 4.4 C). Opposed to the zipper-like fusion observed in the large bilayer experiment, this time fusion occurred in a mixed manner, combining the vertical alignment with the unzipping between neighbouring channels, resulting in a slow release (Videos 2 and 3[1]). Although we can not extract generic behavior from this single experiment, the observed pathway points to a possible role of curvature on the preferred orientation of the lipoplex with respect to the endosomal membrane, with possible consequences for transfection efficiency.

4.2.7. Conclusion

We were able to successfully simulate the lipoplex mediated transfection of dsDNA over a model endosomal membrane and observed two profoundly different fusion pathways. Release of the dsDNA by rapid unzipping of the H_{II} phase occurs when the dsDNA lies perpendicular to the membrane normal, whilst an ejection-like or slow unzipping release of dsDNA is observed when the dsDNA is oriented parallel to the bilayer normal. Transitions between these states are also possible along the entire fusion pathway. Interestingly, our results for the small lipoplex indicate that transfection of the first dsDNA fragment triggers the release of a substantial amount, if not all, of the remaining genetic material likely caused by further destabilization of the lipoplex once cargo gets released.

Considering that the lipid composition of the endosomal membrane is not up for control, the differences between the lipoplex formulations appear to be most interesting from a rational design point of view. However, the endosomal bilayer composition varies per species and cell type, therefore understanding lipoplex-endosome interactions are of equal importance. We found that the transfection efficiency is sensitive to minor changes in both lipoplex and endosomal membrane composition, which is in line with experiments[9]. The highest amount of transfection was observed for the DOTAP:DOPE lipoplex. No transfection was observed for the DPTAP:DOPE complex, and DLITAP:DOPE showed intermediate fusogenicity. Interestingly, the same lipid type modifications (e.g., presence of poly-unsaturated tails) can have both stabilizing and destabilizing effects on fusion efficiency, making rational predictions for optimal delivery vehicles difficult. Moreover, our results point to an important role of lipoplex size, with the larger lipoplex being less fusogenic. This we attribute to an increase of lipoplex stability causing slower lipid mixing of the lipoplex and the target membrane. In addition, the size and shape of the lipoplex affect the way the complex orients itself and interacts with the (curved) endosomal membrane. In this study we focused our efforts on exploring the effect of varying lipid tails, keeping many other parameters of the lipoplex constant. Future endeavors could investigate the role of lipid headgroups and charge ratios, which are known to affect transfection[35, 43]. In addition, the DNA/lipid ratio and size of the DNA fragments could play a role. It would be interesting to see to what extent changing such parameters alters the observed fusion pathways and kinetics. To provide further insights into the process, the observed mechanisms need to be cast into a continuum type of description[44, 45] from which the competing energetic driving forces can be more straightforwardly extracted.

Eventually, our in-silico predictions need experimental validation, although resolving fusion pathways at the molecular level remains highly challenging. Time resolved SAXS in combination with single-molecule fluorescence microscopy could be used to probe the relative orientation of the lipoplex and embedded genetic material during the transfection process. Furthermore, leakiness assays could be used to test whether or not endosomal material leaks out concomitantly, for example via transient pore formation. Although the lipoplex fusion process described here does not involve such transient pores, we note that the presence of open channels in the lipoplex (Fig. S4.4) implies that endosomal material can become transfected along with the dsDNA even if the fusion itself is non-leaky. Besides, leaky fusion pathways have been observed in previous simulation studies between lipid membranes, and appear highly dependent on both lipid compositions and local stress conditions[20]. Therefore, it seems plausible that such transient pores could also form during lipoplex fusion.

To conclude, we demonstrated that detailed computer simulations of the fusion between a lipoplex and model endosomal membranes is nowadays possible, opening the way for systematic studies using the more advanced lipoplex formulations currently available[46–49]. Besides, the observed fusion pathways could be of generic importance for uptake of lipid-complexed nanoparticles into cells following endocytosis.

4.3. Materials and methods

4.3.1. Building the small periodic lipoplex

We followed a three step procedure for setting up the lipoplex-membrane fusion experiment. First we constructed the all-atom (AA) structure of the 24 base pairs dsDNA sequence ([CGCGAATTCGCG]₂) using the B-DNA sequence to structure builder at the website of IIT Delhi²[50]. We utilized Martinize to coarse-grain (CG) the AA structure to a Martini stiff-dsDNA force field and structure (Fig. 2.2 A)[19]. To build the periodic inverted hexagonal phase we roughly used the procedure described by Corsi et al.[23]. The CG DNA was placed on top of a bilayer constructed with the bilayer builder insane which contained DOTAP and DOPE at a 1:4 ratio (Fig. 2.2 B1; 4.2; Supplementary files 1, 2[1]; [18, 29]. This system was duplicated in its x dimension, yielding a single membrane with two dsDNA fragments stacked on top of it with a spacing of 6 nm with respect to each other, using GROMACS gmx editconf[51]. The system was hydrated to achieve a hydration of two atomistic waters per lipid and a salinity of 150 mM NaCl. In addition, each system was charge neutralized by exchanging random water beads for the appropriate counterion (Na⁺/Cl⁻). To prevent freezing of the water we added 10% anti-freeze particles[18]. The bilayer system was stacked on top of itself in the z dimension to obtain system of two bilayers and four dsDNA fragments (Fig. 2.2 C). The stacked system was energy minimized and equilibrated using GROMACS 5.1.5. For energy minimization we used the steepest descent algorithm and for the equilibration we used the default Martini settings making use of a two fs time step up to the point that numerical stability

²<http://www.scfbio-iitd.res.in/software/drugdesign/bdna.jsp>

was achieved[52]. The Verlet cut-off scheme was used with a 1.1 cut-off for both the coulombic (reaction-field) and van der Waals interactions. We used v-rescale for the thermostat at 310 K, coupling the DNA, lipids and solvent in separate groups. Pressure coupling was performed using the Berendsen barostat[53] for anisotropic systems. The production run made use of a 10 fs time step and the pressure coupling was switched to Parrinello-Rahman[54]. Within 50 ns the system changed from a stacked bilayer assembly into a stable inverted hexagonal phase (H_{II}). This general procedure was used for all lipoplex formulations. However, in the presence of a stable lamellar bilayer the pressure coupling was set to semi-isotropic, and in the case of a free particle in solution the pressure coupling was handled isotropically. A detailed description including a step by step tutorial is described in chapter 2 of this thesis.

4.3.2. Building the small solvated lipoplex

Once a stable periodic H_{II} phase was obtained, we put the complex in a bigger box to add another layer of lipids around it. The surface of the lipoplex was approximated to be a parallelepiped to simplify geometric calculations. We used a symmetrical bilayer system to calculate the area per lipid (APL) of the DOPE, DOTAP mixture (Supplementary file 2[1]). The approximated area of the lipoplex was divided by the APL of the lipid mixture. The calculated amount of lipids were added in a hollow cube around the naked lipoplex using PACKMOL[55]. Water and ions were added as before. To test for the accuracy of this method, we also tried half and double the calculated amount and investigated lipoplex stability (Fig. S4.2). The system was solvated, energy minimized and equilibrated using the same settings as before except for the pressure coupling which was set to isotropic. The production run spanned 10 μ s. The radius of gyration (gmx gyration) and hydration (in-house VMD based selection counter) of the lipoplex were analyzed over time, to evaluate the equilibrated state (Fig. S4.1). This general procedure was used for all lipoplex formulations. The DP-TAP and DLi-TAP containing lipoplexes were generated from the DOTAP lipoplex by exchanging the DOTAP lipid force field with either DPTAP or DLi-TAP plus an extra step of equilibration (5 μ s). The small lipoplex had a diameter of 20 nm and the box dimensions were 28.1 28.1 19.8 nm for xyz respectively resulting in a total of 118,718 beads. The periodic boundaries were set to dodecahedron. A detailed description including a step by step tutorial is described in chapter 2 of this thesis.

4.3.3. Building the large lipoplex

The same procedure to build the small lipoplexes was used to construct the large lipoplex. However, instead of four dsDNA fragments these lipoplexes contained twelve dsDNAs which had a length of 48 base pairs ([CGCGAATTCGCG]4) each. The final lipoplex had a diameter of 30 nm.

4.3.4. Building the lipoplex-bilayer system (small)

To investigate the fusion behaviour of lipoplexes with membranes, we built several lipoplexes with varying composition. We prepared a system which has long polyunsaturated tails (DAPC), medium unsaturated tails (DLiPC) and short single unsatu-

rated tails (DYPC) in the bilayer as its unsaturated component (Figure 2). In the case of the short lipid bilayer, all the lipids were shorter. The solvated lipoplex was placed on top of the appropriate bilayer using insane[29]. The systems were solvated and salinized, energy minimized and equilibrated as before, but the pressure coupling was set to semi-isotropic in the xy and z dimensions (with the membrane normal in z). After the system was equilibrated and the lipoplex adhered to the membrane, a pulling potential was added between 10 lipids of the outer leaflet of the lipoplex in the proximity of the bilayer and vice versa. The initial center of geometry distance between the pulling groups was less than 3 nm and the pulling lasted for less than 5 ns (harmonic biasing force; constant velocity -1 m/s). After the initial stalk was formed, the biasing potential was removed. The system was equilibrated and a 10 μ s transfection experiment was performed five times with random starting velocities for each membrane and lipoplex formulation. Again the pressure coupling was set to semi-isotropic. The small lipoplex bilayer box dimensions were 27.5 27.2 31.6 nm in xyz respectively with a total of 193,083 beads. A detailed description including a step by step tutorial is described in chapter 2 of this thesis.

4.3.5. Building the lipoplex-bilayer system (large)

Using the optimal performing bilayer and lipoplex from the small lipoplex-bilayer experiments, we set up a system with a large lipoplex (DOPE:DOTAP; 12*48 bp) and the short lipids bilayer (Fig. 4.2). Adhesion and stalk formation was performed identical to the small version of the experiment. However, after stalk formation two flavors of the experiment were conducted. In the first variant the experiment was left untouched after stalk formation for 10 μ s. In the second variant of the experiment, we simulated until a stable fusion intermediate (no transfection) was obtained and then we initiated another pull. However, this time not pulling the tails of lipids in opposing leaflets together, but the heads of the lipids directly above and under the stalk/diaphragm. Thus pulling headgroups of lipids from inside one of the channels in the lipoplex, towards the headgroups of the opposing side of the bilayer. The initial centre of geometry distance between the two pulling groups was 5 nm and lasted up to five ns (harmonic biasing force; constant velocity -1 m/s). This initiated a fusion pore from which transfection followed. After instantiation of the fusion pore the biasing potential was removed, allowing for unbiased transfection of the following fusion and transfection events. After the biasing potential was released five simulations were started from this point with random initial velocities, each running for 10 μ s. The large lipoplex-bilayer box dimensions were 45.8 45.8 44.5 nm in xyz respectively, with a total of 749,538 beads.

4.3.6. Building the lipoplex-vesicle system (large)

As for the large lipoplex-bilayer system, the best performing lipoplex and bilayer composition in the small experiments were used. The initial state of the vesicle was constructed using an in-house Python three script based on a spherical fibonacci spiral (available on request). The amount of lipids placed on the inside was corrected for the area difference between the inner and outer leaflet, and the APL of DLPC was used to calculate the amount of lipids needed. After placement of the DLPC,

the system was solvated and salted using insane and the appropriate lipid force fields we added in the topology file. The same protocol used in Qi et al.[56], and Risselada et al.[57] to equilibrate a vesicle was used, relaxing the leaflet asymmetry through pores. After equilibration of the inter-leaflet tension (100 ns; production-md-settings with isotropic pressure coupling) the vesicle was dehydrated and desalted. The lipoplex plus its surrounding ions, waters and DNA were placed at the center of the porated vesicle and the system was rehydrated and salted using insane. This system was equilibrated for 800 ns, allowing for the ion content to equilibrate as well, tension release due to leaflet asymmetry occurred in a few ns. The biasing potential for the pore was removed, which rapidly caused closing of the pores (<10 ns). The whole complex was then simulated for 1 μ s during which the lipoplex adhered to the inside of the vesicle. From here the stalk was initiated as before for the small and large lipoplex bilayer simulations. As for the large lipoplex-bilayer simulations, the experiment was split in two, one being the unbiased simulation of the stalk and possible transfection (1 μ s). The second experiment included another biasing potential to initiate channel-vesicle fusion. The pulling procedures used were exactly the same as for the large lipoplex-bilayer systems and after release the unbiased lipoplex-vesicle simulation spanned 5 μ s. The large lipoplex-vesicle box dimensions were 72.8 72.8 72.8 nm in xyz respectively, with a total of 2,994,561 beads.

4.3.7. Analysis

The dsDNA spacing in the small solvated lipoplex was calculated using the center of mass (COM) distance between the first neighbours of each dsDNA over the last 2 μ s of the solvated lipoplex ($n = 5$). We made a Python3 script to analyze the angle of the dsDNA with respect to the z unit vector (normal to the average bilayer plane), using the COM of the first and last base pairs to construct the dsDNA orientational vector. The script treats the dsDNA termini as equals, therefore no higher angle with the membrane normal than 90° can be obtained. The same VMD based counting script used for the calculation of hydration was used to calculate the lipid contacts and dsDNA-water contacts (cutoff 1.2 nm). The dsDNA-water contacts were used for the automated transfection detection script, which reported a transfection event whenever dsDNA hydration rapidly increased (more than 200 CG waters in less than 400 ns). The automated transfection detection was manually checked on five simulations, both with and without fusion, and was always correct. We used all the information above to investigate the angle with respect to transfection, as well as the importance of lipid mixing. VMD 1.9.3 was used to render the images and videos[58]. The Tcl script used in combination with VMD to perform fast selection counting, and the Python script to perform the angle analysis and fusion detection are available in the repository. In general Python three and MDAnalysis were used extensively in the analysis and visualization[59, 60].

4.4. Acknowledgments

We would like to thank Prof. dr. D Hoekstra for his ideas and comments on the construction of the lipoplex model. His experience within the lipoplex field helped

us to mature many of the design choices made in this research. We would also like to thank Dr. T Wassenaar and Dr. J Barnoud for helping with some of the problems we had during coding. SJM acknowledges funding from the ERC through an Advanced grant 'COMP-MICR-CROW-MEM'. This work was carried out on the Dutch national e-infrastructure with the support of SURF-SARA, and the Peregrine HPC at the University of Groningen.

4.5. Supplementary Information

Figure S4.1 | Radius of gyration for the solvated lipoplex. The radius of gyration for the solvated small lipoplex over time. Over the last 2 μs of the simulation the radius of gyration increases less than a quarter of an Ångström. We therefore concluded that the lipoplex complexes are close enough to equilibrium to use for transfection experiments. The water contacts (<1.5 nm) with the lipoplex were calculated using the inhouse VMD counting script. The solvation of the lipoplex shows the same trend as the radius of gyration also confirming that the lipoplex is stable and can be used for transfection. Since the hydration and gyration show the same (minor) growth, we think that the observed growth in size of the lipoplex is due to a minor underhydration upon construction of the periodic lipoplex.

Figure S4.2 | Coating the lipoplex. Upon solvating the periodic lipoplex an extra layer of coating lipids was added. We used the approximate surface area of the lipoplex and divided it by the APL of the DOTAP:DOPE mixture (1:4; 0.63 nm^2). From left to right we have 1/2, 1/1, 2/1 times the calculated amount. It is clear that underestimating the amount of coating lipids leads to loss of core lipids resulting in the loss of the internal structure and internalized dsDNA (left image). Oversaturating the core with coating lipids results in internalization of coating lipids, resulting in a new channel within the crystal lattice (right image). The expected amount results in no loss of internal structure of dsDNA, and no extra channels are formed (central image). Each condition was simulated for $10 \mu\text{s}$ after adding the coating.

Figure S4.3 | Lipid mixing in all endosome bilayer compositions. Snapshots of endosome bilayers after $1 \mu\text{s}$ of equilibration after being generated by insane. From left to right we show the most, more unsaturated and short lipids bilayer respectively (see Figure 2). In purple the unsaturated carbons are displayed. Teal represents the saturated carbons. Only the most unsaturated bilayer shows phase separation (left, DA).

Figure S4.4 | Large lipoplex channel structures with dsDNA inside. The solvated large lipoplex contained channels: regular dsDNA/aqueous channels are indicated in peach (A); an open channel is depicted with a magenta circle (B); and aqueous connecting channels are indicated with red arrows (A/B). The water is drawn in ice-blue (A) and the lipid density in grey (B). The connecting channels were mainly perpendicular to the H_{II} channels and positioned at the end point(s) of the dsDNA (A). All H_{II} channels but two were closed on both ends by lipid density. One channel remained open on both sides over a period of 10 μ s, with the dsDNA in the open channel positioned in the lipoplex in the same manner as the dsDNA in the closed channels. Apparently opening a channel is not enough for dsDNA release from the complex. The bottom panel shows, from left to right, the channel connectivity for the large periodic phase under standard, elongated, and extra hydrated conditions (C). The spacing of the H_{II} phase is already finished at 180 ns (the used equilibration time), however, some connective channels still exist, indicated by neighboring channels having the same color. These connections could be closed by equilibrating slightly longer (230 ns in total). To inspect if these channels could be stabilized, we increased the hydration (lipids:watersAA 1:12). This indeed resulted in connective channels being present until the end of the simulation.

Figure S4.5 | Reorientation of the large lipoplex in the endosome. The graph shows the lipoplex-endosome lipid contacts (<1.2 nm) over time. Within $0.5 \mu\text{s}$ the lipoplex reorients itself such that its H_{1L} -channels are roughly aligned with the membrane normal in the contact region. Components of the lipoplex closer than 4 nm to the endosome are colored blue (a longer cut-off was used to help with perception of orientation), the endosomal lipids are grey and the dsDNA inside the lipoplex is yellow/pink.

4.6. References

- [1] B. M. H. Bruininks, P. C. T. de Souza, H. Ingolfsson, and S. J. Marrink, *A molecular view on the escape of lipoplexed DNA from the endosome*, *eLife* **9** (2020), 10.7554/elife.52012.
- [2] D. Cressey, *Europe nears first approval for gene therapy*, *Nature* (2012), 10.1038/nature.2012.11048.
- [3] R. M. Blaese, K. W. Culver, A. D. Miller, C. S. Carter, T. Fleisher, M. Clerici, G. Shearer, L. Chang, Y. Chiang, P. Tolstoshev, J. J. Greenblatt, S. A. Rosenberg, H. Klein, M. Berger, C. A. Mullen, W. J. Ramsey, L. Muul, R. A. Morgan, and W. F. Anderson, *T lymphocyte-directed gene therapy for ADA- SCID: Initial trial results after 4 years*, *Science* **270**, 475 (1995).
- [4] E. Hanna, C. Rémuzat, P. Auquier, and M. Toumi, *Gene therapies development: slow progress and promising prospect*, *Journal of Market Access & Health Policy* **5**, 1265293 (2017).
- [5] S. Nayak and R. W. Herzog, *Progress and prospects: immune responses to viral vectors*, *Gene Therapy* **17**, 295 (2009).
- [6] C. T. de Ilarduya, Y. Sun, and N. Düzgüneş, *Gene delivery by lipoplexes and polyplexes*, *European Journal of Pharmaceutical Sciences* **40**, 159 (2010).
- [7] M. Rezaee, R. K. Oskuee, H. Nassirli, and B. Malaekheh-Nikouei, *Progress in the development of lipopolyplexes as efficient non-viral gene delivery systems*, *Journal of Controlled Release* **236**, 1 (2016).
- [8] N. Nayerossadat, P. Ali, and T. Maedeh, *Viral and nonviral delivery systems for gene delivery*, *Advanced Biomedical Research* **1**, 27 (2012).
- [9] L. Huang and S. Li, *Liposomal gene delivery: A complex package*, *Nature Biotechnology* **15**, 620 (1997).
- [10] B.-K. Kim, G.-B. Hwang, Y.-B. Seu, J.-S. Choi, K. S. Jin, and K.-O. Doh, *DOTAP/DOPE ratio and cell type determine transfection efficiency with DOTAP-liposomes*, *Biochimica et Biophysica Acta (BBA) - Biomembranes* **1848**, 1996 (2015).
- [11] L. Ciani, S. Ristori, A. Salvati, L. Calamai, and G. Martini, *DOTAP/DOPE and DC-chol/DOPE lipoplexes for gene delivery: zeta potential measurements and electron spin resonance spectra*, *Biochimica et Biophysica Acta (BBA) - Biomembranes* **1664**, 70 (2004).
- [12] P. L. Felgner, T. R. Gadek, M. Holm, R. Roman, H. W. Chan, M. Wenz, J. P. Northrop, G. M. Ringold, and M. Danielsen, *Lipofection: a highly efficient, lipid-mediated DNA-transfection procedure*. *Proceedings of the National Academy of Sciences* **84**, 7413 (1987).

- [13] Z. ur Rehman, D. Hoekstra, and I. S. Zuhorn, *Mechanism of polyplex- and lipoplex-mediated delivery of nucleic acids: Real-time visualization of transient membrane destabilization without endosomal lysis*, ACS Nano **7**, 3767 (2013).
- [14] J. REJMAN, V. OBERLE, I. S. ZUHORN, and D. HOEKSTRA, *Size-dependent internalization of particles via the pathways of clathrin- and caveolae-mediated endocytosis*, Biochemical Journal **377**, 159 (2004).
- [15] I. S. Zuhorn, J. B. F. N. Engberts, and D. Hoekstra, *Gene delivery by cationic lipid vectors: overcoming cellular barriers*, European Biophysics Journal **36**, 349 (2006).
- [16] I. M. S. Degors, C. Wang, Z. U. Rehman, and I. S. Zuhorn, *Carriers break barriers in drug delivery: Endocytosis and endosomal escape of gene delivery vectors*, Accounts of Chemical Research **52**, 1750 (2019).
- [17] S. J. Marrink, V. Corradi, P. C. de Souza, H. I. Ingólfsson, D. P. Tieleman, and M. S. Sansom, *Computational modeling of realistic cell membranes*, Chemical Reviews **119**, 6184 (2019).
- [18] S. J. Marrink, H. J. Risselada, S. Yefimov, D. P. Tieleman, and A. H. de Vries, *The MARTINI force field: coarse grained model for biomolecular simulations*, The Journal of Physical Chemistry B **111**, 7812 (2007).
- [19] J. J. Uusitalo, H. I. Ingólfsson, P. Akhshi, D. P. Tieleman, and S. J. Marrink, *Martini coarse-grained force field: Extension to DNA*, Journal of Chemical Theory and Computation **11**, 3932 (2015).
- [20] A. J. Markvoort and S. J. Marrink, *Lipid acrobatics in the membrane fusion arena*, in *Current Topics in Membranes* (Elsevier, 2011) pp. 259–294.
- [21] Y. G. Smirnova, H. J. Risselada, and M. Müller, *Thermodynamically reversible paths of the first fusion intermediate reveal an important role for membrane anchors of fusion proteins*, Proceedings of the National Academy of Sciences **116**, 2571 (2019).
- [22] M. Pannuzzo, D. H. D. Jong, A. Raudino, and S. J. Marrink, *Simulation of polyethylene glycol and calcium-mediated membrane fusion*, The Journal of Chemical Physics **140**, 124905 (2014).
- [23] J. Corsi, R. W. Hawtin, O. Ces, G. S. Attard, and S. Khalid, *DNA lipoplexes: Formation of the inverse hexagonal phase observed by coarse-grained molecular dynamics simulation*, Langmuir **26**, 12119 (2010).
- [24] I. Koltover, *An inverted hexagonal phase of cationic liposome-DNA complexes related to DNA release and delivery*, Science **281**, 78 (1998).
- [25] G. van Meer, D. R. Voelker, and G. W. Feigenson, *Membrane lipids: where they are and how they behave*, Nature Reviews Molecular Cell Biology **9**, 112 (2008).

- [26] Y. S. Tarahovsky, R. Koynova, and R. C. MacDonald, *DNA release from lipoplexes by anionic lipids: Correlation with lipid mesomorphism, interfacial curvature, and membrane fusion*, *Biophysical Journal* **87**, 1054 (2004).
- [27] F. Cohen and G. Melikyan, *The energetics of membrane fusion from binding, through hemifusion, pore formation, and pore enlargement*, *Journal of Membrane Biology* **199**, 1 (2004).
- [28] W. D. Bennett and D. P. Tieleman, *Water defect and pore formation in atomistic and coarse-grained lipid membranes: Pushing the limits of coarse graining*, *Journal of Chemical Theory and Computation* **7**, 2981 (2011).
- [29] T. A. Wassenaar, H. I. Ingólfsson, R. A. Böckmann, D. P. Tieleman, and S. J. Marrink, *Computational lipidomics with insane: A versatile tool for generating custom membranes for molecular simulations*, *Journal of Chemical Theory and Computation* **11**, 2144 (2015).
- [30] S. L. Leikin, M. M. Kozlov, L. V. Chernomordik, V. S. Markin, and Y. A. Chizmadzhev, *Membrane fusion: Overcoming of the hydration barrier and local restructuring*, *Journal of Theoretical Biology* **129**, 411 (1987).
- [31] V. Knecht and S. J. Marrink, *Molecular dynamics simulations of lipid vesicle fusion in atomic detail*, *Biophysical Journal* **92**, 4254 (2007).
- [32] S. Aeffer, T. Reusch, B. Weinhausen, and T. Salditt, *Energetics of stalk intermediates in membrane fusion are controlled by lipid composition*, *Proceedings of the National Academy of Sciences* **109**, E1609 (2012).
- [33] Y. Kozlovsky, L. V. Chernomordik, and M. M. Kozlov, *Lipid intermediates in membrane fusion: Formation, structure, and decay of hemifusion diaphragm*, *Biophysical Journal* **83**, 2634 (2002).
- [34] R. Koynova, B. Tenchov, L. Wang, and R. C. MacDonald, *Hydrophobic moiety of cationic lipids strongly modulates their transfection activity*, *Molecular Pharmaceutics* **6**, 951 (2009).
- [35] B. Ma, S. Zhang, H. Jiang, B. Zhao, and H. Lv, *Lipoplex morphologies and their influences on transfection efficiency in gene delivery*, *Journal of Controlled Release* **123**, 184 (2007).
- [36] S. Fletcher, A. Ahmad, E. Perouzel, A. Heron, A. D. Miller, and M. R. Jorgensen, *In vivo studies of dialkynoyl analogues of DOTAP demonstrate improved gene transfer efficiency of cationic liposomes in mouse lung*, *Journal of Medicinal Chemistry* **49**, 349 (2006).
- [37] W. E. Teague, N. L. Fuller, R. P. Rand, and K. Gawrisch, *Polyunsaturated lipids in membrane fusion events*, *Cell Mol Biol Lett* **7**, 262 (2002).

- [38] Y. G. Smirnova, S. J. Marrink, R. Lipowsky, and V. Knecht, *Solvent-exposed tails as prestalk transition states for membrane fusion at low hydration*, Journal of the American Chemical Society **132**, 6710 (2010).
- [39] Z.-A. Fan, K.-Y. Tsang, S.-H. Chen, and Y.-F. Chen, *Revisit the correlation between the elastic mechanics and fusion of lipid membranes*, Scientific Reports **6** (2016), 10.1038/srep31470.
- [40] Z. A. Levine, R. M. Venable, M. C. Watson, M. G. Lerner, J.-E. Shea, R. W. Pastor, and F. L. H. Brown, *Determination of biomembrane bending moduli in fully atomistic simulations*, Journal of the American Chemical Society **136**, 13582 (2014).
- [41] J. Pan, X. Cheng, M. Sharp, C.-S. Ho, N. Khadka, and J. Katsaras, *Structural and mechanical properties of cardiolipin lipid bilayers determined using neutron spin echo, small angle neutron and x-ray scattering, and molecular dynamics simulations*, Soft Matter **11**, 130 (2015).
- [42] S. Kawamoto, M. L. Klein, and W. Shinoda, *Coarse-grained molecular dynamics study of membrane fusion: Curvature effects on free energy barriers along the stalk mechanism*, The Journal of Chemical Physics **143**, 243112 (2015).
- [43] S. Eastman, C. Siegel, J. Tournant, A. Smith, S. Cheng, and R. Scheule, *Biophysical characterization of cationic lipid:DNA complexes*, Biochimica et Biophysica Acta (BBA) - Biomembranes **1325**, 41 (1997).
- [44] S. May and A. Ben-Shaul, *Modeling of cationic lipid-DNA complexes*, Current Medicinal Chemistry **11**, 151 (2004).
- [45] M. Hamm and M. Kozlov, *Elastic energy of tilt and bending of fluid membranes*, The European Physical Journal E **3**, 323 (2000).
- [46] A. L. Barrán-Berdón, S. K. Misra, S. Datta, M. Muñoz-Úbeda, P. Kondaiah, E. Junquera, S. Bhattacharya, and E. Aicart, *Cationic gemini lipids containing polyoxyethylene spacers as improved transfecting agents of plasmid DNA in cancer cells*, Journal of Materials Chemistry B **2**, 4640 (2014).
- [47] M. Al-Dulaymi, W. Mohammed-Saeid, A. El-Aneed, and I. Badea, *Peptide-modified gemini surfactants: Preparation and characterization for gene delivery*, in *Pharmaceutical Nanotechnology* (Springer New York, 2019) pp. 203–225.
- [48] P. Severino, M. Szymanski, M. Favaro, A. R. Azzoni, M. V. Chaud, M. H. A. Santana, A. M. Silva, and E. B. Souto, *Development and characterization of a cationic lipid nanocarrier as non-viral vector for gene therapy*, European Journal of Pharmaceutical Sciences **66**, 78 (2015).
- [49] T. L. Nascimento, H. Hillaireau, J. Vergnaud, M. Rivano, C. Deloménie, D. Courilleau, S. Arpicco, J. S. Suk, J. Hanes, and E. Fattal, *Hyaluronic acid-conjugated lipoplexes for targeted delivery of siRNA in a murine metastatic lung cancer model*, International Journal of Pharmaceutics **514**, 103 (2016).

- [50] G. D. Fasman, ed., *Cumulative Series Index for CRC Handbook of Biochemistry and Molecular Biology* (CRC Press, 2019).
- [51] M. J. Abraham, T. Murtola, R. Schulz, S. Páll, J. C. Smith, B. Hess, and E. Lindahl, *GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers*, *SoftwareX* **1-2**, 19 (2015).
- [52] D. H. de Jong, S. Baoukina, H. I. Ingólfsson, and S. J. Marrink, *Martini straight: Boosting performance using a shorter cutoff and GPUs*, *Computer Physics Communications* **199**, 1 (2016).
- [53] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola, and J. R. Haak, *Molecular dynamics with coupling to an external bath*, *The Journal of Chemical Physics* **81**, 3684 (1984).
- [54] M. Parrinello and A. Rahman, *Polymorphic transitions in single crystals: A new molecular dynamics method*, *Journal of Applied Physics* **52**, 7182 (1981).
- [55] L. Martínez, R. Andrade, E. G. Birgin, and J. M. Martínez, *PACKMOL: A package for building initial configurations for molecular dynamics simulations*, *Journal of Computational Chemistry* **30**, 2157 (2009).
- [56] Y. Qi, H. I. Ingólfsson, X. Cheng, J. Lee, S. J. Marrink, and W. Im, *CHARMM-GUI martini maker for coarse-grained simulations with the martini force field*, *Journal of Chemical Theory and Computation* **11**, 4486 (2015).
- [57] H. J. Risselada, A. E. Mark, and S. J. Marrink, *Application of mean field boundary potentials in simulations of lipid vesicles*, *The Journal of Physical Chemistry B* **112**, 7438 (2008).
- [58] W. Humphrey, A. Dalke, and K. Schulten, *VMD: Visual molecular dynamics*, *Journal of Molecular Graphics* **14**, 33 (1996).
- [59] R. Gowers, M. Linke, J. Barnoud, T. Reddy, M. Melo, S. Seyler, J. Domański, D. Dotson, S. Buchoux, I. Kenney, and O. Beckstein, *MDAnalysis: A python package for the rapid analysis of molecular dynamics simulations*, in *Proceedings of the 15th Python in Science Conference (SciPy, 2016)*.
- [60] N. Michaud-Agrawal, E. J. Denning, T. B. Woolf, and O. Beckstein, *MDAnalysis: A toolkit for the analysis of molecular dynamics simulations*, *Journal of Computational Chemistry* **32**, 2319 (2011).

5

Characterization and skin cell membrane penetration of a multicomponent lipid nanodroplet loaded with vitamins

If science is to progress, what we need is the ability to experiment, honesty in reporting results, and finally – an important thing – the intelligence to interpret the results.

Richard P. Feynman

*L*ipid nanoemulsions are promising nanomaterials for drug delivery applications in food, pharmaceutical and cosmetic industries. Despite the noteworthy commercial interest, little is known about their supramolecular organization, especially about how such multicomponent formulations interact with cell membranes. In the present work, coarse-grained molecular dynamics simulations have been employed to study the self-assembly of a 15-component lipid nanoemulsion droplet containing vitamins A and E for skin delivery. Our results display aspects of the unique “onion-like” agglomeration between the chemical constituents in the different layers of the lipid nanodroplet. Vitamin E molecules are more concentrated in the center of the droplet together with

This chapter has been made in collaboration with Neila Machado, Priyanka Singh, Laurita dos Santos, Carine Dal Pizzol, Gustavo de C. Dieamant, Odivanía Krüge, Aírton A. Martín, Siewert J. Marrink, Paulo, C. T. Souza and Priscila P. Favero

other hydrophobic constituents such as the triglycerides with long tails. On the other hand, vitamin A occupies an intermediate layer between the core and the co-emulsifier surface of the nanodroplet, together with lecithin phospholipids. Coarse-grained molecular dynamics simulations were also performed to provide insight into the first steps involved in absorption and penetration of the nanodroplet through skin membrane models, representing an intracellular (hair follicle infundibulum) and intercellular pathway (stratum corneum) through the skin. Our data provide a first view on the complex organization of commercial nanoemulsion and its interaction with skin membranes. We expect our results to open the way towards the rational design of such nanomaterials.

Keywords: lipid nanoemulsion (LNE), vitamin A, vitamin E, infundibulum (INF), stratum corneum (SC), coarse-grained (CG), molecular dynamics (MD), Martini force field

5

5.1. Introduction

A nanoemulsion is a colloidal particle system with a droplet size in the range of 20 to 500 nm. Immiscible liquids are mixed to form a liquid-in-liquid dispersion, by means of an emulsifying agent like surfactants and co-surfactants[1, 2]. These kinetically stable systems appear to be promising for efficient delivery of active molecules. The range of nanoemulsion applications spans diverse fields including drug delivery, as formulations for transdermal delivery of celecoxib[3]; the food industry, where bioaccessibility of long chain triglyceride nanoemulsions with β -carotene have been investigated[4]; and in the cosmetic industry where nanoemulsions have been tested for skin hydration and ease of application[5, 6]. The main advantages of nanoemulsions over other drug delivery vectors are their capability to: (i) encapsulate both hydrophilic and hydrophobic molecules; (ii) improve the bioavailability of such molecules; (iii) deliver selectively at a target site; (iv) and finally, they have reduced toxicity[7, 8].

Although a great variety of nanoemulsion formulations have been synthesized and tested, only limited high-resolution data exist regarding supramolecular organization of such materials. Relevant efforts have been made using a multitude of experimental techniques (e.g. solid-state NMR, AFM) to investigate, for instance: nanoparticle self-assembly from amphiphilic peptides[9]; complexes of cationic lipids and oligonucleotides[10]; and direct and reverse nanodroplets of water, oil and dioctyl sodium sulfosuccinate[11] as well as the interactions of such nanodroplet with membranes. However, the inherently small size of nanodroplets and the often short life times of transition states makes simulation an important complementary approach to experiment[12, 13]. All-atom (AA) and coarse-grained (CG) molecular dynamics (MD) simulations have been used for studies of simple binary and ternary mixtures, representative of nanoemulsion formulations[14–17], and have steadily increased in complexity and scale. Computational and experimental studies involving the interaction of nanocapsules with lipid bilayers have been performed in the last years[18–20]. Most of these simulations involved hard solid-like nanoparticles as the ones composed

by fullerene[21, 22] or gold coated nanoparticles with polymers or surfactants[23, 24]. The wrapping process of softer dendrimer nanoparticles by membrane models has been investigated using dissipative particle dynamics[25] and CG simulations were used to study the fusion of perfluorocarbon-based nanodroplets with phospholipid vesicles[26]. Except for recent efforts investigating lipoplex fusion with endosomal bilayer models[27, 28], most of the computational studies used rather simple bilayer models, and they were focused on understanding how the nanoparticle can penetrate or change the bilayer properties[29]. Recently, much effort has been devoted to model more complex membranes, including skin membranes[30–32], which open the possibility to simulate the uptake of nanodroplets more realistically.

In the current work, we performed CG MD simulations to study the complex organization of nanodroplets from a commercial nanoemulsion formulation. Such nanoemulsions are used as a cosmetic for the delivery of hydrophobic vitamins A and E through the skin. We show that a nanodroplet of given composition presents an onion-like organization, with vitamins A and E differently distributed in different layers of the droplet. Besides, we investigate the onset of the fusion of the multicomponent nanodroplet with two complex bilayer models, mimicking the hair follicle (infundibulum, INF) and stratum corneum (SC) of human skin. To do so we utilized the CG Martini force field for it provides a wide range of compatible molecules allowing for simulation of complex mixtures, with only limited additional parametrization[33, 34]. Another important aspect of CG is that the potentials are smoother and the simulations contain less particles, offering a 3-order speedup compared to AA models. This allows for equilibrating the structural organization of the multicomponent nanodroplets near to their real size, and to simulate the initial stages of the fusion of the nanodroplet with skin membranes.

5.2. Methods

5.2.1. Building the lipid nanodroplet

The chemical composition of the lipid nanoemulsion was provided by O. Boticário's company, which is involved in the production and marketing of this formulation. A complete overview of the composition can be found in Table 1. The molar percentage of components of the same type were equally distributed.

The major carrier components of the formulation are triglycerides, which are often used as carriers of fat-soluble vitamins and other bioactives. The ST21 (polymer-based polyethylene glycol with a 16C acyl terminal) is added to enhance emulsion stability.

In order to produce the lipid nanodroplet model for the nanoemulsion, we first assembled the hydrophobic compounds. Then we added the amphiphilic constituents to form the final nanodroplet. These stages correspond to the experimental process of forming the lipid nanoemulsion (personal communication)[35]. In the first phase, vitamins A and E were mixed with triglycerides (capric and caprylic). The remaining excipients were added in phase two. For the first stage of lipid nanoemulsion formation, we used a box of 26.26-26 nm³ with 123,938 CG water beads (corresponding to 495,752 water molecules). After 1,000 steps of minimization, stage 1 was equili-

Table 5.1 | Components of the lipid nanoemulsion by type and molar percentage. The name of the components follows the INCI nomenclature for cosmetic ingredients.

Component	Abbreviation	Type	Molar Percentage %
Retinol	VITA	vitamin A	20.06
Tocopherol	VITE	vitamin E	20.03
Caprylic/Capric triglyceride	CPLC/CPRC	triglyceride	28.79
Cetyl palmitate	CPAL	lipid	7.21
Glyceryl Stearate Cetyl Alcohol Lauryl Alcohol Behenyl Alcohol Myristyl Alcohol Palmitic Acid Stearic Acid	GLST LAAL BHAL MYAL PALAC STAC	blend lipids	4.27
Glycerin	GLC	humectant	4.80
Hydrogenated Lecithin 90G	HLEC	phospholipid	4.80
Lecithin 80H	LEC	phospholipid	3.02
Steareth-21	ST21	co-emulsifier	7.02

brated for 400 ns at an elevated temperature of 353 K to speed up the equilibration. In the second stage, we used a slightly smaller box of $26 \cdot 26 \cdot 26$ nm with 103,193 CG water beads. The equilibration of stage 2 covered 1,000 ns. A final step was performed for additional 1,000 ns to equilibrate the nanodroplet at 310 K. In all steps, the pressure was controlled with an isotropic coupling scheme.

5.2.2. Building the skin bilayer models

The biochemical composition of the skin is strongly dependent on both extrinsic and intrinsic factors (such as sun exposure, diet, and daily habits; chronological aging)[36–38], thus, the models that we intend to use to mimic the composition of the skin must contain information that particularizes the case to be studied. Our studies were based on the construction of skin models of healthy people[39]. Fig. S5.1 and S5.2 illustrate the INF and SC membranes and the possible permeation pathways in which the nanodroplet interacts with the respective membrane. Bilayer compositions are based on Machado et al. 2016[40]. The INF bilayer includes 14 different types of compounds, among them cholesterol, ceramide, sphingomyelins and various types of phospholipids. This membrane shows similar characteristics to the plasma membrane model of Ingólfsson et al. 2015[41]. The SC was composed of a mixture of ceramides, cholesterol and fatty acids. As low pH conditions could be important for the permeability and phase state of SC bilayers[42, 43], we built two models in this case: neutral (low pH, with fatty acids fully protonated) and charged (neutral pH, with fatty acids deprotonated). All simulation boxes were built with insane.py[34], with the bilayers placed in a box of around $35 \cdot 35 \cdot 35$ nm³ in size, which was solvated with $\sim 300,000$ CG water beads and 0.15 M of NaCl. After 1,000 steps of minimization, the systems were equilibrated at 310 K for 1,000 ns. Pres-

sure was controlled semi-isotropically. The temperatures of the membrane, and the solvent were maintained separately at 310 K.

5.2.3. Fusion simulations

The equilibrated nanodroplet and membranes generated in the previous steps were used as initial conditions for the fusion simulations. We placed the nanodroplet on top of each bilayer at around 9.7 nm distance between their centers of geometry. At this distance, the nanodroplet starts with one to two water solvent shells between its surface and the bilayer head-groups. The standard sized fusion experiments (43·43·29 nm³) contained ~33,451 CG water beads and 0.15 M NaCl. Production simulations were performed at 310 K and 1 bar during at least 1,000 ns. As the bilayers could show great deformations during the fusion, three procedures were used to investigate the process: (i) “free” bilayer (semiisotropic pressure coupling), (ii) flat bilayer (surface tension coupling); and (iii) constrained bilayer (semiisotropic pressure coupling in combination with flat-bottom potentials applied to certain phospholipids). In the case of INF bilayers, a box with lateral dimensions four times larger was also tested.

5

5.2.4. CG models

The Martini force field was used for all CG simulations[33, 44]. Concerning the nanodroplet, topologies of standard lipids were taken from the work of Wassenaar et al. 2014[34]. Topologies for a number of less standard components were optimized to improve their mapping and water-octanol partitioning, including capric and caprylic triglycerides, hydrogenated lecithin cetyl palmitate, glyceryl stearate, cetyl alcohol, lauryl alcohol, behenyl alcohol, myristyl alcohol, palmitic acid, stearic acid, the mapping for these molecules can be found in table S5.1. The topology for the co-emulsifier Steareth 21 was based on the work of Rossi G. et al. 2012[45]. Vitamin A (Retinol), vitamin E (Tocopherol) and humectant (Glycerin) were parameterized de novo, following the standard procedure as described by Marrink et al.[33, 44, 45]. For the non-bonded parameters, the water-octanol partitioning free energy of the CG model was compared to the experimental value for each molecule (Table S5.2). To calculate these free energies of transfer we used the same methodology as was used in[46]. For the bonded parameters, the suggested default values were used (Table S5.3). For the skin bilayer lipids, the current Martini models were used [33, 34, 47].

5.2.5. Simulation parameters

All simulations were performed using GROMACS 5.x/2016[48–52]. The parameters were accordingly taken from the Martini default settings[53]. Each CG production run was preceded by an initial energy minimization of the system using the steepest descent algorithm. The integrator used for equilibration and production runs was the leap-frog algorithm[54], with an integration time step of 20 fs. Neighbour searching was calculated using the Verlet scheme[55] at a 20 step interval. Electrostatic interactions were calculated using the Reaction-Field algorithm with a dielectric constant of 15. The cut-off for van der Waals and electrostatic interactions were set at 1.1 nm, furthermore, the van der Waals and Coulombic interactions were shifted be-

tween 0.9-1.1 and 0.0-1.1 nm respectively using Potential-shift-Verlet. Temperature was set using V-rescale[56, 57], with each phase coupled separately to the thermostat with a coupling time constant of 1.0 ps. Pressure control was performed with the Parrinello-Rahman barostat[58] with a reference pressure of 1 bar, a coupling time constant of 12 ps and an isothermal compressibility of $3 \cdot 10^{-4} \text{ bar}^{-1}$. The pressure coupling was set to semi-isotropic and isotropic for systems with and without a bilayer respectively, unless specifically stated otherwise. Cuboid periodic boundary conditions were used in all cases.

5.2.6. Analysis of nanodroplet and fusion simulations

The radius of gyration (R_g) over time of the nanodroplet was determined using the default GROMACS 2016 gyrate tool. Besides looking at the nanodroplet as a whole we used the radial distribution function (RDF) to study the spherical organization of the individual components and their internal chemical groups with respect to the center of mass of the nanodroplet. The RDF was calculated using the GROMACS rdf tool. The chemical groups were specified based on their CG bead types, separating the hydrophobic (tails; C type) and hydrophilic regions (heads, linkers; N, P, Q types).

Mixing of components during fusion was analyzed using a custom VMD script written partially in TCL and python3. First we determined the distances of all components with respect to all the other components (taking into account periodic boundary conditions). All query molecules within 0.8 nm of a target molecule were considered in contact. The contacts were split into two groups: nanodroplet and bilayer. Contacts within one group were normalized by the initial contacts value, where contacts between the droplet and the bilayer were normalized based on their final amount of contacts. The contacts were calculated for the first 1,000 ns of the fusion process.

Visual inspection and image rendering was performed using VMD[59] and graphs were generated using XMGRACE and the python3[60] matplotlib library[61].

5.3. Results

5.3.1. Assembling the nanodroplet

The nanodroplet was assembled in two stages (Fig. 5.1), which mimics the experimental procedure used to produce the lipid nanoemulsion. In stage 1, at an elevated temperature of 350 K, only the vitamins (A/E) and the capric/caprylic acids were added in a dispersed fashion. The components immediately started to aggregate and within 250 ns a stable spherical assembly was formed with a diameter of 5.47 nm (Fig. 5.1 stage 1 R_g). After obtaining a stable stage 1 complex, the other components were added in the same dispersed fashion (Table 5.1, excluding the stage 1 components). The simulation ran until the radius of gyration was stable for at least 100 ns (Fig. 5.1 stage 2 R_g), which took roughly 650 ns resulting in a nanoparticle size of 6.60 nm. After the assembly was completed the whole system was cooled down to 310 K, in line with the manufacturing protocol. Resulting in a slightly smaller final particle with an R_g of 6.50 nm.

Figure 5.1 | Stage 1 and 2 of the nanodroplet self-assembly. The change of the Rg of the nanodroplet components over time for stage 1 and 2 respectively (first row). The self assembly of stage 1 and 2 indicating the final Rg (second and third row respectively).

5.3.2. Characterization of the nanodroplet

To obtain a deeper understanding of the structure of the cooled stage 2 complex, we looked at the relative position of all the components (Fig. 5.2, left graph). In addition, we labeled the hydrophilic and hydrophobic parts of the components (Fig. 5.2, bottom left graph). We reduced the complexity of the figure by splitting the components based on their relative general position separating the core, intermediate and external layer (Fig. 5.2, layers). This splitting was based on the intersections of the hydrophilic heads and the hydrophobic tails curves observed in the head-tail graph (Fig. 5.2).

Based on the radial distributions, vitamin E (more hydrophobic than vitamin A) shows to be highly enriched in the inner core together with the triglycerides (long chain) and cetyl palmitate, whereas vitamin A was distributed both in the core and intermediate layer. The intermediate layer is formed by the phospholipids (in green),

providing support to both the core and the formed layer of vitamin A (in red) and short-tailed triglycerides (in cyan; Fig. 5.2). The surface of the lipid nanoemulsion was coated by the emulsifier steareth 21, with its hydrophilic polyethylene glycol tail mainly lying on the complex surface and its 16C acyl chain anchored in the outer layer of the nanodroplet. All components show a strong preferred orientation as indicated by the large difference between the solid and dashed curves in the RDF graphs, representing the head and tail parts of the constituents, respectively (Fig. 5.2; all graphs on the right).

Figure 5.2 | Nanodroplet structure. The complete stage 2 nanodroplet (first column). The spatial distribution of all components in the nanodroplet, zero indicates the center of mass (top, first column). A snapshot of the final nanodroplet (middle, first column). The spatial distribution of the heads and tails as defined by their CG bead type (bottom, first column). A snapshot of the individual layers (second column). Specification of the specific distribution of heads and tails for each indicated layer (third column).

5.3.3. Fusion with skin membrane models

After obtaining a stable nanodroplet, we constructed two bilayers to model the skin membrane – SC and INF. Both the SC and INF model might be relevant for delivery of active components such as vitamin A/E over the skin barrier. The SC model represents

the rather saturated and rigid membrane of cells in the outer layer of the skin[40, 62]. As the real protonation state of the fatty acids is unknown, we created both a protonated and deprotonated version of the SC model to at least cover the extremes. The INF model mimics the membrane conditions inside a hair follicle and is known to be more soft and flexible in nature due to its higher unsaturation compared to the SC[63, 64]. Equilibrated snapshots for both INF and SC membranes are shown in Fig. S5.1 and S5.2. To simulate fusion, in both models the full nanodroplet was placed in close proximity of the target bilayer (~ 1 nm). The simulation parameters were chosen such that they mimic biological conditions with respect to salt, pressure and temperature (0.15 M NaCl; 1 bar; and 310 K respectively). All fusion simulations spanned 1 μ s and were performed three times except for the large INF system which was only simulated once.

The charged SC fusion experiments ran stable in all cases and no major membrane deformations were observed. Over a period of 1 μ s the nanodroplet adsorbed onto the membrane and the exchange of components between the membrane and nanodroplet was visible by eye (Fig. 5.3). To quantify this mixing we used the change of molecular contacts over the simulation. From the reduction in nanodroplet self-contacts it is clear that steareth 21 mixed the most. Interestingly the other components of the outer layer (lecithin species) did mix less than almost all components in the intermediate layer. Finally the components dominantly present in the core show an extremely low change in contacts. From the decay in bilayer self-contacts we conclude that it is mostly cholesterol which is moving from the bilayer into the nanodroplet. The other components in the bilayer show only very little change ($\sim 10\times$ less than cholesterol; Fig. 5.3). The neutral SC membrane showed a near identical fusion profile, with respect to both the membrane curvature and mixing. Details of said system can be found in the SI.

In contrast to the SC simulations, the INF membranes showed heavy undulation in our fusion experiments. As for the SC membrane models, steareth 21 and cholesterol were the components mixing the most in the nanodroplet and bilayer respectively. In general the INF-nanodroplet mixes more ($\sim 1.5\times$) than the SC-nanodroplet did. However, not all components are mixing more in case of the INF membrane. Cetyl palmitate mixes less, and capric and caprylic acid mix roughly equally. The INF bilayer itself mixed only very little with the nanodroplet except for the cholesterol, showing a similar pattern as the SC membranes.

Due to the fact that the INF membrane showed to be mixing more with the nanodroplet than the SC membrane did, we built larger membrane models of the INF to investigate the effects of size (Fig. 5.4). The large membrane systems behaved very much similar to the smaller patches with respect to the mixing and undulation. Nevertheless, some components do behave differently in the large system. Especially the bilayer self-contacts show a different amount of mixing of the phosphatidylinositol (3,4,5)-triphosphate⁻⁷ (POP3), which mixes about as much as the cholesterol in the large bilayer. In the self-contacts profile for the nanodroplet it appears that GLST is not mixing at all. Inspection of the complete GLST shows that it is mixing but happens to be exactly at the same value at the end as it was at the start of the simulation. However, due to the relatively low occurrence of these components and

the fact that we only ran the large system once, it is difficult to say how significant these changes are. With respect to the mixing kinetics of the nanodroplet we can conclude that the mixing occurs as fast in the large system as in the small system. This indicates that for the initial 1 μ s of fusion the small bilayer is large enough.

Figure 5.3 | Nanodroplet fusion with skin membranes. The fusion of the nanodroplet with the stratum corneum and infundibulum membrane model (first and second column respectively). Snapshots of unbiased fusion at successive moments in time (first row). The change in self-contacts of the nanodroplet between the initial (0 ns) and final frame (1000 ns). The components are sorted by the distance of their peak in the RDF from the COM of the nanodroplet (second row). The change in self-contacts of the bilayer between the initial (0 ns) and final (1000 ns) frame (third row).

Figure 5.4 | Nanodroplet fusion with a large infundibulum membrane. Snapshots of the nanodroplet with the infundibulum membrane model at different points in time during fusion (left). The zoom boxes show the fusion state at 1000ns, illustrating the relatively high amount of cholesterol compared to the other components. The change in self-contacts in the droplet (topright) and bilayer (bottomright) between the initial (0 ns) and final (1000 ns) frame.

5.4. Discussion and Conclusion

Using the same components and procedures described by the nanoemulsion manufacturer we were able to obtain a stable nanodroplet. The nanodroplet contained a fuzzy-layered onion like internal organization, with the vitamins enriched at its core. Most components showed a strong preferred orientation although the inner core was less ordered than the outer layers. To obtain a better understanding of the possible delivery of active components, the nanodroplet was placed on a variety of skin membrane models. On average the nanodroplet components mixed fifty percent more with the INF than the SC bilayers (neutral/charged). However, the increase in mixing showed to be more complex than a single scaling factor. For example, cetyl palmitate actually mixed less with the INF than with the SC. In general the nanodroplet mainly extracted cholesterol from the membranes and only little vitamin A/E entered the target bilayers in the simulated amount of time (1 μ s).

A downside to investigating droplet-membrane fusion in the manner presented in this manuscript is the fact that we only placed one droplet on a periodic bilayer. As the real product is a macroscopic emulsion smeared on top of the skin, it is hard to say what would be the relevant concentration of emulsion per area skin, thus we might be considerably off. Another issue of only simulating one droplet is that we do not know if macroscopic properties such as the critical micelle concentration are actually modelled correctly. In other words adding more droplets into one simulation might cause them to aggregate too much, not forming an emulsion at all, indicating that our model does not behave correctly. Such underestimation of the droplet solvability could very well have a relevant effect on the fusion dynamics. However, to accurately investigate such phenomena as the critical micelle concentration we would have had to simulate spatial and temporal scales not in reach with our computational resources.

As fusion is an out of equilibrium event, it is useful to obtain a better insight into the role of thermodynamic and kinetic effects. We stated that within our time window of simulation little to no vitamins escape the nanodroplet complexation. However, due to the layered organization and the fact that most vitamins are at the core of the droplet, this could be a mere kinetic effect. Diffusion is expected to take place in an inverted radial manner and the vitamins mixing last is to be expected, as they reside at the core of the nanodroplet. Nevertheless, we think there is more than just a kinetic effect which causes the slow exchange of vitamins. The cetyl palmitate partitions similar to the vitamins in the complex and has roughly the same size, nevertheless its diffusion into the bilayer is significant, in particular for SC membranes. Considering capric and caprylic acid, mixing appears to be unaffected by the type of membrane. Such observations indicate that the lack of diffusion of the vitamins into the bilayer might be due to more than just kinetic effects, as such observations clearly do not obey a simple radial distance rule.

Besides a difference in mixing of nanodroplet components between the INF and SC bilayer models, there was a clear difference in membrane undulation. The SC showed little to none bilayer curvature, where the INF showed heavy undulation. This curvature might induce lipid triggered endocytosis by the target cells, allowing the nanodroplet components to enter the cell by other means than passive diffusion. Although our experiments show curvature in a random direction, the asymmetry in

the plasma membrane has been shown to cause such curvature to bend inwards and is strongly dependent on cholesterol depletion[65, 66]. Other means of endocytosis such as clathrin or caveolin dependent pathways are an option as well for the INF[67]. Therefore it seems likely that more active components would cross over the INF than the SC membrane. However, the hairsacks only take up roughly 0.1 % of the surface of the exposed skin, making this path entropically unfavorable[68].

Therefore most of the components will most likely end up in the membranes/extracellular matrix of the SC. In the SC endocytosis is highly unlikely and pointless, for the corneocytes in the SC lack a nucleus, a mitochondria and are regarded dead with respect to most cellular processes[69]. Thus after entering the SC two opposing fluxes are competing with each other. On the one side, passive diffusion will tend to spread out components down the concentration gradient deeper into the skin. Where on the other side, the SC itself is continuously shed and renewed from the base of the epidermis, causing an outward flow. Therefore we think for any component to reach the deeper layers of the skin, the first flux has to be bigger than the second. From our current study we cannot extract the diffusions rates of active components through the extracellular matrix in the SC and therefore no conclusive answer can be given regarding this aspect of uptake.

5.5. Acknowledgments

The authors acknowledge financial support from the Brazilian agencies CAPES, Finep, Fapesp and Boticário Group. Project FINEP 01.10.0661-00, FAPESP 2011/13250-0, FAPESP 2013/17247-9, FAPESP 2014/05975-2, CAPES 88887068264/2014-00, CAPES 88887.068264/2014-00 and CAPES 88881.062862/2014-01. We would like to show our appreciation to Marcos Guilherme Machado and, to the teachers Ph.D. Herculano Martinho of Universidade Federal do ABC (UFABC), São Paulo, Brazil and the Ph.D. Artur Cavaco-Paulo of University of Minho, Braga, Portugal, for collaboration and relevant discussions.

5.6. Supplementary Information

Table S5.1 | Mappings and bead types of nanodroplet components. List of components of the nanodroplet, including molecule name, molecular formula and CG model, including mapping and bead types.

Molecule name	Molecular formula	Mapping in CG representation
Capric acid (short tail)	$C_{30}H_{56}O_6$
Caprylic acid (long tail)	$C_{24}H_{44}O_6$
Hydrogenated Lecithin	$C_{44}H_{86}[NO]_8P$
Cetyl palmitate	$C_{32}H_{64}O_2$
Gliceryl steareth	$C_{21}H_{42}O_4$
Cetyl alcohol	$C_{16}H_{34}O$
Lauryl alcohol	$C_{12}H_{26}O$
Behenyl alcohol	$C_{22}H_{46}O$
Miristyl	$C_{14}H_{30}O$
Palmitic acid	$C_{16}H_{32}O_2$
Stearic acid	$C_{18}H_{36}O_2$

Table S5.2 | Model validation for vitamins and Glycerin. The results of the computational partitioning free energy are compared to experimental values.

Molecule	Chemical structure	CG representation	Experimental $\Delta G_{\text{part octanol/water}}$	CG MD $\Delta G_{\text{part octanol/water}}$
Vitamin A (retinol)	$C_{20}H_{30}O$		32 kJ/mol	39 kJ/mol
Vitamin E (tocopherol)	$C_{29}H_{50}O_2$		71kJ/mol[70]	70 kJ/mol
Glycerin	$C_3H_8O_3$		-10 kJ/mol[71]	-10 kJ/mol

Table S5.3 | Bonded parameters for glycerin and vitamin A/E.

Molecule	Bead ID	Bead type	Bond Value Constant	Angle Value Constant
Glycerin	1	P3	1-2 0.260 1250	.
.	2	P3	.	.
Vitamin A	1	SC2	1-2 0.289 5000	1-3-4 103 25.0
.	2	SC2	1-3 0.339 5000	2-3-4 145 25.0
.	3	SC4	2-3 0.338 5000	3-4-5 093 25.0
.	4	SC4	3-4 0.304 1250	4-5-6 115 25.0
.	5	SC4	4-5 0.403 1250	.
.	6	P2	5-6 0.316 1250	.
Vitamin E	1	SP1	1-2 0.330 5000	1-2-3 052 25.0
.	2	SC4	1-3 0.270 5000	1-3-4 115 25.0
.	3	SC4	2-3 0.270 5000	1-3-5 171 25.0
.	4	SC2	3-4 0.210 5000	2-3-4 168 25.0
.	5	SN0	3-5 0.270 5000	2-4-5 111 25.0
.	6	C1	4-5 0.240 5000	3-4-5 071 25.0
.	7	C1	5-6 0.310 1250	3-5-6 103 25.0
.	8	C1	6-7 0.350 1250	4-5-6 078 25.0
.	9	C1	7-8 0.420 1250	5-6-7 162 25.0
.	.	.	8-9 0.460 1250	6-7-8 157 25.0
.	.	.	.	7-8-9 162 25.0

5

Figure S5.1 | Hair follicle infundibulum membrane composition. The inf membrane based on Machado et al. 2016[40] with the lipids and the permeation pathways. The composition includes 14 different types of compounds (cholesterols, ceramide II NS, sphingomyelins and 9 different types of phospholipids). This membrane is similar to the plasma membrane model made by Ingólfsson et al. 2015[41] and preserves similar characteristics. (a) frontal view of the infundibulum model with the 14 components, among them cholesterol, ceramide, sphingomyelins and various types of phospholipids. (b) Legend of panel a. (c) The possible permeation pathways. The red circle indicates the pathway in which the inf membrane is involved.

Figure S5.2 | Stratum corneum membrane composition. The structure of the stratum corneum based on Machado et al. 2016[40] and the lipid models were taken from Ingólfsson et al. 2015[41]. (a) frontal view of the stratum corneum model with 3 components (cholesterol, fatty acid, ceramide). (b) Legend of panel a. (c) The possible permeation pathways. This membrane represents the “mortar way in a model brick and mortar”, as indicated by the red circle.

5.7. References

- [1] M. Jaiswal, R. Dudhe, and P. K. Sharma, *Nanoemulsion: an advanced mode of drug delivery system*, 3 Biotech **5**, 123 (2014).
- [2] A. Gupta, H. Burak Eral, T. Alan Hatton, and P. S. Doyle, *Nanoemulsions: formation, properties and applications*, Soft Matter **12**, 2826 (2016).
- [3] S. Baboota, F. Shakeel, A. Ahuja, J. Ali, and S. Shafiq, *Design, development and evaluation of novel nanoemulsion formulations for transdermal potential of celecoxib*, Acta Pharm. **57**, 315 (2007).
- [4] C. Qian, E. A. Decker, H. Xiao, and D. J. McClements, *Nanoemulsion delivery systems: Influence of carrier oil on β -carotene bioaccessibility*, Food Chemistry **135**, 1440 (2012).
- [5] O. Sonnevile-Aubrun, J.-T. Simonnet, and F. L'Alloret, *Nanoemulsions: a new vehicle for skincare products*, Advances in Colloid and Interface Science **108-109**, 145 (2004).
- [6] U. Sakulku, O. Nuchuchua, N. Uawongyart, S. Puttipipatkachorn, A. Soottitantawat, and U. Ruktanonchai, *Characterization and mosquito repellent activity of citronella oil nanoemulsion*, International Journal of Pharmaceutics **372**, 105 (2009).
- [7] S. Jain, V. Jain, and S. C. Mahajan, *Lipid based vesicular drug delivery systems*, Advances in Pharmaceutics **2014**, 1 (2014).
- [8] M. M. Fryd and T. G. Mason, *Advanced nanoemulsions*, Annual Review of Physical Chemistry **63**, 493 (2012).
- [9] M. Rad-Malekshahi, K. M. Visscher, J. P. G. L. M. Rodrigues, R. de Vries, W. E. Hennink, M. Baldus, A. M. J. J. Bonvin, E. Mastrobattista, and M. Weingarth, *The supramolecular organization of a peptide-based nanocarrier at high molecular detail*, Journal of the American Chemical Society **137**, 7775 (2015).
- [10] F. Bruxel, J. M. C. Vilela, M. S. Andrade, Â. Malachias, C. A. Perez, R. Magalhães-Paniago, M. C. Oliveira, and H. F. Teixeira, *Investigation of the structural organization of cationic nanoemulsion/antisense oligonucleotide complexes*, Colloids and Surfaces B: Biointerfaces **112**, 530 (2013).
- [11] J. K. Hensel, A. P. Carpenter, R. K. Ciszewski, B. K. Schabes, C. T. Kittredge, F. G. Moore, and G. L. Richmond, *Molecular characterization of water and surfactant AOT at nanoemulsion surfaces*, Proceedings of the National Academy of Sciences **114**, 13351 (2017).
- [12] J. Thewalt and D. P. Tieleman, *Biophysical experiments and simulation in nanoparticle-based drug delivery systems*, Journal of Drug Targeting **24**, 768 (2016).

- [13] M. Ramezanpour, S. S. W. Leung, K. H. Delgado-Magnero, B. Y. M. Bashe, J. Thewalt, and D. P. Tieleman, *Computational and experimental approaches for investigating nanoparticle-based drug delivery systems*, *Biochim. Biophys. Acta* **1858**, 1688 (2016).
- [14] M. B. A. Rahman, Q.-Y. Huan, B. A. Tejo, M. Basri, A. B. Salleh, and R. N. Z. A. Rahman, *Self-assembly formation of palm-based esters nano-emulsion: A molecular dynamics study*, *Chemical Physics Letters* **480**, 220 (2009).
- [15] R. A. Karjiban, M. Basri, M. B. A. Rahman, and A. B. Salleh, *Molecular dynamics simulation of palmitate ester self-assembly with diclofenac*, *Int. J. Mol. Sci.* **13**, 9572 (2012).
- [16] S. Pirhadi and A. Amani, *Molecular dynamics simulation of self-assembly in a nanoemulsion system*, *Chemical Papers* **74**, 2443 (2020).
- [17] S.-J. Lee, P. H. Schlesinger, S. A. Wickline, G. M. Lanza, and N. A. Baker, *Simulation of fusion-mediated nanoemulsion interactions with model lipid bilayers*, *Soft Matter* **8**, 7024 (2012).
- [18] S. Angioletti-Uberti, *Theory, simulations and the design of functionalized nanoparticles for biomedical applications: A soft matter perspective*, *npj Computational Materials* **3** (2017), 10.1038/s41524-017-0050-y.
- [19] H. ming Ding and Y. qiang Ma, *Computational approaches to cell–nanomaterial interactions: keeping balance between therapeutic efficiency and cytotoxicity*, *Nanoscale Horizons* **3**, 6 (2018).
- [20] M. L. Berkowitz, ed., *Biomembrane Simulations* (CRC Press, 2019).
- [21] R. Gupta and B. Rai, *Molecular dynamics simulation study of translocation of fullerene c60 through skin bilayer: effect of concentration on barrier properties*, *Nanoscale* **9**, 4114 (2017).
- [22] J. Wong-Ekkabut, S. Baoukina, W. Triampo, I.-M. Tang, D. P. Tieleman, and L. Monticelli, *Computer simulation study of fullerene translocation through lipid membranes*, *Nature Nanotechnology* **3**, 363 (2008).
- [23] G. Rossi and L. Monticelli, *Gold nanoparticles in model biological membranes: A computational perspective*, *Biochimica et Biophysica Acta (BBA) - Biomembranes* **1858**, 2380 (2016).
- [24] Z. Shen, D. T. Loe, J. K. Awino, M. Kröger, J. L. Rouge, and Y. Li, *Self-assembly of core-polyethylene glycol-lipid shell (CPLS) nanoparticles and their potential as drug delivery vehicles*, *Nanoscale* **8**, 14821 (2016).
- [25] R. Guo, J. Mao, and L.-T. Yan, *Unique dynamical approach of fully wrapping dendrimer-like soft nanoparticles by lipid bilayer membrane*, *ACS Nano* **7**, 10646 (2013).

- [26] S.-J. Lee, P. H. Schlesinger, S. A. Wickline, G. M. Lanza, and N. A. Baker, *Simulation of fusion-mediated nanoemulsion interactions with model lipid bilayers*, *Soft Matter* **8**, 7024 (2012).
- [27] B. M. H. Bruininks, P. C. T. de Souza, and S. J. Marrink, *Methods in Molecular Biology* (Springer New York, 2019) pp. 105–127.
- [28] B. M. H. Bruininks, P. C. T. de Souza, H. Ingolfsson, and S. J. Marrink, *A molecular view on the escape of lipoplexed DNA from the endosome*, *eLife* **9** (2020), 10.7554/elife.52012.
- [29] E. M. Curtis, A. H. Bahrami, T. R. Weikl, and C. K. Hall, *Modeling nanoparticle wrapping or translocation in bilayer membranes*, *Nanoscale* **7**, 14505 (2015).
- [30] M. Lundborg, A. Narangifard, C. L. Wennberg, E. Lindahl, B. Daneholt, and L. Norlén, *Human skin barrier structure and function analyzed by cryo-EM and molecular dynamics simulation*, *Journal of Structural Biology* **203**, 149 (2018).
- [31] M. Lundborg, C. L. Wennberg, A. Narangifard, E. Lindahl, and L. Norlén, *Predicting drug permeability through skin using molecular dynamics simulation*, *Journal of Controlled Release* **283**, 269 (2018).
- [32] T. C. Moore, C. R. Iacovella, A. C. Leonhard, A. L. Bunge, and C. McCabe, *Molecular dynamics simulations of stratum corneum lipid mixtures: A multiscale perspective*, *Biochemical and Biophysical Research Communications* **498**, 313 (2018).
- [33] S. J. Marrink, H. J. Risselada, S. Yefimov, D. P. Tieleman, and A. H. de Vries, *The MARTINI force field: coarse grained model for biomolecular simulations*, *The Journal of Physical Chemistry B* **111**, 7812 (2007).
- [34] T. A. Wassenaar, H. I. Ingólfsson, R. A. Böckmann, D. P. Tieleman, and S. J. Marrink, *Computational lipidomics with insane: A versatile tool for generating custom membranes for molecular simulations*, *Journal of Chemical Theory and Computation* **11**, 2144 (2015).
- [35] C. D. Pizool, personal communication (2017).
- [36] J. A. Bouwstra and M. Ponc, *The skin barrier in healthy and diseased state*, *Biochimica et Biophysica Acta (BBA) - Biomembranes* **1758**, 2080 (2006).
- [37] J. Rogers, C. Harding, A. Mayo, J. Banks, and A. Rawlings, *Stratum corneum lipids: the effect of ageing and the seasons*, *Archives of Dermatological Research* **288**, 765 (1996).
- [38] G. Jenkins, *Molecular mechanisms of skin ageing*, *Mechanisms of Ageing and Development* **123**, 801 (2002).
- [39] M. A. Lampe, A. L. Burlingame, J. Whitney, M. L. Williams, B. E. Brown, E. Roitman, and P. M. Elias, *Human stratum corneum lipids: characterization and regional variations*, *Journal of Lipid Research* **24**, 120 (1983).

- [40] N. Machado, L. dos Santos, B. Carvalho, P. Singh, C. T. Soto, N. Azoia, A. Cavaco-Paulo, A. Martin, and P. Favero, *Assessment of penetration of ascorbyl tetraisoalmitate into biological membranes by molecular dynamics*, Computers in Biology and Medicine **75**, 151 (2016).
- [41] H. I. Ingólfsson, M. N. Melo, F. J. van Eerden, C. Arnarez, C. A. Lopez, T. A. Wassenaar, X. Periole, A. H. de Vries, D. P. Tieleman, and S. J. Marrink, *Lipid organization of the plasma membrane*, Journal of the American Chemical Society **136**, 14554 (2014).
- [42] I. Plasencia, L. Norlén, and L. Bagatolli, *Direct visualization of lipid domains in human skin stratum corneum's lipid membranes: Effect of pH and temperature*, Biophysical Journal **93**, 3142 (2007).
- [43] J.-P. Hachem, D. Crumrine, J. Fluhr, B. E. Brown, K. R. Feingold, and P. M. Elias, *pH directly regulates epidermal permeability barrier homeostasis, and stratum corneum integrity/cohesion*, Journal of Investigative Dermatology **121**, 345 (2003).
- [44] S. J. Marrink, A. H. de Vries, and A. E. Mark, *Coarse grained model for semi-quantitative lipid simulations*, The Journal of Physical Chemistry B **108**, 750 (2004), <https://doi.org/10.1021/jp036508g>.
- [45] G. Rossi, P. F. J. Fuchs, J. Barnoud, and L. Monticelli, *A coarse-grained MARTINI model of polyethylene glycol and of polyoxyethylene alkyl ether surfactants*, The Journal of Physical Chemistry B **116**, 14353 (2012).
- [46] R. Alessandri, P. C. T. de Souza, S. Thallmair, M. N. Melo, A. H. de Vries, and S. J. Marrink, *Pitfalls of the martini model*, Journal of Chemical Theory and Computation **15**, 5448 (2019), PMID: 31498621, <https://doi.org/10.1021/acs.jctc.9b00473>.
- [47] M. N. Melo, H. I. Ingólfsson, and S. J. Marrink, *Parameters for martini sterols and hopanoids based on a virtual-site description*, The Journal of Chemical Physics **143**, 243152 (2015).
- [48] E. Lindahl, B. Hess, and D. van der Spoel, *GROMACS 3.0: a package for molecular simulation and trajectory analysis*, Journal of Molecular Modeling **7**, 306 (2001).
- [49] H. Berendsen, D. van der Spoel, and R. van Drunen, *GROMACS: A message-passing parallel molecular dynamics implementation*, Computer Physics Communications **91**, 43 (1995).
- [50] D. V. D. Spoel, E. Lindahl, B. Hess, G. Groenhof, A. E. Mark, and H. J. C. Berendsen, *GROMACS: Fast, flexible, and free*, Journal of Computational Chemistry **26**, 1701 (2005).

- [51] B. Hess, C. Kutzner, D. van der Spoel, and E. Lindahl, *Gromacs 4: algorithms for highly efficient, load-balanced, and scalable molecular simulation*, Journal of Chemical Theory and Computation **4**, 435 (2008), pMID: 26620784, <https://doi.org/10.1021/ct700301q>.
- [52] S. Pronk, S. Páll, R. Schulz, P. Larsson, P. Bjelkmar, R. Apostolov, M. R. Shirts, J. C. Smith, P. M. Kasson, D. van der Spoel, B. Hess, and E. Lindahl, *GROMACS 4.5: a high-throughput and highly parallel open source molecular simulation toolkit*, Bioinformatics **29**, 845 (2013).
- [53] D. H. de Jong, S. Baoukina, H. I. Ingólfsson, and S. J. Marrink, *Martini straight: Boosting performance using a shorter cutoff and GPUs*, Computer Physics Communications **199**, 1 (2016).
- [54] R. Hockney, S. Goel, and J. Eastwood, *Quiet high-resolution computer models of a plasma*, Journal of Computational Physics **14**, 148 (1974).
- [55] L. Verlet, *Computer "experiments" on classical fluids. i. thermodynamical properties of lennard-jones molecules*, Physical Review **159**, 98 (1967).
- [56] G. Bussi, D. Donadio, and M. Parrinello, *Canonical sampling through velocity rescaling*, The Journal of Chemical Physics **126**, 014101 (2007).
- [57] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola, and J. R. Haak, *Molecular dynamics with coupling to an external bath*, The Journal of Chemical Physics **81**, 3684 (1984).
- [58] M. Parrinello and A. Rahman, *Polymorphic transitions in single crystals: A new molecular dynamics method*, Journal of Applied Physics **52**, 7182 (1981).
- [59] W. Humphrey, A. Dalke, and K. Schulten, *VMD: Visual molecular dynamics*, Journal of Molecular Graphics **14**, 33 (1996).
- [60] C. Sheridan, *The Python Language Reference Manual* (Lulu Press, Inc, 2016).
- [61] J. D. Hunter, *Matplotlib: A 2d graphics environment*, Computing in Science & Engineering **9**, 90 (2007).
- [62] S. J. Marrink, V. Corradi, P. C. de Souza, H. I. Ingólfsson, D. P. Tieleman, and M. S. Sansom, *Computational modeling of realistic cell membranes*, Chemical Reviews **119**, 6184 (2019).
- [63] M. A. Lampe, M. L. Williams, and P. M. Elias, *Human epidermal lipids: characterization and modulations during differentiation*, Journal of Lipid Research **24**, 131 (1983).
- [64] R. B. Gennis, *Biomembranes* (Springer New York, 1989).

- [65] T. Hirama, S. M. Lu, J. G. Kay, M. Maekawa, M. M. Kozlov, S. Grinstein, and G. D. Fairn, *Membrane curvature induced by proximity of anionic phospholipids can initiate endocytosis*, *Nature Communications* **8** (2017), 10.1038/s41467-017-01554-9.
- [66] T. Hirama and G. D. Fairn, *Induction of spontaneous curvature and endocytosis: Unwanted consequences of cholesterol extraction using methyl- β -cyclodextrin*, *Communicative & Integrative Biology* **11**, 1 (2018).
- [67] Y. Fan, Y. Zhang, W. Yokoyama, and J. Yi, *Endocytosis of corn oil-caseinate emulsions in vitro: Impacts of droplet sizes*, *Nanomaterials* **7**, 349 (2017).
- [68] H. Schaefer and T. E. Redelmeier, *Skin Barrier: Principles of Percutaneous Absorption* (S Karger Ag, 1996).
- [69] A. Pouillot, N. Dayan, A. S. Polla, L. L. Polla, and B. S. Polla, *The stratum corneum: a double paradox*, *Journal of Cosmetic Dermatology* **7**, 143 (2008).
- [70] E. Niki, ed., *Vitamin E* (Royal Society of Chemistry, 2019).
- [71] *Physical and thermodynamic properties of pure chemicals: data compilation*, (1990).

6

Sequential voxel-based leaflet segmentation of complex lipid morphologies

The computing scientist's main challenge is not to get confused by the complexities of his own making.

Edsger W. Dijkstra

As molecular dynamics simulations increase in complexity, new analysis tools are necessary to facilitate interpreting the results. Moving from a few particles to billions, results in many emerging properties. Lipids, for instance, are known to form many complicated morphologies due to their amphipathic nature, becoming more intricate as particle count goes up. A few lipids might form a micelle, where aggregation of tens of thousands could lead to vesicle formation. Millions of lipids make up a cell and its organelle membranes, and are involved in processes such as neurotransmission and transfection. To study such phenomena, it is useful to have analysis tools which understand what is meant by emerging entities such as micelles and vesicles. Studying such systems at the particle level only, becomes extremely tedious, counterintuitive and computationally expensive. To address this issue we developed a method to track all the individual lipid leaflets, allowing for easy and quick detection of topological changes at the meso scale. By using a voxel-based approach and focussing on locality, we forgo costly geometrical operations without losing important details and chronologically identify the lipid segments using the Jaccard Index. Thus we achieve a consistent sequential segmentation on a

This chapter has been made in collaboration with Albert Thie, Paulo C. T. de Souza, Tsjerk A. Wassenaar, Shirin Faraji and Siewert J. Marrink

wide variety of (lipid) systems, including monolayers, bilayers, vesicles, inverted hexagonal phases, etc. It also discriminates between adhesion and fusion of leaflets. We show that our method produces consistent results without the need for pre fitting parameters, and segmentation of millions of particles can be achieved on a desktop machine.

Keywords: molecular dynamics (MD), consistent sequential segmentation, visualization, leaflet identification, membranes, Martini

6.1. Introduction

The increase in complexity of molecular dynamics (MD) simulations poses challenges for current analysis software. This is well illustrated in the field of biological applications. Such simulations changed from a single protein in vacuum[1] in the late 1970s to complex membrane based systems containing millions of molecules such as (phospho)lipids, cholesterol, proteins, DNA, drugs, water, etc[2–7]. The number of molecular aggregates in simulations has also increased, and nowadays it is not rare to see a mixture of micelles, bilayers and vesicles in a single simulation[4, 8–11]. This increase in complexity of both composition and aggregation state requires a higher level of abstraction in analysis than typically offered. During analysis we would like to work with concepts such as bilayers, vesicles, leaflets, protein aggregates, phases, etc. on top of simple identifiers such as molecule names and indices, often used in current tools. This would dramatically reduce the complexity of coding the analysis software and would allow for a more user friendly interface with data. Current MD analysis software, such as MDAnalysis[12], Pytim[13], FATSLIM[14] and VMD[15], come with built-in functions or have packages to detect e.g. bilayer leaflets and aggregates, and offer some of the desired level of abstraction. However, these tools were developed to deal with relatively simple compositions and topologies, and may fail as systems contain more than one or two segments. On top of this, they rely heavily on expensive geometrical operations, resulting in intrinsically slower execution. To avoid these problems and develop a fast and robust method for consistent sequential segmentation, we explore a voxel oriented analysis paradigm. No assumptions are required regarding the number of segments or molecules. Furthermore, voxel-based methods offer very good scaling with the increase of particles and are excellent for locality queries[16–19].

To guide the reader through this work we will first present pseudo code to illustrate the general implementation of our leaflet segmentation algorithm, which is implemented in a tool coined MDVoxelSegmentation. We then show a wide range of examples, and we end with a discussion of the limitations and further prospects of the tool.

6.2. Design and implementation

In the following sections, we describe the complete algorithm in two sections of pseudo code. We start with considering a single frame and perform our multi-step connective components analysis for lipid-based systems. Then we describe how we add the extra layer of chronological consistency on top.

6.2.1. Pseudo code for spatial segmentation of leaflets

To successfully segment lipid densities into leaflets we require three things, a reference, a trajectory and a selection file. The reference file should contain information regarding atoms and their naming (e.g. GRO, PDB). The trajectory file should contain coordinates for each point in the reference file in every frame (e.g. GRO, XTC, TRR). Finally, the selection file should contain the selections (lipid heads, lipid tails, lipid linkers and exclusions). By default most CG Martini lipids are supported[20], meaning no selection has to be specified for the leaflet segmentation of those systems. However, if you are working on a custom lipid, it can easily be added to the default selections file. After segmentation each lipid is assigned a segment identity which is a residue level property. To make MDVoxelSegmentation a complete segmentation tool, we also support basic connected components analysis on non-amphipathic systems.

Read input

The leaflet segmentation starts with mapping all the lipid tail, lipid head, and exclusion beads to a boolean voxel matrix (usually 0.5 nm binning for Martini lipid systems; Fig. 6.1 1). This results in three 3D boolean matrices: tails, heads and exclusions. The exclusions are used as local stops during segmentation. For CG systems hyper-resolution is often advised, which places virtual densities around the point positions to allow more smooth mapping and preventing empty voxels to be present in a continuous density.

Connected components for tails

The tails matrix is segmented using the adapted connected components algorithm, subtracting the heads and using the exclusions for halting (Fig. 6.1 3). Usually the exclusions are grown once, which guarantees the exclusions act up to at least one time the binning distance (Fig. 6.1 2). We found that using exclusions can be beneficial when working with proteins, since the protein boundaries are prone to be edge cases.

Connected components for heads of each tail segment

Roughly the same procedure is followed for segmenting the heads as was done for the tails. However, instead of using the complete heads matrix at once, it is first deconstructed by the tail segments. Each tail segment is individually processed. Only the headgroups which have a lipid residue index which is also in the active tail segment are considered (Fig. 6.1 4 and 5). The connected components segmentation is performed for each of the resulting selections (Fig. 6.1 6 and 7). For the heads segmentation, all tails are used as a subtraction mask and exclusions are used for halting.

Figure 6.1 | Spatial leaflet segmentation. (1) Input data. Unsegmented lipid headgroups, water and protein are purple, blue and pink respectively. (2) Lipid tail density (green) and headgroup occlusion masking (red). A white block indicates the exclusion zone around the protein. (3) Lipid tail segmentation of the upper and lower bilayer (orange and grey). (4) Headgroups density (green) with tail occlusion (red) of tails segment I. (5) Headgroups masking of tails segment II. (6) Headgroups segmentation of tails segment I (orange and navy blue). (7) Headgroups segmentation of tails segment II (light yellow, grey). (8) Backmapping to point cloud labels. (9) Polishing of the result using iterative force-segmentation, no unsegmented lipids remain in this example.

Forced iterative segmentation

Although the result at this point is good (Fig. 6.1 8), there are some lipids which still have no segment assigned. For every lipid which does not have a segment assigned, all neighbouring lipids are checked within a certain cutoff distance (Fig. 6.1 9). If the dominant segment in that search is not 0 (unsegmented), the lipid is appointed to that dominant segment. Our implementation performs this operation in an iterative manner always trying to satisfy small distances first. The iterations halts if the force-segmentation returns a stable output (i.e. no further changes occur). Splitting the segmentation in two steps, using force-segmentation as a post processing step, means that simple cases are handled in a relatively low-cost manner where hard

cases might use a different approach altogether. This mixed use of a low-cost algorithm with more precise and expensive algorithms reduces the total complexity of the calculation drastically, whilst at the same time hardly affecting the segmentation quality.

Spatial segmentation output

The result after spatial leaflet segmentation is a trajectory.npy which contains a segmentation entry for each index in each frame. These indexes correspond to the atom indices in the original structure file. Atoms which have no segmentation are assigned to 0. However, due to the arbitrary picking of our initial voxels for segmentation, the segmentation labels miss sensible consistency over time for all the other labels. To solve this lack of chronological consistency, we need to perform temporal segmentation.

6.2.2. Pseudo code for temporal segmentation

All that is required as input for the temporal segmentation is the spatial segmentation output.

Matching

To perform temporal segmentation, we need to find the identity of segments for each frame sequentially. When comparing a new frame with a current frame, the algorithm first considers the identities of segments of the current frame. The identified segments of the current frame are then matched to the segments in the new frame (Fig. 6.2 A). The atoms within a segment are used as the basis for the identity of the segment. Segments are treated as sets containing unique elements. The identity of a segment in the new frame is the segment in the current frame with which it shares the most atoms. In this way segments can be matched across frames, even if due to shrinking or growth their atom contents are not identical. Consider frame N which contains two segments $N1 = a1, a2, a3, a4$ and $N2 = b1, b2, b3, b4$ and frame M which contains $M1 = a1, a2, a3, b4$ and $M2 = b1, b2, b3, a4, c1$. When we compare frame N with frame M, we see that the elements $b4$ and $a4$ have swapped places and $c1$ is introduced to $M2$. However we still want to assign $N1$ to $M1$ and $N2$ to $M2$, to allow for the growing and shrinking of segments. We therefore need a way to identify the amount of shared elements between two segments.

$$J(A, B) = \frac{A \cap B}{A \cup B} \quad (6.1)$$

To identify how many atoms are shared between two segments across frames the Jaccard index is used (eq. 6.1). This index is defined as the intersection of two sets divided by the union of two sets. This provides a ratio between 0 and 1, where 1 is complete similarity and 0 means no shared atoms. The parameter identity threshold defines the minimum score on the Jaccard index that is required for two sets to have the same identity. For example a minimum Jaccard score of 0.75 means that only if two segments have a score higher than 0.75 together, they have the same identity. This allows for consistency while also identifying large shifts

between frames in the simulation. Because of the possibility of these large shifts the algorithm does not assume that the amount of segments between frames remains the same. It therefore allows for the disappearance of identified segments and the appearance of new identities. Identified segments can disappear when no match in the new frame can be made that exceeds the minimum Jaccard score. An identified segment disappearing is considered to have merged with another segment. A new identity can be created when in the new frame a segment exists to which no old identified segment has matched. This new segment then receives a new identity. A newly created identified segment is considered to have split from a previously existing identified segment.

Figure 6.2 | Finding the identity of the segments in the new frame. (A) The ordered 1 segment has a best match with unordered 2. (B) The ordered 1 and 2 are merged into one segment, with ordered 2 providing the identity for unordered 1. (C) The ordered 2 is split into two new labels, unordered 1 and 2. Ordered 0 is always matched as the new segment zero, as this is the segment which contains all non-segmented particles. Capital N indicates the total number of segments in each frame.

6

Merge

In the case of a merge (Fig. 6.2 B), two or more identified segments in the current frame are merged into an unidentified segment in the new frame. Two distinct cases are possible in the new frame. Either one of the identities of the merging segments is inherited by the new segment or a new identity is created. If one of the identities is maintained, the Jaccard index score between one of the identified segments and the unidentified segment has exceeded the minimum score. This indicates a large segment is merging with one or more smaller segments. If a new identity is created, this means none of the previous segments contained enough atoms to attain a Jaccard score higher than the minimum when compared to the unidentified segment. In this case all previous identified segments are merged into a new segment with a new identity.

To allow for the reappearance of identities, the disappeared identities are stored in a database. This database contains the identity and the atoms present in the identified segment at the current frame. It also logs where into which segment the disappeared segment has merged. The segment into which the disappeared segment has merged is the segment for which it had the highest Jaccard index score. This score will always be below the Jaccard minimum, as otherwise the identity would be inherited. This information can be used to restore identities when splitting events take place.

Split

In a split, two or more unidentified segments in the new frame have the highest Jaccard index score with the same identified segment in the current frame (Fig. 6.2 C). In such a split there are also two distinct cases. In one case a small part splits off from an existing identified segment. However one of the unidentified segments still has a high enough Jaccard score to match with the identified segment. This means that one of the unidentified segments inherits the identity of the identified segment. The other unidentified segments will then either gain a new identity or take the identity of an old segment from the database. New segments are compared with the database. The new segment takes the identity of the segment with the highest Jaccard index score in the database, but only if the score exceeds the minimum Jaccard score. If the minimum is not exceeded, the segment is assigned a new identity.

In the other case the identified segment could not match with one of the unidentified segments. This will lead to the disappearance of the identified segment. The disappearing segment will be stored in the database, while the new segments will each be compared to the database. Each segment which has a match with the database will continue with the identity of this match, while the unmatched segments continue with new unique identities.

Temporal segmentation output

The result after temporal segmentation is a consistent sequential segmentation over the trajectory, both spatially and over time. Identities between time frames are consistent, based on the atoms within each segment. Merging segments are detected and identified in a size dependent way. Splitting segments are detected as well. The database of previous segments allows identities to persist over a longer time frame.

6.3. Results

To validate the leaflet segmentation algorithm, we prepared a wide range of lipid systems using the Martini CG force field[21]. The same default input was used for all segmentation unless specifically stated otherwise. We start with simple lipid bilayer systems in varying topologies (Fig. 6.3). Then we add proteins and a high variety of compositional complexity (Fig. 6.4). This is followed by a complex phase transition and fusion of DNA-lipid complexes (lipoplexes) with endosome membrane models to illustrate segmentation quality in the presence of tight, curved and complex dynamic geometry (Fig. 6.5). We end with some more general real world examples (Fig. 6.6).

Pure bilayer systems

Leaflet assignment for a simple bilayer containing 270 DOPC and 66 cholesterol (4:1) without force-segmentation resulted in good segmentation, but some lipids, in particular cholesterol, are unlabeled (Fig. 6.3 A, C orange). By turning on force-segmentation all lipids including the cholesterol could be assigned successfully to a leaflet (Fig. 6.3 B, C green).

A common lipid property to investigate is the flip-flop rate. We compared the assigned flip-flopping of our leaflet segmentation to a common procedure. This

common analysis would assign a lipid flip-flop based on the z-coordinate of a lipid headgroup with respect to the average z-position of the bilayer. The downside is that this can only be performed for small patches of a relatively flat bilayer with its normal pointing along the z dimension. Our method takes a rather different approach. We check for changes in leaflet membership for each lipid. To reduce noise, a lipid which is not segmented is assigned to its previous valid leaflet. The advantage of our segmentation based method over the conventional approach is that it still works if the membranes/leaflets are curved, or if they are not a well defined bilayer at all. In our comparison of cholesterol flip-flopping, our algorithm performs well which is indicated by the close relation between the segmentation labels and the discretization of the lipid z-height (Fig. 6.3 C). Segmentation without force-segmentation is good enough for lipid flip-flop calculations. The segmentation based flip-flop analysis compares well with the z-height approach if a deadzone of ± 0.6 nm is used in the latter (Fig. 6.3 D). Although the flip-flop patterns for both methods are highly similar, they are not the same, therefore, direct quantitative comparison between both methods should be performed with care. However, if a z-height deadzone of ± 0.3 nm is used, all flip-flops indicated by the segmentation analysis are in the set of the z-height approach. In other words it appears that the flip-flops indicated by the segmentation are a subset of the z-height approach with a relatively small deadzone.

Since stacked bilayers are notoriously hard to label correctly in other methods, we designed a test system containing two closely stacked and intercalating bilayers of POPC (1-3 CG water beads, Fig. 6.3 E). The close proximity and or intercalation of the bilayers did not hinder correct leaflet segmentation at all over a trajectory spanning 10 μ s (Fig. 6.3 F).

The final pure bilayer system tested is a bilayer in which we artificially opened and closed a pore using a biasing potential. We used the leaflet segmentation to detect the instances of pore opening and closing (Fig. 6.3 G and H).

Consistent leaflet segmentation was successfully obtained using the default settings for all pure bilayer systems even in the presence of cholesterol or intercalating headgroups. The leaflet segmentation can also be used for flip-flop analysis and the detection of a toroidal pore with high accuracy.

Complex bilayers with proteins

After confirming that our leaflet segmentation performs well on simple lipid-only bilayer systems, we went one step further and considered systems of higher compositional complexity. As for the lipid only segmentation, segmentation of a simple bilayer containing POPC lipids and a single protein is perfect using the default settings (Fig. 6.4 A). Next, we tested a crowded plasma membrane model. This model contains a high variety of lipids (>60) and a wide range of proteins in a planar membrane. Segmentation resulted in a nearly perfect assignment (Fig. 6.4 B). However there are 12 lipids which remain unlabeled (Fig. 6.4 B zoom box), caused by an exclusion region. Setting the force-segmentation cutoff radius slightly higher resolved the issue, but the current result is shown to demonstrate the quality of the default settings. Missing 12 lipids out of tens of thousands of lipids would probably not affect the analysis to any serious degree. However, if perfection is required some tweaking

Figure 6.3 | Leaflet segmentation and analysis for pure bilayer systems. (A, B) Snapshots of a DOPC/CHOL(4:1) bilayer with and without force-segmentation. Some lipids were unassigned without force-segmentation due to diving/occlusion (A, yellow). Using force-segmentation non-segmented lipids are assigned to a segment as a post-processing step (B). (C) The flip-flop traces of a single cholesterol in the DOPC/CHOL bilayer, comparing the z-height and segmentation flip-flop approaches. Whenever the orange line is not visible it lies perfectly behind the green line. (D) A comparison of the up- and downward cholesterol flip-flop events between the z-height and segmentation based flip-flop analyses. A more intense single dot represents more than one cholesterol flip-flopping at that given instant. The deadzone for the z-height analysis was ± 0.6 nm. (E) A snapshot of closely stacked and intercalating bilayers of DLPC. (F) Leaflet segmentation of the intercalating bilayers over 10 μ s. The width of a segment represents its relative size (G) A snapshot of a DLPC bilayer with a toroidal pore (frame 750 of H). (H) Leaflet segmentation of pore formation. The pore forming biasing potential was turned on between frame 500 and 1000. The width of a segment represents its relative size.

might be needed. For example the force-segmentation range or the binning distance could be altered as well as the minimum cluster size.

Based on these results we conclude that MDVoxelSegmentation works well on systems with a complex composition even in the presence of protein(s).

Figure 6.4 | Leaflet segmentation in the presence of protein. (A) Adding protein (lime) to a bilayer does not interfere with the leaflet assignment of a CG POPC bilayer (red/blue). A video of the bilayer protein system can be found in section 6.7.2. (B) The bottom and top leaflet of the plasma membrane. Even extremely complex bilayers such as the plasma membrane including many different proteins (transparent/lime) and lipids (red/blue) do not hinder leaflet assignment. However, there is a small cluster of non-assigned lipids (yellow) around one of the proteins (zoom box).

Complex, curved and tight geometry

Up to now all our test systems were relatively flat, therefore as our final test set, we introduce heavy curvature as well as closely packed segmentation. In our previous work we showed the phase transition of a stacked bilayer system to an inverted hexagonal phase (H_{II})[4, 22], which was used as a system which should push the time consistency part of our algorithm, as well as the agonistic behaviour with respect to the amount of segments. A snapshot of the first intermediate and last frame of the trajectory is shown in Fig. 6.5 A. The corresponding graph shows that the segmentation is extremely stable over time. A video of the full phase transition is available in section 6.7.3. During transition the following happens: First the leaflets which share a common water reservoir (Fig. 6.5 A I) fuse together by multiple stalk formation (Fig. 6.5 A II). The leaflets are then split into smaller segments, more closely resembling the classic view of the H_{II} phase. However, due to the high concentration of water in this system not all aquatic channels are singular and some remain connected through water pores perpendicular to the view normal (Fig. 6.5 A III). The final result is a stable H_{II} phase of channels with a varying degree of interconnection. Note that the visualization of the segments immediately makes it clear which channels are connected and which not.

In our previous work we showed how transfection with lipoplexes can be simulated using the Martini force field[4]. Such lipoplexes are complexes of lipids and dsDNA. The dsDNA resides inside the inverted hexagonal phase of the lipoplex lipids. Upon fusion with a membrane the dsDNA is released into the cytosol. The tight packing of the lipids with the dsDNA in combination with the change in topology during transfection, makes this another challenging system to analyse. A render of the leaflet segmentation is shown in Fig. 6.5 B. A video of the complete transfection process is available in section 6.7.4. Although the segmentation is pretty good, it does contain some noise, illustrated by minor flickering in the video. This noise could be removed with a noise filter, but we left it in for fair comparison and to create a realistic reference for the quality of the output under comparable conditions. As for pores in the bilayer, we can use the amount of leaflet segments identified in the simulation as a reliable metric for fusion events (Fig. 6.5 B, graph). We did find that decreasing the resolution of the leaflet segmentation for complex geometry resulted in a steep decrease in quality of segmentation. For more simple geometry 1 nm voxels without hyper-resolution resulted in acceptable segmentation. However, for these systems 0.5 nm voxels with hyper-resolution is required.

The presence of (high) curvature does not hinder correct leaflet segmentation, even if the lipid arrangement does not follow a strict bilayer definition or if the amount of segments is highly dynamic.

Further tests

The leaflet segmentation was tested on three additional complex systems to obtain familiarity with the quality. The first system contains a small fatty acid vesicle (Fig. 6.4 and section 6.7.5). Around the vesicle fatty acids are suspended in a water buffer. The goal was to measure the imbalance of molecules in the inner and outer leaflet of the vesicle as it absorbs the surrounding fatty acids in solution. As the outer leaflet

Figure 6.5 | Leaflet segmentation and analysis for both tight and curved complex geometry.

(A) The phase transition from a multilamellar bilayer system to an H_{II} phase. Snapshots of the first, intermediate and last segmented frame of a 1 μ s trajectory. The accessory video is available in section 6.7.3. The corresponding graph shows all segments formed in chronological order with the snapshots indicated with the dotted lines. The width of a segment indicates its relative size. (B) Snapshots of a lipoplex transfecting dsDNA over a bilayer (first, intermediate and last segmented frame). The inner core (orange, yellow, green, gray) is an intact H_{II} phase containing the dsDNA (not visible), whereas the upper bilayer leaflet (red) has already fused to the outer leaflet of the lipoplex by means of a fusion stalk (I). During transfection the inner core segments fuse with the bottom leaflet of the bilayer (II), resulting in a single flat bilayer with all the dsDNA on the other side (III). The transfection can be tracked in detail using the segmentation graph. The width of a segment indicates its relative size.

of the vesicle merges with more and more small micelles in solution the imbalance increases, potentially resulting in division. Some flip-flopping occurs to release the tension due to the imbalance between the inner and outer leaflet. However, there are two moments at which the inner and outer leaflet of the vesicle temporarily fuse by means of a toroidal pore. These pores quickly cause a significant amount of acyl chains to migrate from the outer to the inner leaflet, indicated by a jump in the inner leaflet's size (Fig. 6.6 A). Passive flip-flop occurs all the time and is indicated by the more continuous increase in size of the inner leaflet. For division to occur, the rate of absorption of fatty acids from solution has to be increased to outcompete the flip-flop. This example furthermore illustrates the use of our segmentation tool to track flip-flop and pore formation events with high resolution.

The second test was to assign the leaflets in the showcase mitochondrion model presented in the work by Pezeshkian et al.[10]. For the mitochondrion we expected to observe four separated leaflets, namely the separate leaflets of both the inner and outer membrane. However, after leaflet segmentation of the complete mitochondrion we only observed three segments. To further investigate this issue we cut the mitochondria in twenty slices. Each piece was segmented individually (Fig. 6.6 B). This revealed that there are issues in two of the twenty slices. When these two problematic slices are excluded the expected four leaflets were found by segmentation analysis. This indicates unexpected overlap is occurring in those two slices, likely as a result of stalk formation between closely apposed membrane segments. Further subdivision could be utilized to pinpoint the exact location, although doing so by hand becomes rather cumbersome. Nevertheless this example shows that large structures can be successfully handled by our algorithm and the results can be used for quality control of systems too large for visual inspection by traditional means.

Finally, in our previous publication on lipoplexes we described the existence of connective channels in an overly hydrated inverted hexagonal lipid phase, shown in the example above on formation of an H_{II} phase (Fig. 6.5 A)[4]. These channels probably have huge effects on the fusion kinetics of such phases. Related work was performed at atomistic scale by Ramezanzpur *et al.*, in which they did not describe any of such channels[8]. After contacting the group and confronting them with this paradox, we were kindly allowed access to their data. We used MDVoxelSegmentation to find any possible unexpected and undescribed connective channels. The channels were confirmed to also exist in the atomistic simulations, mostly forming at the mid to end of their highly hydrated systems (Fig. 6.6 C). The channel was stable, but difficult to spot using the conventional tools available. This due to the fact that it spanned over the PBC, a common problem in periodic systems. Checking the marked frames by eye confirmed that the results of MDVoxelSegmentation were correct, the authors agreed after reevaluating their data. The formation of these pores was not observed at lower temperatures, suggesting that there is a kinetic factor and that their simulations might benefit from extension.

Together, these examples show that MDVoxelSegmentation is a fast and useful tool for spotting complex segmentation properties, otherwise easily missed or extremely cumbersome to analyse.

Figure 6.6 | Additional systems. (A) A snapshot of a fatty acid vesicle (red/grey), surrounded by individual fatty acids in the medium (green). The amount of fatty acids in the inner and outer leaflet of the vesicle over time (graph). A video of the fatty acid dinner is available in section 6.7.5. (B) A snapshot of the segmented mitochondrion (~83 million beads). Four distinct leaflets can be traced (green, yellow, purple, red) if the intersection zones are not included in segmentation (white). If the white zones are included in the segmentation, the pink and red leaflets are fused together. (C) Snapshots of the atomistic POPC system in the H_{II} phase as published in [8]. Segmentation analysis reveals previously unnoticed connections between the initial four channels at the end of the simulation. The width of a segment in the graph indicates its relative size.

6.3.1. Performance

To get some insight in the performance of the current MDVoxelSegmentation tool we timed the segmentation for some of the test systems using a varying amount of threads. In all these tests we turned off force-segmentation, for it can have a large impact on performance based on the exact system (especially at higher maximum recursion depth). The results are summarized in Fig. 6.7. From the single thread values we can deduce that our algorithm indeed scales pretty much linearly with some variance due to packing and complexity (table 6.1). Systems containing thousands to millions of particles of interest will roughly take between less than seconds to minutes per frame respectively. The biggest system we timed was the 83 million particle model of a mitochondrion 10, which took over 4 hours for a single frame on a single thread. The system of the mitochondrion was the only system which could not be handled with 16GB of RAM and needed a specialized node to run. Its memory consumption peaked at 150GB.

Figure 6.7 | Single frame scalability. The time it takes to segment a single frame with increasing particle count. Exact data is represented in table 6.1. Some variation from linearity is expected due to the fact that the specific geometry of a system can have an impact on the complexity.

Table 6.1 | Performance of MDVoxelSegmentation. All systems were run on the same Ubuntu desktop except for the mitochondrion. The test machine contained a Ryzen 1600 (6 cores, hyper threading 3.8 GHz) CPU and 16GB of DDR4 memory (2x 8GB 2.133 MHz). No GPU acceleration is currently implemented. The MDVoxelSegmentation parameters were the same as stated in the examples above except for force-segmentation, which was turned off (-fs 0). Timing was performed using the default Ubuntu time functionality. The CPU for the mitochondrion system ran at 2.4 GHz for a single thread.

System	Particles	Frames	Threads	Real time	User time	Sec/Frame
Bilayer pore	10,436	1,500	12	2m25s	21m41s	0.096
Fatty acid growth	66,278	1,000	12	3m35s	27m37s	0.215
Bilayer pore	10,436	1,500	1	13m13s	15m23s	0.529
Fatty acid growth	66,278	1,000	1	16m20s	18m30s	0.980
Plasma membrane	1,343,450	1	1	4m35s	4m36s	275
Mitochondrion	83,288,300	1	1	260m18s	-	15,618

6.4. Discussion

6.4.1. Future development of MDVoxelSegmentation

In this manuscript we show the use of voxel based approaches for analysis of lipid and surfactant based systems. However, our implementation is not optimal yet. What we hope to achieve is to stimulate developers of MD software which have a large user base (MDAnalysis[12], VMD[15], Pymol[23], UnityMol[24]) to embrace voxels and add user intuitive voxel APIs with common components analysis to their packages. VMD already has a well optimized grid density function, which could be made more accessible for users, or even contain some GUI. The VMD code could probably perform all operations in this manuscript at runtime for systems of considerable size due to their efficient usage of GPU(s). How the segmentation API could best be merged into the current selection syntax is not trivial, and will require careful consideration.

6.4.2. Potential extra features

As shown here, leaflet segmentation can be solved with high fidelity, using only local checks, and chronological segmentation. We showed accurate segmentation of a wide range of lipid systems with or without the presence of cholesterol, protein and or (extreme) curvature. Although the focus of this manuscript was lipid leaflet detection, we would like to point at some possible additional features that further broaden the scope of the applications. One of those features is the tracking of membrane pores and stalks. This should use the same basic voxel operations such as erosion, growth and common neighbor segmentation, but results in tracking abstract segments such as pores. This would allow us to treat pores as segments, propagating them through the trajectory using the same selection syntax and data structures. Current attempts show to be promising, although this pore segmentation appears to rely on a good understanding of subdividing the segmentation results. Some preliminary attempts of the porefinding can be found in the MDVoxelSegmentation github pore branches.

Another possible application is in the analysis of interfaces and the manner in which they percolate between set targets. In this case the interphase itself might be the entity tracked over time using a distance query on multiple segments.

Detecting lipid lateral phase separation is another problem which we think could

be potentially tackled with our tool. In this case we should not map lipid densities to voxels, but lipid compositions. In the voxeled compositions a 3D edge detection algorithm could be used to find transitions in local composition. Since we know that a phase its composition should be constant in the bulk region, we can start with finding volumes which are constant in composition by taking the derivative. This would work similar to the work of Sodt et al. where they demonstrate such approach on a 2D grid[25]. Probably some form of compositional smoothing or dimensionality reduction would be required to reduce noise in the spatial composition of complex systems.

Finally, the analysis is generic enough to be used practically without adaptations on lipid related systems such as amphipathic proteins or fatty acids (section 6.7.6, 6.7.5), thus showing promise to be used as a more general procedure for amphipathic systems.

6.4.3. How to contribute

The MDVoxelSegmentation package will be expanded by our own attempts to tackle problems within this framework. We do invite interested researchers to contribute to the further development of the tool. To add your own segmentation routines to the package, simply open a ticket at our github and indicate your interest.

(<https://github.com/marrink-lab/MDVoxelSegmentation>)

6

6.4.4. To conclude

We showed that consistent and accurate leaflet segmentation can be obtained with our sequential segmentation algorithm. The algorithm proved to be successful on a wide range of lipid and lipid related systems. These systems contained both high compositional and topological complexity which are not analysable with the currently available methods. To achieve this we used a voxel data structure, which allows for fast local queries, segmenting each frame individually. To achieve consistent segmentation over multiple frames, this was followed by set theory, making use of the Jaccard index. In our tests we show many examples of CG Martini lipid systems, mostly performed with the same parameter settings. These settings show very few artefacts indicating that in many cases no time needs to be spent on parameter tweaking. Besides CG systems we also looked at an atomistic system for which segmentation showed to be of high quality with only minor changes needed to the default settings.

Our algorithm practically scales linearly with the increase in particles and is able to handle millions of particles in minutes on a desktop machine. However, performance gains of orders of magnitude could probably be achieved by proper SIMD/GPU implementation[26] and careful memory management using more advanced voxel data structures[16–18]. The resulting segmentation is easy to work with in python or VMD and can directly be used for visualization due to its supported segmentation data format.

6.5. Methods

6.5.1. Molecular dynamics simulations

To simulate the lipid only systems we made use of GROMACS 2019.1[27]. Martini 2 was used as the force field[21]. All lipid parameters were taken from cgmartini.nl (2019) and initial configurations were generated using insane[20]. Each simulation contained 0.15 M NaCl and 10% antifreeze (WF) to prevent unnatural freezing of the water as described in the original Martini paper[21]. Rectangular periodic boundary conditions were employed to prevent artificial wall effects (xy 10 nm). The steepest descent algorithm was used for energy minimization (1,000 steps) and equilibration was performed using the default Martini settings with a time step of 2 fs for 25.000 steps[21, 28]. The Verlet cut-off scheme was used with a 1.1 cut-off for both the coulombic (reaction-field) and van der Waals interactions. The v-rescale scheme (tau-p=1 ps) was used for the thermostat at 310 K, coupling the lipids and solvent/ions in two separate groups. Pressure coupling during equilibration was performed using the Berendsen barostat for semi-anisotropic systems (tau-p=3 ps, compressibility=3e-4 bar-1, ref-p=1.0 bar in each dimension)[29]. The production run made use of a 20 fs time step and the pressure coupling was switched to Parrinello-Rahman (tau-p=12)[30].

To simulate pores in the DLPC bilayer, positional restraints were used in the form of a cylindrical biasing potential acting on the C1A and C1B tail beads (GROMACS potential function 2, shape 8, hole radius -1.2 nm, force constant 1000 kJ mol⁻¹). The negative sign in front of the hole radius indicates an inversion of the biasing potential in GROMACS. A reference file was created with all beads at the same position, acting as the pore center reference position (-r flag in GROMACS). The pore formation was simulated by running a production run for 10 ns without the biasing potential, then the potential was turned on for another 10 ns and turned off again for the final 10 ns.

6.5.2. Leaflet segmentation

We used the same default settings for all the presented segmentation unless specified differently. These settings used a binning resolution of 0.5 nm and made use of hyper-resolution of 0.5. The headgroups, linkers, tails and exclusions were as specified in the default selection input. The minimum size of a segment was 50 particles and iterative force-segmentation was turned on with a maximum distance of 20 Å. An identity threshold of 0.75 was used for the Jaccard index. All keyword arguments represent default values which do not need to be specified except for the reference frame and trajectory file (any MDAnalysis supported MD format).

```
mdvseg -f start.gro -x trajectory.xtc
--resolution 0.5 --hyper_resolution 0.5 --recursion_depth 10
--force_segmentation 20 --force_information False
--minimum_size 50 --begin 0 --end None --stride 1 --threads 12
--bit_size 32 --output clusters --verbose False
```

For the segmentation of the atomistic H_{II} system hyper-resolution was turned off

(`--force_segmentation 0`). In general hyper-resolution is not required for atomistic systems. The respective headgroup tails and linkers selection were:

```
"name N P C12 C11 O11 O12 O13 O14" (heads)
"name C1 C2 O21 C21 C3 O31 C31" (tails)
"name C22 C23 C24 C25 C26 C27 C28 C29 C210 C211 C212 C213
C214 C215 C216 C217 C218 C32 C33 C34 C35 C36 C37 C38 C39
C310 C311 C312 C313 C314 C315 C316 C317 C318" (linkers)
```

6.5.3. Lipid flip-flop

The average z-height of the cholesterol headgroups was subtracted from the head-group z-position. This resulted in headgroups lying either above or below zero. Headgroups with a normalized z-position less than 0.6 nm away from 0 were labeled either up or down (discretization of the z-height). Differentiating the individual lipid traces results in peaks at flip-flop events. These peaks were counted, registering if the flip occurred up- or downwards. This calculation was repeated for all cholesterol in all frames and a graph was made using python3 matplotlib[31]. The z-height and flip-flop analysis is available at the MDVoxelSegmentation github.

To show the flip-flopping of cholesterol using our leaflet segmentation we used the default settings except for the force-segmentation which was turned off. Lipids which were not assigned to segment 0 were assigned to their previous segment. A graph was made using matplotlib and the plotting script included with mdvseg. The leaflet flip-flop analysis is available at the MDVoxelSegmentation github.

6

6.5.4. Visualization

All figures and videos were rendered using VMD and a little TCL script of MDVoxelSegmentation to load the segmentation array. Examples of TCL scripts are provided together with the code at:

<https://github.com/marrink-lab/MDVoxelSegmentation>

6.6. Acknowledgements

Bart M. H. Bruininks would like to thank Manuel N. Melo, Jonathan Barnoud, Weria Pezeshkian, John Stone and the rest of the VMD developers, as well as Elmar Eisemann for the many fruitful discussions. We would also like to thank Melanie König (fatty acids), Haleh Abdizadeh (plasma membrane), Josef Melcr (amphipathic peptides), Weria Pezeshkian (mitochondrion) and Peter Tieleman (atomistic H_{II}) for sharing their data.

6.7. Supplementary Information

6.7.1. Bilayer pore

bilayer_pore The opening and closing of an artificial pore in a DLPC bilayer. Right before the stable toroidal pore is formed, a short lived water pore spans the bilayer. This pore was visualized using the following selection syntax in VMD: '(name W) and (within 20 of name PO4 and user 1) and (within 20 name PO4 and user 2)'. User 1 and 2 are the respective leaflets. This selection syntax could be used for the input of connected neighbours segmentation on its own. This brings the advantage that we can check if any of such water densities bridges the two leaflets. Thus we can visualize water pores connecting two leaflets and segment them. However, we lose all information of the pore as soon as the leaflets become continuous (as illustrated in the video).

6.7.2. Protein bilayer

protein_bilayer Stable segmentation of a bilayer in the presence of a protein. Close to the protein some lipids can be observed which are diving relatively deep under the surface, however, this diving is handled without missegmentation.

6.7.3. Lipid phase transition

phase_transition Lipid segmentation of a phase transition from dehydrated stacked bilayers to inverted hexagonal (left). Water densities (right).

6.7.4. Lipoplex transfection

transfection A lipoplex was placed on top of a model endosomal membrane. The simulation was started in the presence of a fusion stalk, connecting the outer leaflet of the lipoplex coat with the upper leaflet of the endosomal membrane (red, transparent). Over time the internal compartments of the lipoplex (orange, yellow, pink, green) fuse with the bottom endosomal leaflet (red, transparent). At the end of the simulation all dsDNA from inside the lipoplex was transfected over the endosomal membrane. Some flickering is observed, but overall the segmentation is very stable and flickering could easily be removed with a noise filter.

6.7.5. Acyl chain dinner

acyl_chains, acyl_chains_dof In an attempt to understand early life 'cell' division we placed a small vesicle (acyl chains) in the presence of food (also acyl chains). The vesicle adds the food from its surroundings to its own composition, but without internal reorganization the food is only added to the outer leaflet. With a large enough leaflet disbalance three things can happen to restore the balance. The acyl chains flip-flop from the outside to the inside. The leaflets temporarily fuse to balance the amount of acyl chains in the inner and outer leaflet, or the disbalance causes the vesicle to divide. MDVoxelSegmentation was used to analyse the imbalance as well as to monitor the feeding rate. Both flip-flopping and toroidal pore formation takes place. No divisions occur.

6.7.6. Segmentation of other amphipathic systems

`amphipathic_peptides` With trivial adaptations of the input GRO file we used our method to perform segmentation of amphipathic peptides. Due to their amphipathic nature, micelles from such peptides can touch but not fuse (like lipid leaflets). MD-VoxelSegmentation shows to be suited to be used for such systems.

6.7.7. Self assembly of a DPPC bilayer

`bilayer_self-assembly` For the self-assembly of this bilayer only the linkers were selected in the headgroups. This has to be done due to the toxic PBC conditions and its effect on the topology of the system. This is illustrated by the fact that the intermediate cubic phase is only one continuous leaflet. Something which would not have been the case if it was built in a PBC with an even number of bilayers.

6.8. References

- [1] J. A. McCammon, B. R. Gelin, and M. Karplus, *Dynamics of folded proteins*, Nature **267**, 585 (1977).
- [2] S. J. Marrink, V. Corradi, P. C. de Souza, H. I. Ingólfsson, D. P. Tieleman, and M. S. Sansom, *Computational modeling of realistic cell membranes*, Chemical Reviews **119**, 6184 (2019).
- [3] F. J. van Eerden, D. H. de Jong, A. H. de Vries, T. A. Wassenaar, and S. J. Marrink, *Characterization of thylakoid lipid membranes from cyanobacteria and higher plants by molecular dynamics simulations*, Biochimica et Biophysica Acta (BBA) - Biomembranes **1848**, 1319 (2015).
- [4] B. M. H. Bruininks, P. C. T. de Souza, H. Ingólfsson, and S. J. Marrink, *A molecular view on the escape of lipoplexed DNA from the endosome*, eLife **9** (2020), 10.7554/elife.52012.
- [5] M. Vögele, R. M. Bhaskara, E. Mulvihill, K. van Pee, Özkan Yıldiz, W. Kühlbrandt, D. J. Müller, and G. Hummer, *Membrane perforation by the pore-forming toxin pneumolysin*, Proceedings of the National Academy of Sciences **116**, 13352 (2019).
- [6] H. I. Ingólfsson, P. Tieleman, and S. Marrink, *Lipid organization of the plasma membrane*, Biophysical Journal **108**, 358a (2015).
- [7] J. R. Perilla, J. A. Hadden, B. C. Goh, C. G. Mayne, and K. Schulten, *All-atom molecular dynamics of virus capsids as drug targets*, The Journal of Physical Chemistry Letters **7**, 1836 (2016).
- [8] M. Ramezanpour, M. L. Schmidt, B. Y. M. Bashe, J. R. Pruim, M. L. Link, P. R. Cullis, P. E. Harper, J. L. Thewalt, and D. P. Tieleman, *Structural properties of inverted hexagonal phase: A hybrid computational and experimental approach*, Langmuir **36**, 6668 (2020).
- [9] D. A. Holdbrook, R. G. Huber, T. J. Piggot, P. J. Bond, and S. Khalid, *Dynamics of crowded vesicles: Local and global responses to membrane composition*, PLOS ONE **11**, e0156963 (2016).
- [10] W. Pezeshkian, M. König, T. A. Wassenaar, and S. J. Marrink, *Backmapping triangulated surfaces to coarse-grained membrane models*, Nature Communications **11** (2020), 10.1038/s41467-020-16094-y.
- [11] H. Alasiri, *Determining critical micelle concentrations of surfactants based on viscosity calculations from coarse-grained molecular dynamics simulations*, Energy & Fuels **33**, 2408 (2019).
- [12] R. Gowers, M. Linke, J. Barnoud, T. Reddy, M. Melo, S. Seyler, J. Domański, D. Dotson, S. Buchoux, I. Kenney, and O. Beckstein, *MDAnalysis: A python package for the rapid analysis of molecular dynamics simulations*, in *Proceedings of the 15th Python in Science Conference (SciPy)*, (2016).

- [13] M. Sega, G. Hantal, B. Fábián, and P. Jedlovsky, *Pytim: A python package for the interfacial analysis of molecular simulations*, *Journal of Computational Chemistry* **39**, 2118 (2018).
- [14] S. Buchoux, *FATSLiM: a fast and robust software to analyze MD simulations of membranes*, *Bioinformatics* **33**, 133 (2016).
- [15] W. Humphrey, A. Dalke, and K. Schulten, *VMD: Visual molecular dynamics*, *Journal of Molecular Graphics* **14**, 33 (1996).
- [16] B. Dado, T. R. Kol, P. Bauszat, J.-M. Thiery, and E. Eisemann, *Geometry and attribute compression for voxel scenes*, *Computer Graphics Forum* **35**, 397 (2016).
- [17] A. J. Villanueva, F. Marton, and E. Gobbetti, *SSVDAGs*, in *Proceedings of the 20th ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (ACM, 2016).
- [18] V. Careil, M. Billeter, and E. Eisemann, *Interactively modifying compressed sparse voxel representations*, *Computer Graphics Forum* **39**, 111 (2020).
- [19] M. Sualeh and G.-W. Kim, *Dynamic multi-LiDAR based multiple object detection and tracking*, *Sensors* **19**, 1474 (2019).
- [20] T. A. Wassenaar, H. I. Ingólfsson, R. A. Böckmann, D. P. Tieleman, and S. J. Marrink, *Computational lipidomics with insane: A versatile tool for generating custom membranes for molecular simulations*, *Journal of Chemical Theory and Computation* **11**, 2144 (2015).
- [21] S. J. Marrink, H. J. Risselada, S. Yefimov, D. P. Tieleman, and A. H. de Vries, *The MARTINI force field: coarse grained model for biomolecular simulations*, *The Journal of Physical Chemistry B* **111**, 7812 (2007).
- [22] B. M. H. Bruininks, P. C. T. de Souza, and S. J. Marrink, *A practical view of the martini force field*, in *Methods in Molecular Biology* (Springer New York, 2019) pp. 105–127.
- [23] S. LLC, *The pymol molecular graphics system*, (2020).
- [24] S. Doutreligne, T. Cragolini, S. Pasquali, P. Derreumaux, and M. Baaden, *UnityMol: Interactive scientific visualization for integrative biology*, in *2014 IEEE 4th Symposium on Large Data Analysis and Visualization (LDAV)* (IEEE, 2014).
- [25] A. J. Sodt, M. L. Sandar, K. Gawrisch, R. W. Pastor, and E. Lyman, *The molecular structure of the liquid-ordered phase of lipid bilayers*, *Journal of the American Chemical Society* **136**, 725 (2014).
- [26] L. Cabaret, L. Lacassagne, and L. Oudni, *A review of world's fastest connected component labeling algorithms: Speed and energy estimation*, in *Proceedings of the 2014 Conference on Design and Architectures for Signal and Image Processing* (IEEE, 2014).

- [27] M. J. Abraham, T. Murtola, R. Schulz, S. Páll, J. C. Smith, B. Hess, and E. Lindahl, *GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers*, *SoftwareX* **1-2**, 19 (2015).
- [28] D. H. de Jong, S. Baoukina, H. I. Ingólfsson, and S. J. Marrink, *Martini straight: Boosting performance using a shorter cutoff and GPUs*, *Computer Physics Communications* **199**, 1 (2016).
- [29] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola, and J. R. Haak, *Molecular dynamics with coupling to an external bath*, *The Journal of Chemical Physics* **81**, 3684 (1984).
- [30] M. Parrinello and A. Rahman, *Polymorphic transitions in single crystals: A new molecular dynamics method*, *Journal of Applied Physics* **52**, 7182 (1981).
- [31] J. D. Hunter, *Matplotlib: A 2d graphics environment*, *Computing in Science & Engineering* **9**, 90 (2007).

7

Summary and Outlook

7.1. What has been done

We started with a description of the Martini force field and presented a hands-on section to allow anyone with a desktop to perform molecular dynamics simulations. This is important for the force field will dictate how our system evolves over time. In our brief review on the Martini force field we explain how Martini can be used to access timescales and systems sizes out of reach for all atom force fields such as CHARMM. We also reflected on the high variety of biologically relevant molecules which are available in Martini. This allows for biological systems of high compositional complexity to be modelled. However, Martini 2 is not perfect and it comes with some fundamental flaws which were discussed in the special notes section. Some molecules, like proteins, are too sticky and special beads are required to prevent water from freezing above its physical freezing point. Other flaws, more general in coarse-grained systems, are the (over)simplification of electrostatic interactions and the loss of entropy due to the reduced particle count. Nevertheless, Martini shows to be capable to produce molecular dynamics models of high quality and is one of the most used coarse-grained force fields. As a matter of fact, more and more scientific work is published using Martini. Therefore if we want to simulate parts off, or a complete minimal cell, with molecular detail Martini seems to be a suited force field.

Besides from being able to propagate our molecular models, we need to have an initial state to start from. As described in the preface, cells consist of trillions of atoms. Since the molecular models presented in this thesis require a position and velocity for all these atoms (or groups of atoms in the case of the Martini force field), constructing the initial state by hand would be near impossible and a huge waste of human resources. Therefore an automated procedure for building biological systems is required. Part of this issue of creating the initial state has been addressed in chapter 2 where we presented a method for building basic lipid bilayer geometries of heterogeneous composition. Our work is not the first to focus on tackling this issue and certainly will not be the last. Bilayers are but one macroscopic assembly

present in cells and we need to be able to build them all (e.g. scaffolding proteins, DNA packing, the cytoplasm, etc.).

To explore what will be possible and what will be the new problems we might encounter as we scale up our molecular models, two studies were performed. First we presented the construction and simulation of lipoplexes which we fused with endosomal bilayers. These systems were stable and showed fusion under the by experiments described conditions. Giving a first insight into the molecular mechanisms involved in transfection, an important step in gene therapy. The Martini force field showed to be sensitive to the subtle difference in both the lipoplex and endosome formulations, as is also indicated by literature. Second, we used the Martini force field to simulate the delivery of insoluble compounds to the skin through the means of nanodroplets. We succeeded in constructing nanodroplets of high compositional complexity and fused them with two different skin membranes. In the case of the stratum corneum the lipiddroplet is absorbed without much disturbance of the membrane geometry. In the contrary, fusion of the nanodroplet with the infundibulum membrane resulted in heavy membrane undulation. Cholesterol also appears to behave rather different than any other component in both skin bilayer models. Both these studies show that the Martini force field can be utilized to model systems of complex composition whilst still being computable within a reasonable amount of time. During these two studies we found that both constructing the initial states, and the analysis of said systems becomes much more challenging as the systems tend to become bigger and more complex in composition.

In the last part of this thesis we presented a tool for the analysis of complex lipid systems. By using macroscopic labels such as bilayers and leaflets, the analysis of a wide variety of lipid systems is made considerably less time consuming. The labels are created using a topological description, using connectivity, instead of geometry. This causes the analysis to be rotation and translation invariant making it rather robust. The tool presented showed to be able to correctly label lipid assemblies of complex morphology and composition and track those labels over time. The performance is reasonable well and analysis results can be expected at near real time for many systems currently investigated. The algorithm scales well within the range of tested systems, but further optimization in both the memory consumption and utilization of accelerating hardware will be required for it to be used on simulations at the scale of cells.

7

7.2. What is to come

In this thesis we started with a quick consideration of the (im)possible with respect to modeling nature with molecular precision. We concluded that simulating a full human is out of scope, however, parts of life such as cells and their organelles are within reach. Nevertheless, one should keep in mind that there is a substantial difference between the theoretically possible and practically achievable. Therefore we will give our view on what is to come in the next years with respect to simulating the minimal cell.

If we want to model a (minimal) cell we will have to be careful with the details included in our model. If we make it too detailed we might end up with a model that

is exact but incomputable. In the contrary, if we make our model to coarse we might be able to compute it, but the results could have lost their physical meaning. As of yet we are not aware of any theory which can predict the level of detail required to study a phenomenon. This means that we cannot be certain that the resolution we choose for our model of the cell will be the minimum resolution required for useful results. Even worse, depending on the phenomena we will probably have to change the amount of detail in our models. Therefore we think it will be paramount to be able to easily change the resolution of our models. By easily we mean that a coherent framework is required to use an existing model and its results, and convert it into a best first guess of a model with a different resolution. For example, a complete systems at atomistic detail has to be convertible to a complete systems at coarse-grained resolution and coarse-grained models need to be convertible into continuum models. The other way around should also be possible, although there is a fundamental lack of degrees of freedom which will probably have to be estimated. Besides being able to change resolution, it will be important to change model at the same resolution. Most force fields excel at a specific task and it would be very useful to be able to use the best force field for the task. Although this is currently mostly possible for force fields of atomistic resolution, the same can not be said for coarse-grained models, even if the resolution of the coarse graining is very comparable. If conversion of resolution and model can be achieved at a whim, we can start building a model of the cell using a continuum/coarse-grained approach and swap to a higher resolution, or a different model, depending on the question at hand. In our opinion Martini could play an important role in being an intermediate force field which preserves molecular detail whilst being much more robust than their atomistic counterparts.

The collaboration between experimental and theoretical structural microbiologists needs to be fostered. In the past decades many molecular dynamics simulations made use of the steadily growing databases of (bio)molecular structures such as the protein data bank. In many cases starting from a configuration close to the biologically relevant configuration saves a considerable amount of computational resources. This allows to push our models to go beyond what is achievable by using theory and predictions only. The same will be true when we will work on modeling cellular structures. High resolution 3D scans of different complete microorganisms will be required to serve as the blueprint. These scans can be converted into continuum models which should be convertible all the way down to the required level of accuracy. We will not only require information on the composition of certain cellular compartments such as the different bilayers and their sub-leaflets, but also positional information on where those compositions occur in these cellular structures. For a cell is not in equilibrium and for our results, of the minimal cell, to be useful it will become more and more important to understand where and how much the cell deviates from equilibrium properties.

Chemistry and non-equilibrium physics have to be added to our models at all resolutions. Most molecular dynamics simulations are performed under equilibrium conditions without chemistry or flux in composition. This is partially because many of the studied phenomena are at, or close, to equilibrium. However, we also study such phenomena because our theoretical framework of phenomena close to equi-

librium is most developed. Especially the grand canonical ensemble in which the amount of matter is not constant is notoriously hard to sample/calculate efficiently. This is partially a (bio)informatics problem, but also due to a more fundamental lack of understanding of out of equilibrium systems. Still, at the atomistic and coarse-grained level chemistry will probably be very similar to grand canonical simulations as a reaction can be considered a function which removes the reactants and adds the products. As the product and their amount does not have to match the reactants, a varying amount of particles has to be allowed (albeit of the same total mass). Therefore fast and accurate simulation and analysis software has to be developed to handle the grand canonical ensemble which we could use to simulate chemistry.

Handling and understanding large amounts of data will be fundamental in our effort of simulating a minimal cell. We already reached a point where the raw amount of data easily surpasses the amount of data a human can understand and work with. Therefore dimensionality reduction will probably become one of the most important and challenging parts of simulating and understanding our models of the minimal cell. Nowadays the field of artificial intelligence appears to be the way to go if one wants to convert bits into the objects we all love and appreciate. And, maybe even more important, we have a huge amount of knowledge at the level of objects which we could use to make our models at the molecular level even better (top-down information).

All in all we think that simulations of eukaryotic organelles or simple prokaryotes should not only be theoretically possible, but also achievable in the coming decade and we cannot wait to contribute to, and appreciate the wondrous models of the microscopic that are yet to come.

Acknowledgements

I would like to thank the following people for their support during my (scientific) development.

Family I am raised in a scientific family in which critical and creative thinking is always encouraged. My mother Lies, you have a very keen mind, a big heart and always try to make sure my education does not cause harm to my personal well being. My dad Ab, we had countless conversations about nearly everything in the world and I hope we will have countless more. My sister Lotte, you taught me to be, you are so much like me, and yet so different. You show me where ambition can get you and taught me how to fight for my beliefs. My brother in law Jelle, you are awesome and thanks to you I started reading Pratchett. My two nieces Ella and Mare, you make me smile when I think of all the trouble you are going to cause. The love of my life Joske, will you marry me?

I also fondly remember all the birthdays with tricky math riddles presented by my uncle Wim. The riddles sometimes got tackled, giving everyone involved the feeling of being the smartest person in the (multi/uni)verse. Especially the concepts of distributed knowledge and emergent properties – which were usually wrapped up in leprechauns with red and blue hats – were a recipe for riddles which I recall as being the toughest and most intriguing of them all. I would like to thank my other uncle Gertjan for solving one of the hardest sliding puzzles I ever encountered in my life. You saved me from countless of sleepless nights. You and Ria served as my part time mom and dad during a large part of my youth. Which brings me to my cousins Jochem, Sybren and Nnaniki. There were periods of my life in which I would spend as much time as possible with you on the farm inventing all kind of crazy devices and constructions or just baking tons of ‘wentelteefjes’. The sky is the limit and I think we poked a good hole in it. I would also like to thank my uncle Ton for his refreshing view on doing what you enjoy, I hope I will keep doing the same.

Friends If one can judge a man by the company he keeps, I consider myself lucky. There are some friends I would like to thank in particular for they have been with me practically since it all began. Lars, my brother. Kaj, my second brother. Elwin, my worst enemy and best friend. Wouter, Rock Lee. Tim, MIDI/music demigod. Ruben, ‘meisje in de tuin. Je loopt een beetje schuin’. Tristan, I do not think you should be better in chess than me. Petran, unforgettable, relaxed and pragmatic. Bram, party starter XXL deluxe. Sybe, no joke untold. Roald, actually living the life and catching sick (shell)fish. Giel, he who appears even later than me. Aad, actually a nice person. Max, my muse. Dorine, you *saved* me with your intervention. Albert, my wall. Jan, we share our love for pigeons. Thomas, DM extraordinaire. Jantien, we share a

decade of coffee by now. Paulo, the best colleague a friend can be. If you are a friend and I forgot to mention you, please add yourself below, sorry for the inconvenience.

Name	Description
Colleagues A working place can make all the difference. Paulo, my mentor and scientific guide. Siewert-Jan, you gave me all the space and help required, when needed. Manel, a remarkably skilled scientist and friend (you can add your name above). Helgi, the reason I fell in love with lipids. Ignacio, I always hoped you would be there for a short conversation. Alex, a very precise scientist with an astonishing amount of ears available. Weria, if it curves talk to this guy. Haleh, you were always very open and I enjoyed working together in the scientific image competition. Jonathan and Peter, you were there to make me a programmer. Tsjerk, I loved the sparring on geometry and in the end we even wrote a script together. Bea and Anmara, without you I would have been so lost. I would also like to thank all other members/staff at the MD group, I hope the good vibes stay and that many more will enjoy their time in your presence.	

Special Thanks I would like to thank Melle for his immaculate effort in proofreading my thesis.