Universität Bielefeld

Dissertation

Computer simulations of lipid bilayers

Dipl.-Inform. Olaf Lenz

December 4, 2007

Dissertationsschrift zur Erlangung des Doktorgrades der Naturwissenschaften (Dr. rer. nat.) an der Technischen Fakultät der Universität Bielefeld

Contents

List of Figures

1. Introduction

1.1. Computer simulations in soft-matter research

A broad range of materials that we interact with in our daily lives are neither solid nor gaseous, but *soft matter*, like liquids, polymers, liquid crystals, biological tissues or colloidal suspensions, and they are the subject of *soft-matter research*. From a physical point of view, what is common to all of these materials is that the binding energies of the molecules within the material are relatively small and comparable to the thermal energies.

Because the bonds involved in soft matter are usually of a molecular nature, and furthermore the major part of all biological tissues can be considered as soft, soft-matter science is a highly interdisciplinary research field, where physicists cooperate with biologist and chemists.

The various phenomena that occur in soft matter live of a multitude of length and time scales, ranging over several orders of magnitude from subnanometers to centimeters and from picoseconds to minutes. For physics research, these mesoscopic length and time scales are hard to access both experimentally and theoretically. Therefore, computer simulations are a crucial tool for their understanding, and they are sometimes even considered as the "third pillar" of physics research next to theory and experiment.

1.2. Biomembranes

Biomembranes play an important role in all biological cells. They subdivide the cell into different compartments (the organelles), and they separate the cell's interior from its exterior (the cell wall). While a biomembrane is almost completely impermeable for water and other molecules, it is extremely flexible and cannot easily be destroyed by mechanical stress. However, a membrane is not just a passive wall, but it plays an active role in the transport of particular molecules or information from one side to the other [Gen89].

The modern view of the molecular structure of a biomembrane is described by the *dynamically structured mosaic model* [VSM+03], a descendant of the well-known *Singer-Nicholson fluid-mosaic model* [SN72] (see figure 1.1 on the next page). In both models, the bulk of a biomembrane is formed by a bilayer of lipids (mainly phospholipids). A bilayer consists of two layers of amphiphilic lipid molecules with their hydrophobic tails pointed to the inside and the hydrophilic heads exposed to the watery environment. The bilayer mainly serves as the passive element of the membrane, as it

is flexible and at the same time impermeable for water and water-soluble molecules. Membrane proteins and other molecules are distributed on the membrane. These proteins, including receptors, enzymes and ion channels, perform most of the membran's active functions, like the transport of molecules across the membrane. Although the different membranes in a cell have a very different lipid and protein composition (proteins make up 20 to 80 per cent of the membrane's weight), the basic structure is the same.

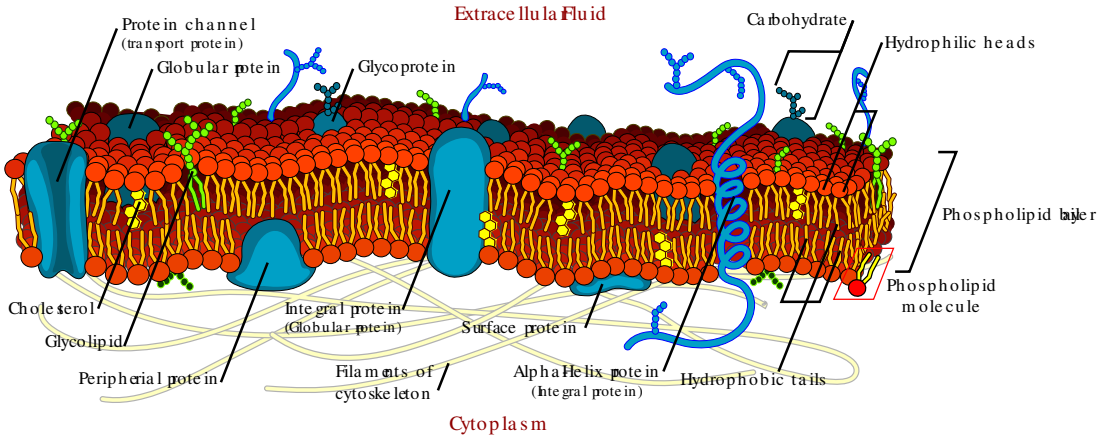


Figure 1.1: Singer-Nicholson fluid mosaic model of a membrane. Drawing provided by M. R. Villarreal [1].

The bilayer component of a biomembrane normally is in the fluid phase, and can be envisioned as a two-dimensional liquid. The *fluidity* of the membrane is the central paradigm of the fluid-mosaic model. It is crucial for the mobility of the different lipids, proteins and other molecules in the plane of the membrane. This mobility allows the lipids and proteins to distribute on the whole membrane area by diffusion. However, the proteins and lipids are not homogenuously distributed on the membrane.

During the last decades, it has become evident that the membrane itself is structured into subdomains with very different lipid and protein composition. Domains with a high concentration of particular lipids (so-called *rafts*) interchange with protein-rich domains and clusters of specific proteins, such as *receptor islands*. In the dynamically structured mosaic model, the *mosaicity* of the membrane is presumed to be at least as important for the membrane function as is the fluidity. However, neither its exact meaning for the membrane function, nor the mechanisms and dynamics used by the cell to control the domain formation have been completely understood by now.

[1] `http://commons.wikimedia.org/wiki/Image:Cell_membrane_detailed_diagram.svg` (November 11, 2007)

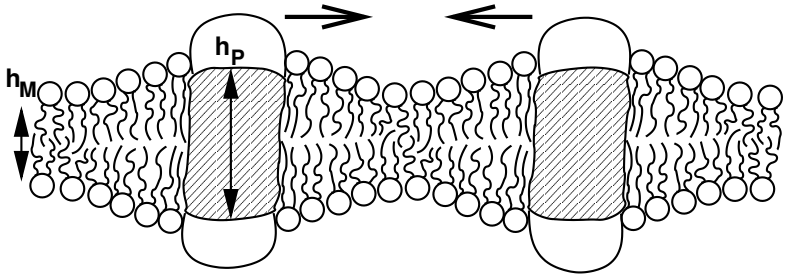


Figure 1.2: Hydrophobic mismatch

1.2.1. Meaning of the mosaicity

Apparently, mosaicity may serve a number of functions.

On the one hand, clusters of proteins may influence the function of these proteins. For example, in the cell wall membrane of nerve cells, potassium ion channels assemble in clusters. This allows them to collectively open or close, thus shaping the sharp rise in the action potentials of nerve impulses [NWV06].

On the other hand, subdomains with different lipid and protein composition have fundamentally different physical properties such as the flexibility, fluidity and curvature. This may also affect the function of the cell membrane. A good example is the so-called *domain induced budding* [Lip02]: When a bilayer is made up from two lipid types that do not mix very well, the bilayer will phase separate, forming two domains that contain the one or the other type. Under these circumstances, the line tension at the domain boundary will eventually dorce one domain to form a bud that can separate from the membrane. Such buds are essential for transportation purposes in the cell.

1.2.2. Controlling the mosaicity

What is still poorly understood is, how the cell controls this domain formation. How do proteins and lipids interact to form *e.g.* receptor islands or clusters of ion channels? Electrostatic interactions are very much screened by the ions in the biological environment, chemical signals are not feasible on the molecular level, and direct molecular interactions are very short-ranged. Therefore, it has been argued, that a significant contribution to these interactions are indirect interactions mediated by the lipids [LZR98].

An example for these *lipid-mediated interactions* is the *hydrophobic mismatch* (see figure 1.2). If a membrane integral protein has a hydrophobic thickness h_P that is greater than the hydrophobic core thickness h_M of the bilayer, the lipids close to the protein need to stretch, as otherwise the hydrophobic amino acids of the protein are exposed to the hydrophilic head groups of the lipids or even the watery environment outside. This loss of entropic freedom can be counteracted by the system by reducing the protein-lipid interface, *e.g.* by grouping several such proteins together. This results

in an attractive effective interaction between the proteins.

Another poorly understood problem is the detailed effect of the lipid composition on the domain structure of the bilayer. For both of these effect, the understanding of the lipid behaviour is crucial.

The lipid behaviour strongly depends on the lipid bilayer phase. Usually, the lipids in the bilayer are in a fluid state with disordered tails and a high in-plane mobility of lipids and membrane-integral proteins. It is interesting to note, however, that the temperature of the so-called *main transition* is close to the body temperature for most of the membrane lipids, where the system goes over to the so-called gel phase with ordered lipid tails and a much lower in-plane mobility. The proximity of this transition supposedly has a profound effect on both the domain structure of the lipids and the lipid-mediated interaction of proteins. The phase behaviour of the bilayer, however, is strongly influenced by the bilayer's lipid composition.

Apparently the key factor to the controlling of the domain formation and consequently of some aspects of the membrane function is the lipid composition. Lipids are relatively simple molecules and can be produced by the cell with less effort than proteins, using the lipid composition seems to be an elegant way for the cell to control the membrane function.

1.2.3. Computer simulations of biomembranes

All of these phenomena occur on mesoscopic length scales of a few tens of nanometers or more. This makes it difficult to approach the problem both experimentally and theoretically. Although in recent years there have been many improvements in experimental techniques, this length scale is still difficult to access [MJ97]. On the other hand, for analytical theory, most of the systems and effects involved are too complex.

Therefore, a good method to investigate these phenomena are computer simulations. Despite the fast progress in this area, so far the power of even the largest supercomputer is not enough to do systematic all-atom simulations of large lipid bilayer chunks in watery environment. Instead, *coarse-grained* simulations of the system can be used. In a coarse-grained model, the number of the degrees of freedom and the number of model parameters of the system are reduced to yield shorter simulation times.

Biomembrane models are to be found on many different levels of coarse-graining, from almost atomistic models that only reduce the degrees of freedom of the hydrogen atoms up to models that consider the membrane as a two-dimensional fluctuating surface. Each of the models can be used to answer specific questions.

1.3. This work

The agenda of this work is two-fold: on the one hand, it will describe the tools that have been developed in the frame of this work for the soft-matter research community from the point of view of computer science. Chapter 2 on page 15 will describe

the "SoftSimWiki", a collaborative platform using a wiki-software that supports rapid knowledge exchange for the community of soft-matter researchers on methodology and should thus fill the gap that is left open by the classical scientific publications. Furthermore, a toolbox of C++-classes, the "Off-Lattice Template Library" OLTL, has been developed and will be presented in chapter 3 on page 29.

The second goal of this work was the development of a coarse-grained computer model of a lipid bilayer that can be employed to approach the problems described above, namely the influence of the phase behaviour of a lipid bilayer on the lipid-mediated protein-protein interaction.

To enable the reader to understand lipid bilayer systems, chapter 4 will introduce the reader to the chemistry, properties and phase behaviour of lipid bilayers. In chapter 5, a coarse-grained bead-spring model for a lipid bilayer and its solvent environment is developed. The parameters of the simulations done and the observables that were measured are detailed in chapter 6. The results that were obtained on the phase behaviour of the model lipid bilayer are given in chapter 7. In the course of the simulations, an interesting structure has been found to emerge. The structure has been identified with the *ripple phase* of lipid bilayers. This has led to the understanding of the molecular structure of the ripple phase (chapter 8) that has been an open question for the last 30 years.

2. SoftSimWiki

This chapter will present the SoftSimWiki, a collaborative online resource for soft-matter methodology.

During the last decades, the number of available computational methods in soft-matter research is rapidly increasing and the methods and their implementation are getting more and more complex. Unfortunately, classical scientific publications do not greatly support researchers on questions of the methodology (section 2.1). On the other hand, more and more collaborative platforms occur on the internet, that allow many people to efficiently exchange information on many subjects (section 2.2). Therefore, SoftSimWiki, a new collaborative platform is proposed (section 2.3), that should enable scientists to exchange knowledge and information on the methodology of soft matter research.

2.1. Simulation methods in soft-matter research

Due the rapid development of computer hard- and software, more and more computing power has become available for researchers in theoretical physics, and computer simulations and related computational methods have become more and more common and widely used, up to the point that simulations are sometimes considered as the "third pillar" of physics research next to theory and experiment.

The phenomena studied in soft-matter science occur on a multitude of length and time scales, ranging over several orders of magnitude from subnanometers to centimeters and from picoseconds to minutes. Even though it is possible to handle some aspects of these problems using analytical theory, most problems are nowadays tackled using simulations and related computational methods.

Only on the smallest scales in soft-matter research, the relatively well-understood *atomistic models* can be used, where every atom is represented by a (classical) particle interacting via so-called *force fields*. For this kind of model, a standard set of methods and algorithms exists, and elaborate simulation software packages implement these methods (*e.g.* GROMACS [LHvdS01], NAMD [PBW+05] and others).

Unfortunately, the relevant length and time scales for many soft-matter systems are often larger than are accessible to these models, and the importance of these larger systems is ever increasing. This applies in particular to complex biological systems, like biomembranes, ion channels etc. To be able to cope with the larger size and longer simulations times of the systems, it is necessary to reduce the number of degrees of freedom of the model systems, so that only the "important" ones have to be simulated. This process is called *coarse-graining*, and it plays a major role in soft matter

research.

A wide variety of such coarse-grained models exists, from near-atomistic models, where a small number of atoms are represented by so-called *unified atoms*, up to large scale continuum models that cover length scales of tens of micrometers and time scales of seconds.

Even near-atomistic models often can greatly profit from simulation methods and algorithms that are adapted to the model, and the more specialised models evidently require specialised methods. Even though most of these algorithms are based on the fundamental Monte-Carlo (MC) or Molecular Dynamics (MD) methods (see, for example [FS02]), a number of highly advanced and complex modifications and improvements of these basic algorithms for specific types of models have been developed during the last decades, for example the Lattice-Boltzmann method [BSV92, CCM92], Dissipative Particle Dynamics [HK92], Particle-Mesh Ewald methods [HGE73, DYP93, EPB+95] or the Fast Multipole Method [GR87] for electrostatic interactions, Parallel Tempering (or Replica Exchange) [ED05], or Adaptive Resolution Molecular Dynamics [PSK05], to name just a few of the more significant ones. Futhermore, the models often require highly specialised methods to analyse the data or to visualise the simulations. To sum up, as the size and complexity of the systems that are studied in soft-matter research are ever increasing, correspondingly the methods are getting more complex and the number of available methods is constantly growing. These facts make it hard or even impossible for a researcher or a group of researchers to have expertise on all the available methods at hand.

However, scientific working groups and projects in theoretical soft-matter research are organised around a soft-matter system or a type of problem, and not around a method, algorithm or model type. To tackle a specific kind of system or problem to the greatest effect, it might be necessary for a scientist to employ any of the various available methods, even though he has never worked with it before. Therefore, solving any particular problem usually comprises four stages. First, the scientist has to choose the methods that are best suited for the problem at hand. To be able to do that, he has to maintain or get an overview of the broad range of available methods in the field of soft-matter research. Second, he has to obtain in-depth knowledge of the often complex methods to be able to use them. Third, he has to search for suitable software packages, tools or program code that implement at least some of the required methods. Finally – if he is not very fortunate – he will have to implement at least parts of the methods himself, or at least he will have to adapt it to the own needs.

Unfortunately, none of these stages is greatly supported by the classical information resources of science. As will be described in the following sections, the classical channels of information exchange between scientists, *i.e.* scientific publications, are mostly insufficient or blocked for information exchange on the methodology of soft-matter research.

2.1.1. Maintaining an overview

Overview of a scientific field is usually provided by textbooks, review articles, conference talks, workshops, schools and lectures, which present the subject on an introductory level, that does not go into very much detail of a method. However, in the field of methodology, with few exceptions, these forms of presentation cover only well-established methods, and it usually takes several years until a new method is covered (*e.g.* [AT87, FS02, FCB06a, FCB06b]). New applications of a method, small improvements and variants that are suited for specific problems are typically not subject of the presentation.

2.1.2. Obtaining in-depth knowledge

In-depth knowledge on a subject in the sciences is typically passed on via scientific articles. For a few of the more successful and inventive methods, methodological articles are published in relevant journals, describing the new method in detail. In general however, articles in scientific journals mostly focus on the physical problem, which is considered more valuable, and not so much on the methods used to tackle it. Therefore, new methods are often obfuscated and hard to find in articles on a physical subject. This is true in particular for small improvements and variants of older methods, or new fields of application, which would not be considered publishable otherwise.

Furthermore, such articles usually do not describe the general applicability of the method, *i.e.* for what kind of problem it could be theoretically used, but only how it was employed in context of the problem at hand. Also, instead of describing the method in more general terms that can be easily adapted to other systems, it is normally described using the highly specialised terminology of the own field of research. These facts make it hard for scientists in soft-matter research to find useful references to in-depth knowledge on a specific method.

An example of this is the following: in many cases, coarse-grained off-lattice models can be effectively simulated using Monte-Carlo simulations. However, it is not obvious, how the algorithm can be correctly and efficiently implemented on parallel computers. In 2002, Uhlherr *et al* [ULA+02] published a journal paper that contains a simple parallel Monte-Carlo algorithm, that can be employed for many types of off-lattice models. Unfortunately, the article describes the algorithm in terms of an atomistic polymer simulation, so that it is not obvious that the article in fact contains a generic method for parallel off-lattice Monte-Carlo simulations of arbitrary systems. Consequently, the algorithm is unknown even in the working group of Daan Frenkel at the University of Amsterdam [1], an expert on this simulation type.

Another problem that classical printed information resources (like textbooks and journal articles) have to face, is, that they are relatively slow compared to the development in the field. Sometimes, it takes several years until a new method is covered

[1] Oral communication with Dr. Axel Arnold, October 2nd, 2007.

in a textbook, and it usually takes at least several months until an article is published in a journal.

2.1.3. Finding software and tools

After the soft-matter researcher has determined the best method for his problem, he might want to find software packages or program codes that implement the required methods, or at least some parts of it, and that can actually be applied to the problem at hand.

Classical scientific resources basically do not support this stage. Journal articles that describe software packages are very rare, and program code is almost never published. An exception to this is the *Computer Physics Communications Program Library*[2], a library, where the author of any article of the scientific journal *Computer Physics Communications* can publish accompanying program code or software. This does, however, require that an article is published in the journal. Otherwise, no common platform exists, where decriptions of software packages or program codes for soft-matter research are bundled.

In general, due to the relatively small community of soft-matter research, only very few useful software packages and tools that are dedicated to soft-matter research are publicly available, and it is even harder to find them. Instead, most research groups have in-house tools and program code, that are passed within the group. Such programs are usually badly documented and can not easily be employed for other problems, and therefore the developers are reluctant to publish the code, even though they might be willing to give the code to any researcher in need.

Another aspect is, that in some cases, software that has not explicitely been created for soft-matter research, can nonetheless – with some know-how – be effectively used for it. For example, even though the molecular viewer software VMD [HDS96] is intended mainly for the visualisation of atomistic models, it can easily be employed for many types of coarse-grained models. Again, this knowledge can not be scientifically published. So far, besides in direct exchange, no resource exists where scientists can gain this kind of knowledge.

2.1.4. Implementing the method

On the last stage, the reseacher has to actually implement the new method or at least parts of it. In this stage, the scientist faces a dilemma: On the one hand, the program code is expected to be very efficient from a computational point of view, as the problems are typically adapted to the available computing power, and none of the precious computing time is to be wasted. On the other hand, the methods and consequently their implementation is getting more and more complex. This problem is further enhanced by the advent of new hardware architectures (Multicore CPUs,

[2] `http://cpc.cs.qub.ac.uk/` (November 2, 2007)

GPGU processors, Cell processors, BlueGene architecture), that require new optimisation and programming techniques for effective utilisation, while older optimisation methods may not work anymore. Efficient parallelisation is nowadays mandatory, as all of these new hardware architectures massively exploit parallelism. To be able to use these techniques effectively, massive knowledge is required.

Unfortunately, again, the classical scientific resources mostly fail to support a researcher in this respect. Even if a scientific article on a specific method exists, the implementation and optimisation of the method is rarely its subject, and even less how the method can be adapted to a specific hardware. New ideas on how a method can be implemented or further optimised, or how it can be effectively adapted to a new hardware type, are not published. If it is communicated at all, knowledge on these subjects is almost exclusively passed orally from researcher to researcher. Furthermore, due to the fast development cycles of computer hardware, this type of knowledge outdates even faster, than the knowledge of the methods itself, so that the relatively slow process of scientific publication could not hold pace.

Evidently, the lack of resources for implementation and optimisation issues often has to result in non-optimal programs.

2.2. Collaborative online platforms

In 1990, Tim Berners-Lee and Robert Cailliau at CERN proposed the *World Wide Web* (WWW), a hypertext system that was originally meant to support rapid exchange of information between scientists [BLCG92]. In the original proposal, it was intended that any user of the WWW could as easily create and modify a web page, as he could read it. Since then, the WWW has rapidly developed and grown. Meanwhile, it is presumably the most important information resource for most people, and in particular for scientists, as it provides access to most of the classical scientific resources and a number of additional sources.

For a while, the WWW had mostly lost the possibility, that any user was able to create or modify its contents [GC00]. However, since a few years, with the coming of what is popularily dubbed the "Web 2.0", some of the original concept has been reintroduced into the WWW. Nowadays, a large number of web sites exist that provide participatory web applications, where users can not only read, but also create and publish their own contents. Popular examples of participatory web applications are wikis, web forums, blogs, video blogs and so-called "social networks" (*e.g.* MySpace). What is common to all of these applications is, that none of them requires special software, but that they can be used with the help of a standard web browser.

Curiously, although these kind of web applications enjoy great popularity and scientific communities could presumably greatly benefit from such platforms for information exchange, they are accepted only very reluctantly by scientists, and in particular in physics [But05]. This is surprising, as the concepts of many alternative forms of scientific publication are strongly supported by, or even originated in, the physics com-

munity, like the electronical preprint server arXiv[3], or the Open-Access movement, which propagates free and open access for all scientific publications[4].

In the following, two of these applications, namely wikis and web forums, and their use in the scientific community are presented in greater detail.

2.2.1. Wikis

A *wiki* (from Hawaiian: "*quick*"), or *WikiWeb*, is a collection of web pages on a web site, that are linked to other pages of the site via *hyperlinks*. What is special about this web application, is that the pages can not only be read easily, but also that it is very simple to create new pages, to modify existing pages, and to link them to other pages. The only technical requirement for this is a connection to the internet, and a web browser. The pages in a wiki are written in a very simple markup language, that enables even new users to create and modify pages very quickly. In many cases, wikis even allow *anonymous edits*, where the user is not required to log in or in any way to identify himself to be able to modify or create pages.

The concept of a wiki was developed by Ward Cunningham in 1995, who described it as "The simplest online database that could possibly work."[5]. Meanwhile, many different wikis are used all over the world, for information exchange on very different subjects and in very different contexts, from public wikis on games and other freetime-related activities, over public wikis on software packages and operating systems, and wikis that are used as content management systems for web sites, to wikis that are used as platform for information exchange on the intranets of large companies. The best-known public wiki is probably the Wikipedia[6], a collaborative encyclopedia, where articles are created and edited by millions of volunteers from all over the world. Surprisingly, even though the encyclopedia can be modified by anybody who can use a web browser, the quality of the articles is comparable to the Encylopedia Britannica [Gil05].

Meanwhile, a number of wikis are used in different scientific contexts, for example for nanotechnology labs and knowledge exchange in nanotechnology [CJR+06, JTS06] and in computer science education [ET05]. In the biological sciences, and in particular in bioinformatics, about ten different wikis are currently active[7].

2.2.2. Web forums

A forum is a web service, where users can start or participate in discussions. A user starts a discussion ("*thread*") with a contribution ("*posting*") on a specific subject (*e.g.* a question or a problem description). Other users can read all postings and *reply* to

[3] `http://www.arxiv.org` (November 4, 2007)
[4] `http://www.open-access.net/` (November 4, 2007)
[5] `http://www.wiki.org/wiki.cgi?WhatIsWiki` (November 2, 2007)
[6] `http://en.wikipedia.org/` (November 2, 2007)
[7] `http://wikiomics.org/wiki/Other_wikis_and_forums` (November 2, 2007)

any posting with their own. As in a wiki, web forums do not necessarily require a user to log in for posting, but sometimes also allow anonymous participation.

The user interfaces of web forums are often reminiscent of the interfaces of email clients. They allow to display the discussions in chronological order or in a threaded view, where replies to a posting are shown in a tree-like structure next to the original posting.

Web forums are commonly used for questions and answers in many domains, but also for general discussions on specific subjects. The concept of forums is not a new one, and it is probably the first type of platform where users could easily participate and publish information for other users to see.

The concept was probably first employed on electronical media by the so-called *newsgroups* of the Usenet, which basically provide the same functionality as web forums. Newsgroups have been invented by Tom Truscott and Jim Ellis as early as 1979, but are still used today, even if they get more and more replaced by web forums. Other platforms that employ similar concepts are the *bulletin boards* of classical modem mailboxes, that were mainly used in the 1980s and early 1990s, as well as *mailing lists*, that are still commonly used. The main difference between newsgroups, bulletin boards, mailing lists and web forums is the user interface and the kind of software that is required to use them.

So far, in science, web forums seem to be very rare and not widely used. An example of a scientific web forum is the site `http://forum.physorg.com`, where general questions of physics and technology are discussed between interested scientists.

2.3. SoftSimWiki: A collaborative online platform for soft-matter simulations

In the previous sections, it was shown, that on the one hand, the classical channels of scientific information exchange are blocked or insufficient for information on soft-matter simulation methodology, and on the other hand, that collaborative online platforms provide simple means of information exchange, where any user of the internet can easily participate.

Therefore, in the frame of this work, SoftSimWiki, a collaborative online platform for soft-matter simulations was developed, that is intended to provide an alternative channel and thus to fill this information gap. As of December 1, 2007, SoftSimWiki can be found under the WWW address

```
http://fias.uni-frankfurt.de/~olenz/softsimwiki
```

In the following sections, the concepts that SoftSimWiki employs, and solutions to the various problems that it faces, are described in detail, and it will be explained, why these concepts are supposed to be well-suited for soft-matter research. For technical reasons, as of November 24, 2007, not all of the described solutions are yet implemented in SoftSimWiki.

Fundamentally, SoftSimWiki combines a wiki and a web forum. The subject domain of SoftSimWiki are numerical methods in soft-matter research, with a focus on simulations. The platform is neither intended as a knowledge base or as a discussion forum for soft-matter physics in general, as the classical scientific media (conferences, articles, textbooks) are better suited for this task, nor is it intended as a general forum for technical support, as there are enough platforms of this type. To avoid that the subject of the platform gets out of hand, all contents should at least be remotely related to numerical methods of soft-matter science.

The wiki component will serve as a knowledge base for the domain. The pages in the wiki (which will be called *articles* in the following) should contain information on a specific subject, like a method or a software package, and they will structure the knowledge domain.

Each of these articles has an associated *discussion forum*, where users can start threads to ask questions on the subject of the article or to discuss the article itself. Even if an article on a specific subject does not yet have any contents, the forum can still already be active and be used to discuss the subject. Ideally, after a while, the knowledge from the discussion forum will be condensed into an article.

A new platform like SoftSimWiki faces several problems and has to overcome several barriers to be useful for the soft-matter community. In the following sections, these problems will be addressed, and the features of SoftSimWiki that are meant to solve them will be presented.

2.3.1. Technical fundament: Mediawiki and extensions

Technically, SoftSimWiki is based upon the MediaWiki software[8]. MediaWiki is a free open-source software package, that is used by Wikipedia. Caused by the great success of Wikipedia, it is probably the most frequently used wiki software. The software extends the basic idea of a wiki by many useful features that can be employed in SoftSimWiki. It is highly configurable, and it is constantly developed and improved. Furthermore, the software supports *extensions*, that allow to easily extend the functionality of MediaWiki. Meanwhile, a large number of extensions for various tasks exist, like advanced permission management functions, and also integration with various other web applications like web forums, Google Maps, YouTube, etc. As SoftSimWiki is a combination of a wiki with a web forum, the most important extension for this application is the intergration with a web forum.

The server that runs the software requires only standard tools that are easy to install on most computers: the webserver Apache, the database server MySQL, and the programming language PHP. All of these software packages are free and open-source. The combination of these components and the free open-source operating system GNU/Linux is used so frequently that it is has been dubbed with the acronym LAMP, and its functionality is comparable to any commercial software solution[9].

[8] `http://www.mediawiki.org/wiki/MediaWiki` (November 4, 2007)
[9] `http://www.heise.de/ct/english/98/12/230/` (November 4, 2007)

The great popularity of the LAMP server software and the MediaWiki wiki software ensures, that it will most probably be available and well maintained for at least several years.

2.3.2. Structuring the wiki

A wiki is a *hypertext*, *i.e.* a collection of pages that are linked via hyperlinks. Technically, no further assumption on the link structure is made. However, if a hypertext can only be navigated via these hyperlinks and there is no further underlying visible structure, users can easily get "lost in hyperspace" and find it very hard to effectively navigate the hypertext [Con87]. Therefore, it is useful to superimpose a structure on the hypertext that simplifies the navigation. The optimal navigational structure depends very much on the subject domain of the hypertext.

The most commonly used navigational structure of a hypertext is a *hierarchy*, where the articles are ordered in a tree-like structure. Hierarchies help to quickly locate the wanted information and to get an overview over the hypertext. In fact, they are a very common structure not only for hypertexts. Programs with a graphical user interface employ hierarchical menus, file systems use a hierarchy of folders, subfolders and files, and scientific publications use a hierarchy of chapters, sections and subsections to make a text easier to read.

To superimpose a hierarchical navigational structure to the articles in SoftSimWiki, MediaWiki provides the concept of *categories*. Categories can be used to group articles. Each article in the wiki can belong to an arbitrary number of categories. Furthermore, categories can have subcategories, thus forming a hierarchy. To support navigation, each category has an associated, special *category article*, which contain an automatically maintained, alphabetically ordered list of all articles and subcategories that belong to the category.

Initially, the topmost level of the hierarchy of categories in SoftSimWiki divides the articles into three topics, which correspond to the last three stages in the work of a soft-matter scientist as described before: *Methods and Algorithms*, *Software and Tools* and *Implementation*. The category articles of the top level are directly accessible from the wiki's main page and thus provide the main navigational structure of the platform.

Methods and Algorithms

The category *Methods and Algorithms* is meant for articles, that describe a method or an algorithm used in soft-matter research. The articles contain information of the following type:

- a short description of the method
- references to relevant scientific publications that describe or use the method
- information to which problem the method can be applied

- which software packages implements the method
- example code
- ...

Currently, the category has five subcategories. The category *Molecular dynamics* contains articles on methods that are specific for Molecular dynamics simulations, like for example thermostats. The *Monte-Carlo* category contains articles on Monte-Carlo specific methods. An example for an article in this category would be articles on special Monte-Carlo moves. *Long-ranged interactions* contains articles that describe methods that can be used for long-ranged interactions, for instance Particle-Mesh Ewald methods, or the Fast Multipole Method, while *Short-ranged interactions* contains articles on methods that are useful for systems that mostly employ short-ranged interactions, like cell lists or Verlet lists. The *Boundary conditions* category contains articles on methods that handle various types of boundary conditions, like for example sheared periodic boundary conditions.

Software and Tools

The category *Software and Tools* and its subcategories contain articles that give information on software packages and tools:

- a short description of the software
- what problems can the software be applied to
- who wrote the software and how it is developed
- a link to the software, if available
- the cost of the software
- what license is used (free, commercial, GPL ...)
- on which operating systems the software runs
- which libraries and tools are required to run the software
- whether it is open-source or not, and if it is, which programming language is used

The category has four subcategories. *Simulation software* is software that can be employed for numerical simulations, *e.g.* ESPResSo, or GROMACS. *Data analysis software* is software that can be used for statistical analysis of observables in simulations (like, for example, MATLAB), while *Visualization software* is used to create complex, three-dimensional representations of the studied system (*e.g.* VMD). *Plotting software* can create two- or threedimensional graphs of observables.

Note that the categories are not mutually distinct. A software package like MATLAB can perform as data analysis tool, but also for plotting or even simulation. In that case, the article should be added to all relevant categories.

Implementation

The category *Implementation* contains articles that are concerned with implementation issues. No further structure of the articles is imposed. Articles in this category may for example detail the implementation of a specific method, in which case the article would probably belong to both a subcategory of the *Methods and Algorithms* category, but also to the *Implementation* category. The subcategories of the category are *Optimization*, that contain articles on optimization techniques and *Parallelization* for articles that are related to parallelisation.

2.3.3. Supporting the scientific work style

A wiki that is to be used in a scientific context has to provide a number of features that are otherwise not usually found in wikis, so that it can be easily employed by scientists. Fortunately, already a number of the required features are provided by the MediaWiki software, while others can easily be added via extensions.

For basic editing tasks, like writing articles with basic layout functions, creating hyperlinks to other articles and web pages and decorating the texts with images, MediaWiki provides a simple markup language. Scientists that are used to write texts in LaTeX or HTML can usually learn and apply the language within minutes. Nonetheless, even though the basics of the language are simple, it also allows for complex layout tasks and provides the full power of HTML and CSS.

In the context of physics, a special requirement is, that the software has to allow to typeset mathematical formulas. In that respect, MediaWiki is ideally suited for the task, as all formulas are written using the LaTeX markup. As LaTeX is routinely used in soft-matter research for most scientific publications, this makes it particularly easy for soft-matter researchers to write mathematics in SoftSimWiki.

Another important feature for a platform in scientific context is, that it should support bibliographic references and citations. In the frame of this work, an extension to MediaWiki has been developed, that allows to include references from the bibliographic system BibTeX, a system that is commonly used in the soft-matter community in combination with LaTeX. The references are stored in a central database that can easily be queried, and they can be cited in articles via a unique key of the reference. In combination with the Biblio extension[10] of the Open Bioinformatics wiki Wikiomics[11], support for references from other bibliographic systems will be added and also the ability to automatically fetch those references from online resources like the preprint server arXiv[12].

[10] `http://wikiomics.org/wiki/Wiblio` (November 4, 2007)
[11] `http://wikiomics.org/wiki/` (November 4, 2007)
[12] `http://www.arxiv.org` (November 4, 2007)

2.3.4. Lowering the initial barrier

One of the fundamental problems of a newly introduced collaborative online platform is to get it off the ground in the first place, so that other people start to participate, even when there is no obvious benefit in being the first to do so. Once the platform is established and used by many users, it is usually no problem to find more users, that are attracted by the contents. To lower the initial barrier that keeps people from participating, SoftSimWiki employs the following strategies.

The first obstacles that could keep people from participating or even accessing the platform are technical barriers. Whenever it is necessary to install a particular software to be able to use a platform, many potential users are lost. Therefore, the technical requirements of SoftSimWiki are as low as possible. To be able to participate in SoftSimWiki, one needs a standard web browser and internet access. Both are expected to be readily available in the soft-matter community.

Furthermore, SoftSimWiki will initially contain knowledge, that the author of this work has collected during his career in soft-matter research. It is questionable, whether this will be enough for other researchers to see the benefit of the platform, in particular when they are working with other methods than the author.

To lower the effort for new users, another important aspect at least in the initial stage of SoftSimWiki is, that participating will not require registration or logging in to the system. Although being a registered user and logging in will have a number of advantages, a user does not have to do so and can still access all information and participate in any possible way.

Finally, it is important to note that SoftSimWiki is not only a wiki, that serves as the knowledge base for soft-matter methodology, but also a web forum, that allows researchers to ask and get answers on specific methods, software packages and implementation issues. Being able to answer or ask a question lowers the psychological barrier that might keep people from modifying an existing article on a specific subjects in the wiki.

2.3.5. Ensuring the quality

Given the fact that the platform allows anonymous and simple participation, another evident challenge for a collaborative platform like SoftSimWiki is to ensure the quality of the contributions.

Caused by the fact, that all articles can easily be modified and deleted by any user, the platform is prone to any kind of malicious modification, like random deletion or insertion of inappropriate contents (so-called *vandalism*), or insertion of unsolicited material, for example advertisement (*spam*). Although this kind of modifications can not easily be detected automatically, they can easily be identified by the regular participants of the platform. MediaWiki supports users to easily track down and undo malicious modifications, and to prevent their reoccurence. In general, every article has an *article history*, that lists every single modification since its creation date, and that allows to quickly revert the article to any previous version.

To track down acts of vandalism or spamming in the first place, MediaWiki provides a number of useful tools. As long as the platform is relatively small, it might be feasible for interested users to watch a special article that lists all recent changes done to any article or forum thread. Alternatively, it is possible to track all changes using an RSS newsfeed. Furthermore, registered users have *watch lists*, where they can add arbitrary articles. Whenever there are any changes to an article in the watch list, the user will be notified, and he can revisit and revert any modifications, if necessary. Whenever a malicious modification is found, MediaWiki allows to list all contributions that were done by the same user, or from the same IP-address, and if necessary, an administrator can mass revert all of the changes. To prevent the repetition of malicious modifications, administrators can block certain IP-addresses or users from editing any articles, or to protect certain article from being modified by unregistered or any users.

What is harder to prevent and detect are malicious modifications by educated users, or errors that are introduced unintentionally, so that they can not easily be recognised by the most users. In scientific publications, this is ideally prevented by the peer-review process, where other scientists from the field anonymously review articles before they are published.

In SoftSimWiki, a full-scaled peer-review process is out of the question, as it is far to time intensive and will not be able to keep up with small-scale changes. Instead, it is possible to give certain, trusted users the permission to *patrol* articles and modifications. When these users read an article or revisit a modification to an article, they can mark it as *patrolled*. Depending on the settings of MediaWiki, it is possible to have the wiki only show the last patrolled version of any article, or to clearly notify users when they are reading a version of an article that was not yet patrolled.

Note that initially, patrolling will not be activated in SoftSimWiki.

2.4. Conclusions

In the beginning of this chapter, it was shown, that although computational methods in soft-matter research are steadily getting more complex and difficult to master, the classical scientific publications do not sufficiently support scientists to exchange information on these methods. Neither in-depth knowledge of the methods, nor practical information concerning the methods like links to software packages and implementational issues are typically published in scientific journals or textbooks.

On the other hand, it was shown, that the increasingly popular collaborative online platforms like wikis and webforums allow rapid information exchange on various subjects for people from all over the world.

Therefore, a collaborative online platform, called SoftSimWiki, was developed and deployed in the frame of this work. Various issues concerning the use of an online platform for science have been discussed.

3. Off-lattice Template Library OLTL

In this chapter, the C++ class template library OLTL is presented, that provides classes and class templates that are useful for simulation programs, in particular of the Monte-Carlo type. In the first section, the design of the library will be discussed, while the consecutive sections describe the different classes and class templates in the library, providing examples of how the classes can be employed. As of December 1, 2007, the library, its documentation and its full API can be downloaded from

`http://fias.uni-frankfurt.de/~olenz/softsimwiki/OLTL`

3.1. Library Design

As was shown in the previous chapter, computational methods in general and simulations in particular have become a central tool in soft-matter research. When atomistic models are used, a number of highly useful software packages are available (*e.g.* GROMACS [LHvdS01], NAMD [PBW+05] and others).

For coarse-grained models, the situation is more complicated. Although most methods base on either the fundamental Molecular Dynamics (MD) or Monte-Carlo (MC) algorithms, due to the great number, variety and complexity of the methods and models, no single software package can possibly implement all of them. Nonetheless, there has been some progress in recent years, in particular for simulations of the MD type. Programs like ESPResSo [LAMH06] are flexible enough to cope with at least some of the various different MD-based methods. For simulations of the Monte-Carlo type, however, no comparable software packages exist, even though the Monte-Carlo method often has a number of advantages over the Molecular Dynamics method.

Consequently, in many cases, scientists have to implement the simulation programs themselves. Clearly, this has a number of disadvantages: the development time of hand-written code is far greater, than when a ready-to-run program can be used, the programs are more prone to significant errors, they are often not well optimised etc. OLTL is meant to support scientists in the task of writing their own simulation code.

3.1.1. Design criteria

In the development of the library, the following design criteria were considered to be most important, with the first being the most important:

Computational efficiency All classes that are likely to be employed in the central loop of the simulation are time-critical, *i.e.* it is important to make the classes

as efficient as possible from a computational point of view. In OLTL, all time-critical classes (Potentials and forces, cell lists, boundary conditions, random number generators) underwent extensive benchmarking, where several alternative implementations of the same procedure were tested. Also, comparisons with implementations of the same functionality in ANSI C were done to make sure that no efficiency is lost because of the use of the C++ programming language.

Correctness Of course, correctness of the classes is crucial. In OLTL, all classes are tested in an automated test suite to ensure correctness even after some changes in the implementation.

Simplicity of use Another central criterium for the design of the classes was, that they are as simple to use as possible. This means, for example, that overloaded variants with different parameters of the most methods exist that allow users to employ the most convenient variant. Also, the names of the methods are chosen to be as verbose as possible.

Consistent interface To make it easier for users to start using a new class, the interface of the different classes was made as consistent as possible. This holds in particular for classes that have a comparable functionality (like `Array1D`, `Array2D` and `Array3D`). Furthermore, consistent naming conventions for classes, methods and class members were employed.

Extensibility Furthermore, the extensibility of the library was considered to be important. As the library so far holds only the classes implemented for one specific simulation, it is by no means complete. To be of use for many scientists, it will be necessary to significantly extend the library. However, the interface of the classes in the library have been made a generic as possible to serve as template for further development.

Readability of the library's code To allow users to verify the correctness of the code, the libraries code was made as readable as possible. Extensive in-code and external documentation is part of the library.

Comprehensivity To give users the most freedom, many of the libraries classes are actually class templates, that allow a user to employ his own data types. For example, the cell lists algorithm does not impose any restriction on the type of the vector that is passed to locate a particle.

3.1.2. Toolbox vs. framework

Object-oriented libraries usually come in either one of two fashions, that will be labelled *framework* and *toolbox* in the following: while a *framework* provides complex, highly interlocked classes, a *toolbox* consists of mostly independent classes.

Classical examples for the framework type of library are libraries that provide graphical user interfaces, like Qt[1] or GTK[2]. Frameworks usually provide classes with powerful and complex functionality, that are highly interwoven and strongly depend on each other. Within the framework, the interfaces of the different classes are consistent and well-designed. These advantages come, however, at the expense that it is necessary to use either none or all of the library. This also means, that it is generally not possible to add some of the functionality of the library to an already written program, but that the program has to be made to fit into the framework. Furthermore, even though it might be highly productive to use such a library once one feels comfortable with it, the initial barrier and the time required until one can start being productive is generally relatively high.

For developers, the complex structure of a framework generally makes it much harder to extend its functionality.

Examples for the toolbox type of library are the well-known Standard Template Library (STL) [MS95], or the most libraries of the Boost project [3]. Toolboxes contain classes, that are mostly independent of each other. For the users of the library, this has a number of advantages. First of all, using a toolbox usually does not require, that the program was designed with the toolbox in mind, but the toolbox can be incrementally included into existing program code. Independent classes give the user much more freedom in the design of his own program, and they allow to use only those classes that he feels to be useful, while ignoring the others. Furthermore, the initial barrier that has to be overcome to use the library is much lower, as the user only has to master one class at a time. However, these advantages come at a cost for the developer of the library. The biggest problem for developing a toolbox is to identify the relevant operations and objects and to encapsulate them into independent objects. Also, the developer typically has to put more effort into providing a convenient interface for the classes, that limit the user as few as possible. A C++-technique that is particular useful on that behalf are *class templates*, that allow to define classes and algorithms independently of some of the data types used within the class, so that they can be specifiec by the user. An advantage of the toolbox approach from the developer's point of view is, that toolboxes are easily extensible.

Considering the previously described facts and comparing them with the design criteria, it was decided to structure OLTL as a toolbox. Note, that the following descriptions do describe only those classes that are considered to be interesting. When OLTL is to be actually employed, it is recommended to have a look at the documentation of the full API first.

[1] `http://trolltech.com/products/qt` (November 30, 2007)
[2] `http://www.gtk.org/` (November 30, 2007)
[3] `http://www.boost.org/` (November 30, 2007)

3.2. Boundary conditions

The *boundary conditions* in an off-lattice simulation define, how the particles behave at the system's boundaries. Only in a few rare cases, when all particles in the simulation are sufficiently bound to each other, it is possible to simulate a system with *open boundary conditions*, *i.e.* a system without any boundaries, otherwise the particles would drift to arbitrary distances from each other.

A possible way to solve this problem would be to introduce artificial walls in the system that confine the particles into the space between them. With the exception of simulations of very small structures, like for example nanofluidic devices, where the walls are part of the real system, these walls typically cause unwanted wall effects.

To avoid these, *periodic boundary conditions* (PBC) pose an elegant solution. A three-dimensional system employing PBC can be thought of as a four-dimensional torus, where a particle that crosses a boundary and leaves the system reenters the system at the opposite boundary. Furthermore, particles close to one boundary interact with particles at the other boundary [FS02]. This effectively approximates an infinite system without any boundaries. Alternatively, one can think of a system using PBC as a system, where every particle has infinitely many copies (or *images*) in all directions and interacts with the (infinitely many) images of all other particles.

The shape of the unit cell for periodic boundary conditions does not have to be rectangular; any space-filling shape is possible. In particular in atomistic simulations of proteins, truncated octahedrons are a commonly used shape, as it relatively closely resembles a sphere and the (typically globular) protein and therefore requires fewer water molecules to fill up the empty space.

If a rectangular unit cell is used, this has the advantage that the different directions are independent. Therefore, it is possible to apply different boundary conditions on the different axis. For example, it is possible to employ open boundary conditions in the x-y-plane and confining the particles within a cylinder, while using periodic boundary conditions along the z-axis, thus mimicking an infinitely long cylindrical channel.

In some cases, it is furthermore necessary to generalise the rectangular shape of the unit cell to an arbitrary triclinic unit cell, which can be considered as a *sheared rectangular unit cell*. This can be the case, for example, when the simulated system has a crystalline structure, where the lattice constant of the crystal doe not match the system size. When a generic triclinic unit cell is used, the different directions are not completely independent anymore. Note, that any triclinic unit cell can be mapped onto a corresponding rectangular unit cell, where each of the images of the unit cell in a certain direction is shifted with respect to the previous image (see figure 3.1).

From an algorithmical point of view, different types of boundary conditions influence, how distances and angles between coordinates within the system are computed, and whether or not coordinates and distances can be *folded* into the central periodic image. To support the handling of the different types of boundary conditions used in soft-matter simulations, OLTL provides the following classes:

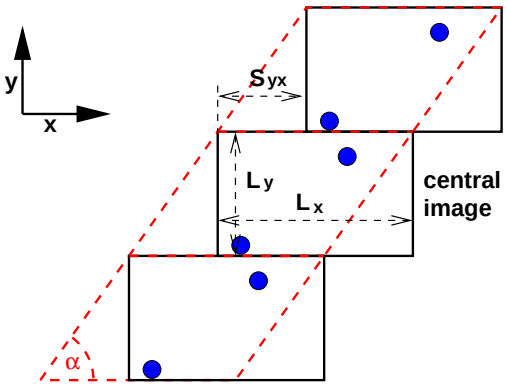


Figure 3.1: Two-dimensional scheme of periodic boundary conditions with a sheared rectangular unit cell (black solid lines) and the corresponding triclinic unit cell (dashed red lines). While in the triclinic unit cell, a non-orthogonal angle α is used, in the sheared rectangular unit cell, the next image of the cell along the y-axis is shifted by S_{yx}.

`RectangularBoundaryConditions` implements rectangular boundary conditions in a three-dimensional system. It allows to independently define the boundary conditions in the different directions to be open or periodic.

`ShearedBoundaryConditions` implements periodic boundary conditions with a sheared rectangular unit cell. Note that computing distances and angles in sheared periodic boundary conditions is computationally much more expensive, than in rectangular boundary conditions.

Both classes share a common interface. The usage of both classes is demonstrated in listing 3.1. Given these two examples of classes, it should be easy to extend OLTL by classes that implement other types of boundary conditions.

3.3. Cell lists

In a naive approach to an off-lattice simulation, the most significant contribution to the computational effort stems from the computation of the interactions between particle pairs, which scales as $O(N^2)$ for N particles.

In many cases, however, only short-ranged interactions are used in soft-matter systems, where the interactions between particles that are far from each other can be neglected. This fact can be exploited to speed up the simulation. To this end, we first define the cut-off distance r_{cut}, which corresponds to the distance between two particles at which the particles do not interact anymore.

In the cell lists algorithm [FS02], the whole coordinate space is divided into *cells* with a minimum side length of r_{cut}. Each of the cells has an associated linear list of all particles that are inside the cell. This means, that when picking out an arbitrary

```
// define rectangular boundary conditions, with open boundaries in x
// and y, and periodic boundaries in z
RectangularBoundaryConditions< OpenBC, OpenBC, PeriodicBC > rect_bc;
// define sheared rectangular boundary conditions
ShearedBC3D sheared_bc;

// set the lengths in the periodic dimension
rect_bc.set_length_z(100.0);
// define one of the shifts for the sheared rectangular BCs
sheared_bc.set_shift_y_x(0.1);

// get the difference vector c of the coordinates a and b
c = rect_bc.get_difference(a, b);
c = sheared_bc.get_difference(a, b);

// get the length of the difference vector of a and b
l = rect_bc.get_distance(a, b);
l = sheared_bc.get_distance(a, b);

// get the angle spanned by the coordinates a, b and c
phi = rect_bc.get_angle(a, b, c);
phi = sheared_bc.get_angle(a, b, c);

// get the central image of a
a1 = rect_bc.fold(a);
a1 = sheared_bc.fold(a);
```

Listing 3.1: Example code using rectangular and sheared rectangular boundary conditions.

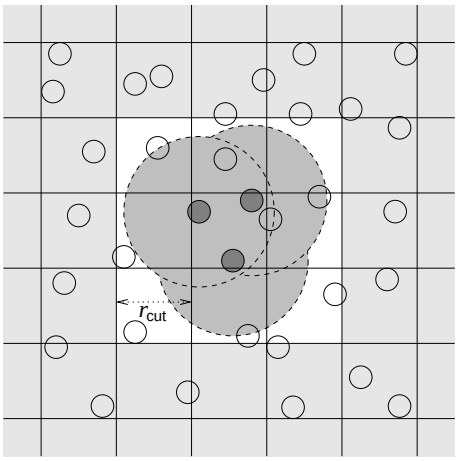


Figure 3.2: Sketch of the cell lists algorithm. Only the particles in the neighbouring cell lists can have a distance less than r_{cut} to any particle in the central cell list.

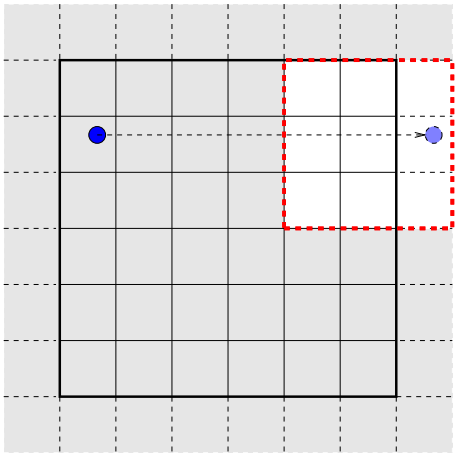


Figure 3.3: Sketch of the shadow cell lists algorithm. When a particle close to a boundary is inserted, a shadow particle at the opposite boundary is created and put into the shadow cell.

particle A, all particles that are closer than r_{cut} to particle A are either in the same cell list as particle A, or in one of the neighbouring cell lists. Consequently, instead of looping over all particles when computing the interactions of particle A, it is sufficient to only loop over the particles in the neighbouring cell lists. Figure 3.2 constains a sketch to illustrate the cell lists algorithm.

Depending on the size of the system, this can significantly reduce the computing time. When the density of the system is kept fixed, the cell lists change the scaling of the simulation algorithm to $O(N)$ for N particles.

The cell lists algorithm strongly interacts with the boundary conditions of the system. In a rectangular system with periodic boundary conditions, the cells close to the boundary have to consider the cells at the opposite boundary as neighbours.

Interestingly, the coupling of the cell lists algorithm with the boundary conditions allows for an elegant treatment of periodic boundary conditions, which will be called *shadow cell lists* algorithm. This algorithm has the advantages, that it reduces the otherwise relatively costly computation of distances in periodic boundary conditions to the computation of distances in a non-periodic system at the cost of a slightly larger computational effort when inserting the particles into the shadow cell lists data structure.

In the algorithm, all cells that are next to a boundary have virtual copies, the "shadows", directly outside of the opposite boundary. This means, that whenever a particle that is closer to the boundary than r_{cut} is inserted into the cell lists structure, a periodic image of the particle at the opposite boundary is created as a virtual particle (the "shadow"), and it is inserted into the shadow cell, that is the direct neighbour of the cell at the opposite boundary (see figure 3.3). Correspondingly, when the particle is erased from the cell lists structure, all of its shadows have to be erased.

Consequently, when the distances between any particle and its neighbours are it-

erated, the cell list will contain the correct periodic image, and the distance can be computed non-periodically. The shadow cell lists algorithm is useful in particular in conjunction with sheared rectangular boundary conditions (see section 3.2), as the cost of the distance computation in sheared periodic boundary conditions is very high.

In OLTL, three variants of the cell lists algorithm are implemented.

`CellLists3DTemplate` implements the basic cell lists algorithm.

`ShadowCellLists3DTemplate` implements the shadow cell lists algorithm for rectangular unit cells and arbitrary boundary conditions along the three axes.

`ShearedCellLists3DTemplate` implements the shadow cell lists algorithm for sheared rectangular unit cells.

All three share a common interface, so that they can easily be exchanged. To give users the full freedom, the algorithms are implemented as *class templates*, that can be used with any representation of the particles. Each of the implementations provides the central subclass `CellLists3DTemplate::niterator`, that implements the *Iterator* design pattern [GHJV95, 257] and that can be used to iterate over the particles in a cell list and its neighbours.

Listing 3.2 shows a usage example of a cell lists class template.

3.4. Potentials

The interactions within an off-lattice simulation are usually defined via potentials. OLTL defines a common interface for potentials, plus a number of wrapper templates (following the *Decorator* design pattern [GHJV95, 175]) that can be used to modify the potential in various ways.

OLTL implements the following potentials:

`LJPotential` implements the classical (12,6)-Lennard-Jones potential, which is defined by

$$V_{\mathrm{LJ}}(r) = \epsilon \left(\left(\frac{\sigma}{r}\right)^{12} - 2\left(\frac{\sigma}{r}\right)^{6} \right) \tag{3.1}$$

`LJPotential_from_sqr` also implements the classical (12,6)-Lennard-Jones potential, but expects the square of the distance as the input.

`FENEPotential` implements the finite extensible nonlinear elastic potential (FENE), defined by

$$V_{\mathrm{FENE}}(r) = -\frac{1}{2}\epsilon(\Delta r_{\max})^2 \log\left(1 - \frac{r - r_0}{\Delta r_{\max}}^2\right) \tag{3.2}$$

```
// create a cell lists structure that saves pointers to a vector and
// is periodic in all directions
typedef CellLists3DTemplate<Vector3D*, true, true, true> CellLists3D;
CellLists3D celllists;
CellLists3D::cell_list_type cl;

// define the cell size
celllists.set_min_cell_length_x(r_cut);
celllists.set_min_cell_length_y(r_cut);
celllists.set_min_cell_length_z(r_cut);
// define the domain size
celllists.set_length_x(100.0);
celllists.set_length_y(100.0);
celllists.set_length_z(100.0);

// initialise the cell lists
celllists.init();

// insert a particle at coordinate p
celllists.insert(&p);
// get the cell list of a particle
CellLists3D::cell_list_type &before = cell_lists.select_cl(&p);
// iterate over all neighbours of p
CellLists3D::niterator neighbour;
for (neighbour = cl.neighbours_of(&p); neighbour != cl.nend(); neighbour++) {
  dx = *p[0] - **neighbour[0];
  dy = *p[1] - **neighbour[1];
  dz = *p[2] - **neighbour[2];
  d = sqrt(dx*dx + dy*dy + dz*dz);
}
```

Listing 3.2: Example code using the cell lists.

```
// create a Lennard-Jones potential
CutOffShift< LJPotential > lj_pot;
// set the parameters
lj_pot.set_parameters(1.5, 1.0);
// set the cutoff of the potential and make it continuous
lj_pot.set_cutoff(2.0);
lj_pot.make_continuous();

// compute the potential
v = lj_pot(r);
```

Listing 3.3: Example code using potentials.

`UpperHalfFENEPotential` and `LowerHalfFENEPotential` implement variants of the above `FENEPotential`. They are defined by

$$V_{\text{upper}}(r) = \begin{cases} V_{\text{FENE}}(r) & ,\text{if } r > 0 \\ 0 & ,\text{otherwise} \end{cases} \tag{3.3}$$

$$V_{\text{lower}}(r) = \begin{cases} V_{\text{FENE}}(r) & ,\text{if } r < 0 \\ 0 & ,\text{otherwise} \end{cases} \tag{3.4}$$

`BendingPotential` implements a bending potential defined by

$$V_{\text{BA}}(\theta) = \epsilon\,(1 - \cos\theta) \tag{3.5}$$

Listing 3.3 demonstrates the usage of the `LJPotential` class and the `CutOffShift` wrapper template, that cuts off and shifts the potential at a chosen distance so that it is continuous.

3.5. Random number generators

In Monte-Carlo simulations, it is crucial to have access to good pseudo-random number generators. The GNU scientific library (GSL) [GDT+06] [4], a library for the C-programming language, provides more than 20 different random number generators. For better convenience, OLTL provides C++-classes that form a wrapper around the C functions from GSL. Listing 3.4 demonstrates the usage of the `RNG` class.

3.6. Nematic order parameter

The class `Nematic` computes the *nematic order parameter* of a set of vectors. The nematic order parameter of N vectors is defined to be the largest eigenvalue of the

[4] `http://www.gnu.org/software/gsl/` (December 3, 2007)

```
// create a Tausworthe high quality random number generator
RNG rng(taus2());
if (rng() < 0.5) { cout << "heads" << endl; }
else { cout << "tails" << endl; }

// create a Mersenne Twister RNG
RNG rng(mt19937());
```

Listing 3.4: Example code using the GSL random number generator wrapper class `RNG`.

```
// init a nematic object
Nematic nematic;
// add three vectors
nematic.add(v1);
nematic.add(v2);
nematic.add(v3);
// compute the nematic order parameter
nematic.compute();
S = nematic.get_S();
direector = nematic.get_director();
```

Listing 3.5: Example code using the `Nematic` class.

matrix

$$S_{ij} = \frac{1}{2N} \sum_{n=1}^{N} (3x_i^{(n)} x_j^{(n)} - \delta_{ij}) \tag{3.6}$$

where x_i is i'th component of the n'th vector. The eigenvector corresponding to the nematic order parameter is the so-called *director*[dGP93]. Listing 3.5 shows the usage of the class.

Based on the examplary interface of this class, many other statistical observables can be implemented.

3.7. Multi-dimensional arrays

Although the Standard Template Library STL provides an excellent implementation of arrays in one dimension, the class template `vector`, it is not trivial to extend this class to two or three dimensions. As in simulations and other programs, two- or three-dimensional arrays are a frequently used data structure, OLTL provides the class templates `Array1D <T>`, `Array2D <T>` and `Array3D <T>`, which implements one-, two- and three-dimensional arrays, respectively.

Furthermore, the array class templates provide (optional) automatic enlarging of the array. Whenever an element is requested, whose index is out of the bounds of the array, the array is autoamtically enlarged such that the element exists.

All classes share a common interface, that allows for

```
// create a three dimensional array with initial size 10x10x10
Array3D< double, enlarge > I(10, 10, 10);
// set the prototype for automatic enlarging to 0.0
I.set_prototype(0.0);
// set one element of the array to 1.0
I(4,4,4) = 1.0;
// add 2 to all elements
I += 2.0;
// automatically enlarge the array
I.at(12,13,11) = 3.0;
```

Listing 3.6: Example code using the `Array3D` class.

```
// create an automatically enlarging, three dimensional histogram
// the bins have a size of 2x2x2
Histogram3D< enlarge > histo(2.0,2.0,2.0);
// add two elements to the histogram
histo.add(1.0,1.0,1.0);
histo.add(7.0,3.3,17.3);
// normalise the histogram
histo.normalise();
// output the histogram
cout << histo << endl;
```

Listing 3.7: Example code using the `Histogram3D` class.

- easy and fast element access via the operator `()`
- STL-style iterators to iterate over its elements
- Operators `+=`, `-=`, `*=`, `/=` for element-wise manipulation
- range-checked element access via `at()` with (optional) automatic resizing

The one-dimensional array class template `Array1D` has a functionality comparable to the STL `vector<T>` class template, but has an interface equal to the `Array2D` and `Array3D` class templates and furthermore provides (optional) automatic resizing.

3.8. Histograms

When computing observables in soft-matter simulations, *histograms* are frequently required. OLTL provides convenient class templates `Histogram1D`, `Histogram2D` and `Histogram3D` for one-, two- and three-dimensional histograms with a common interface. The classes provide automatic enlarging, which is very useful when the value domain is not known before the histogram is used. Listing 3.7 shows the usage of the histogram class templates.

```
try {
  ParamSet params("test.par");

  // read the input file name
  // if the parameter is not defined in the set, substitute "data.in"
  // as default
  char[30] input_file;
  params.get_key("input_file", input_file, "data.in");

  // read the double "pi"
  // if the parameter is not defined, throw an exception
  double pi;
  params.get_key("pi", pi);

  // read a number between 0 and 100, throw an exception if the
  // parameter is out of range
  int number;
  params.get_key("number", number, 0, 100);
} catch (logic_error &er) {
  cerr << er.what() << endl;
  exit(1);
  }
```

Listing 3.8: Example code using the `ParamSet` class.

3.9. Parameter files

In many cases, it is useful to be able to read sets of parameters from one or multiple files of the format:

```
# This is a comment

input_file    =    foobar.txt
pi            =    3.141
number        =    42
```

OLTL contains a convenience class `ParamSet` that allows to easily access this kind of parameter files. Listing 3.8 demonstrates the usage of the class.

3.10. Conclusions

In this chapter, the Off-Lattice Template Library OLTL has been presented. The library is a toolbox of independent classes that can be used in the context of off-lattice simulations in soft-matter research, and it is easily extensible.

4. Lipid bilayers

This chapter will present an introduction to the physics and chemistry of lipid bilayers. A full treatment of these subjects would be far beyond the scope of this thesis. For more detailed information, please refer to textbooks on biomembranes (*e.g.* [Gen89]) or to reviews of the characteristics of lipid bilayers (*e.g.* [KC94a, KC94b, KC95, KC98]).

The chemical structure of lipids (section 4.1) gives rise to a number of interesting phases of lipid-water mixtures (section 4.2). One of these phases is the lipid bilayer. Lipid bilayers themselves exhibit several different subphases (section 4.3).

4.1. Lipid structure

In general, the term *lipid* is used to denote a large number of different molecules that are not soluble in water. In this work, we will restrict the term to refer only to *amphiphilic* lipids as they occur in biological membranes. These molecules have an elongated, hydrophobic moiety (the *tail* of the lipid) and a more globular, hydrophilic *head*.

The head and tail moieties can be of very different chemical nature. To understand the rest of this thesis, these chemical details are mostly irrelevant, as the molecules will be coarse-grained to a level that corresponds to the description above. However, to be able to compare the data obtained by the simulation of the lipid model with real lipid data and to be able to incorporate changes to the model, the relevant chemistry is described in the following.

The lipids most abundant in the membranes of animals, plants and bacteria are the phospholipids, in particular phosphatidylcholines (lecithins, *abbr.* PCs), phosphatidylethanolamines (PEs), and sphingolipids.

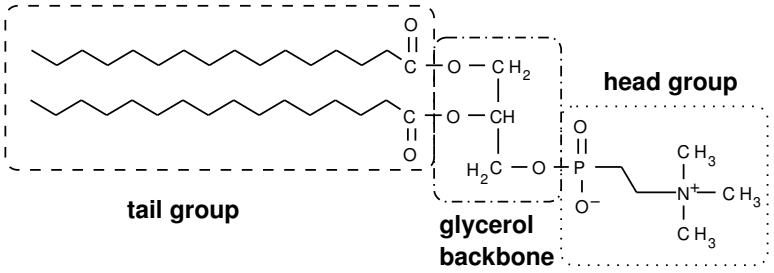


Figure 4.1: Chemical structure of dipalmitoyl phosphatidylcholine.

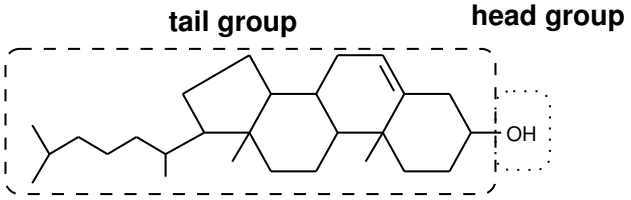


Figure 4.2: Chemical structure of cholesterol.

Figure 4.1 on the preceding page depicts the chemical structure of the lipid dipalmitoyl phosphatidylcholine (DPPC), that is often used as a benchmark molecule for bilayer simulations and the lipid best researched experimentally. The tail group of DPPC consists of two fully saturated hydrocarbon chains of fatty acid origin that are esterified to the glycerol backbone. The third position of the glycerol backbone is connected to the phosphocholine head group.

Other membrane lipids differ from DPPC in the following characteristics:

hydrocarbon chain length The lipids dimyristoyl phosphatidylcholine (DMPC) and dilauroyl phosphatidylcholine (DLPC) differ from DPPC only in the length of the hydrocarbon tail groups (DPPC: $C_{16}H_{33}$, DMPC: $C_{14}H_{29}$, DLPC: $C_{12}H_{25}$) but are otherwise chemically identical to DPPC. The tails of membrane lipids usually have a length between 10 and 22 acyl groups, with a preference on even numbers. Although both tail groups may be of different lengths, they are typically the same.

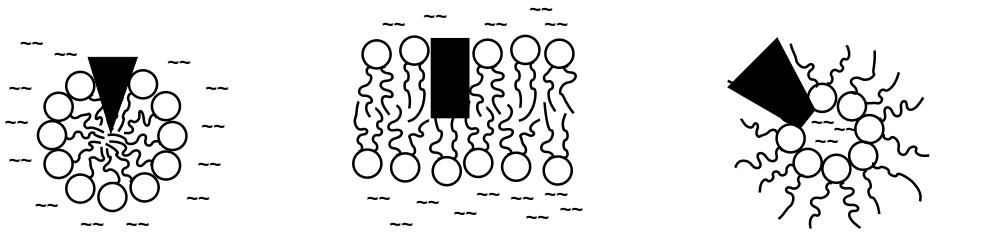
tail branching and degree of unsaturation Some lipids found in biomembranes possess unsaturated or branched hydrocarbon tails. While unsaturated chains are common among membrane lipids, branched tails are very rare.

chemical nature of the head group There is a great variety of head groups, with very different sizes, polarity and chemical characteristics. PCs have a choline head group while PEs have an ethanolamine head group. In the model described in this thesis, different head group characteristics are almost completely ignored.

number of tails A phospholipid or sphingolipid may have one or two hydrocarbon tails. However, in biological membranes, lipids with two hydrophobic tails are the most prevalent. This can be understood from purely geometrical considerations (see section 4.2).

chemical nature of the tails Sphingolipids differ from phospholipids in the nature of the hydrophobic tails. A well-known example of this type of lipid is sphingomyelin. In sphingolipids, the hydrophobic group is a ceramide.

overall chemical structure Although most of the lipids that occur in biomembranes are phospho- or sphingolipids, there are also lipids that have a completely different chemical structure. The most prominent example is cholesterol, that occurs in significant amounts in most biological membranes (see figure 4.2).



(a) *micelles*, *cubic* or *sponge phase*

(b) bilayer

(c) *inverted micelles*, *hexagonal* or *inverse cubic phase*

Figure 4.3: Lipid phases in a watery environment.

4.2. Lipid phases in watery environment

Because of their amphiphilic nature, in watery environment, lipid molecules tend to assemble into mesoscopic aggregates. A great number of different phases of water-lipid systems has been identified.

The most important parameter of the lipids that controls the system's phase behaviour is the head-tail size relation as expressed by the packing parameter S. The packing parameter is defined by

$$S = \frac{V_t/l_t}{A_h} \tag{4.1}$$

where A_h is the surface area of the hydrophilic head group, V_t is the volume and l_t the length of the hydrophobic tails of the lipid, *i.e.* V_t/l_t is the average cross sectional area of the tail groups parallel to the surface. Geometrical considerations show that the lipids form the following aggregates for different values of S [IMN76, Isr92]:

$S < 1/2$**: micelles/cubic phase/sponge phase** (figure 4.3a) For low values of S, the surface area of the head group is much larger than the average cross sectional area of the tail groups. The lipid has an overall cone-like shape. Therefore, it can not form a flat bilayer, but instead creates spherical ($S < 1/3$) or cylindrical micelles ($1/3 < S < 1/2$), or interconnected structures like the cubic or sponge phases. Phospholipids that have a single aliphatic tail group fall into this range.

$1/2 < S < 1$**: bilayers** (figure 4.3b) For values of S up to one, the lipid has a roughly cylindrical shape, *i.e.* the head surface area is roughly identical to the tail cross section area. These lipids can assemble into flat bilayers. Most phospholipids with two aliphatic tail groups fall into this category.

$1 < S$**: inverse micelles ("hexagonal phase")** (figure 4.3c) For values of S greater than one, the lipid has the shape of an inverted cone, as the head group area is smaller than the tail cross section. These lipids create densely packed inverted spherical micelles that enclose some of the water on the inside. Cholesterol, but also phospholipids with a small head group and two branched or unsaturated tails form these kind of structure.

As lipid bilayers are the basic structure of a biological membrane, this explains why most membrane lipids possess two hydrocarbon chains that are mostly saturated.

4.3. Lipid bilayer phases

This section deals with the phase behaviour of pure lipid bilayers. Several phases can be clearly distinguished by differences in the head and tail ordering. In the following, only a very simplified and generalised overview of the phases and the phase behaviour of pure lipid bilayers can be given. For detailed information, please refer to [Gen89, KC94a, KC94b, KC95, KC98].

Biomembranes consist of a complex mixture of different lipid types with very different characteristics. The phase behaviour of such mixtures is much more complex than the phase behaviour of a single lipid bilayer. In this work, only bilayers of a single lipid type are considered.

4.3.1. Temperature dependence

Figure 4.4 shows the temperature dependence of the molecular volume and the heat capacity of a DPPC bilayer. Both graphs show the signature of three distinct phase transitions in a temperature range between 10 and 45 ^0C.

At low temperatures, the bilayer is found to be in a so-called *subgel phase*, the *lamellar crystalline* phase L_c. In this phase, the aliphatic tail chains are mostly stretched, parallel to each other and very densely packed and well ordered. The long axis of the head group is oriented parallel to the bilayer plane, and the heads are closely packed in a lattice with presumably orthorombic cells.

When the temperature is increased, the so-called *subtransition* from the subgel phase L_c to the *gel phases* L_β or $L_{\beta'}$ is encountered. The transition is mainly driven by the head group interaction: the polar head groups become hydrated and orient themselves perpendicular to the bilayer plane, the distance between the heads increases

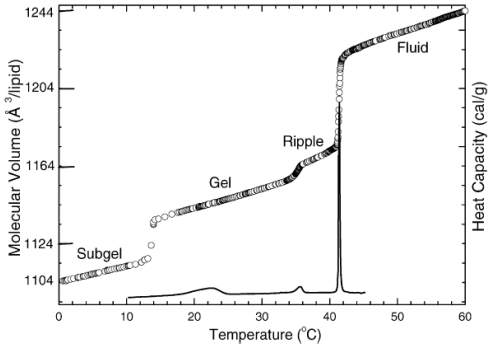
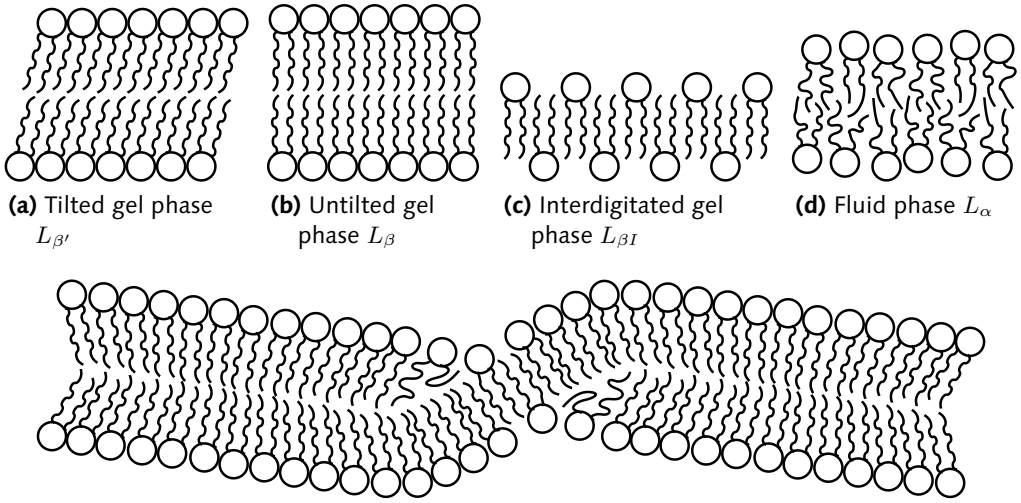


Figure 4.4: Temperature dependence of the molecular volume and the heat capacity of a DPPC bilayer (from [TNN04]).



(a) Tilted gel phase $L_{\beta'}$ **(b)** Untilted gel phase L_β **(c)** Interdigitated gel phase $L_{\beta I}$ **(d)** Fluid phase L_α

(e) Ripple phase $P_{\beta'}$

Figure 4.5: Sketch of the bilayer phases.

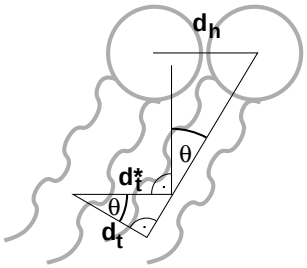


Figure 4.6: Sketch of the lipid tilt. By tilting the lipid tail chains with a cross section d_t by the angle θ, the component in the bilayer plane d_t^* is increased to match the head cross section d_h.

and the head group ordering changes to a hexagonal or quasi-hexagonal lattice. However, the tail chains are still well ordered and densely packed (see figure 4.5).

The behaviour of the tail groups in the subgel phase L_c and the gel phases L_β and $L_{\beta'}$ again very much depends on the relation of the head surface area and the tail cross section, *i.e.* on the packing parameter S. When the cross section of the head group is larger than the cross section of the tails, the lipid tail chains become tilted with respect to the bilayer normal. This effectively increases the tail cross section parallel to the bilayer plane, so that the mismatch between head and tail size is reduced (see figure 4.6). Consequently, the tails can align and densly pack, and a stable bilayer can be maintained. This effect is mostly emphasised in the gel phases, leading to the distinction between the *tilted gel phase* $L_{\beta'}$ and the *untilted gel phase* L_β. While PCs generally exhibit the tilted gel phase $L_{\beta'}$, PEs or sphingolipids possess an untilted gel phase L_β.

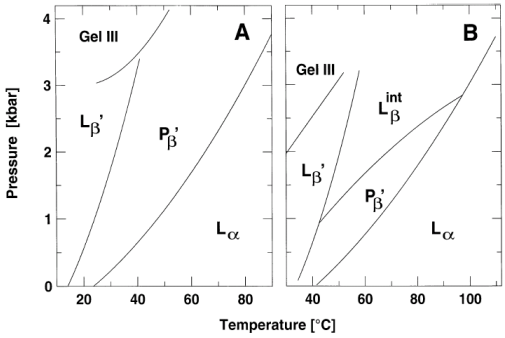


Figure 4.7: Experimental p-T phase diagram of a DMPC-bilayer (A) and a DPPC-bilayer (B) from [KC98]

Upon further increasing the temperature, many lipid systems that exhibit the tilted gel phase $L_{\beta'}$ undergo the so-called *pretransition* to the *ripple phases* $P_{\beta'}$ or $P_{\beta}^{(mst)}$. In these curious phases, the bilayer, that is flat in all other phases, exhibits a rippled structure with a repeat distance of a few tens of lipids. The molecular structure of these phases is not well understood. Although there seems to be a certain degree of disorder in the system, the bilayer is generally well ordered, comparable to the gel phase $L_{\beta'}$. Only recently, molecular dynamics simulations of lecithin bilayers have unveiled the molecular structure of the phase [VYMM05], but the nature of the transition is still not well understood. Chapter 8 will shed further light onto the characteristics and the structure of the phase as well as the mechanisms driving the transition.

When heating up the system some more, the system undergoes a highly cooperative phase transition, the so-called *main transition* or *chain order/disorder transition* of lipid bilayers. Above the transition temperature, the *liquid-crystalline* or *fluid phase* L_{α} is found (see figure 4.5d on the previous page). Although the lipids are bound to the bilayer and can not escape into the watery environment, the heads and tails are quite disordered and fluid-like, and a high in-plane mobility can be observed (figure 4.5d). The bilayer in this phase can be thought of as a two-dimensional fluid. With only a few exceptions, the lipid component of biomembranes in natural environment is found to be in this phase [Gen89].

4.3.2. Pressure dependence

Figure 4.7 shows experimental pressure–temperature phase diagrams of bilayers of DPPC and DMPC. From these diagrams it can be seen that the main and subtransition-temperatures increase with increasing pressure. However, what is more eye-catching is the fact that two new phases are observed at higher pressures.

In DPPC bilayers, at intermediate temperatures and high pressure, a new phase, the so called *interdigitated phase* $L_{\beta I}$, occurs. In this phase, the hydrophobic chains of the lipids interdigitate and create a bilayer with a very low thickness and no tilt. The chains are very densely packed, however, there is a large head–head distance (see

figure 4.5c on page 47). At normal pressures, the phase is absent in most PCs, as it has unfavourable voids between the heads and increases the tail–water interface. However, the phase can be induced by adding alcohol or chaotropic salts to the lipid-water mixture[KS04]. As lipids, alcohols are amphiphilic molecules, that assemble at the hydrophilic-hydrophobic interface, *i.e.* they go between the head groups, effectively increasing the average head-head distance and resulting in the interdigitated phase.

At low temperature and high pressure, the system exhibits an additional subgel phase (*Gel III*) with monoclinic chain packing.

4.3.3. Lipid type dependence

Clearly, the lipid type has a profound influence on the phase behaviour of the lipid bilayer.

The length of the hydrophobic tails influences the transition temperature: the longer the hydrophobic tails, the higher the phase transition temperature. However, the tail length can also completely suppress some phases or transitions. For example, there is no difference between the gel and subgel phase ($L_{\beta'}$ and L_c) for PCs with chains that have less than 13 acyl groups. The interdigitated phase $L_{\beta I}$ strongly depends on the interaction of the head groups. For long tails, this interaction is less important, and therefore the interdigitated phase can be seen for DPPC bilayers at high pressures. Furthermore, the ripple phase is strongly influenced by the chain length (see chapter 8).

When the lipid contains an unsaturated hydrocarbon tail, the normal zig-zag structure of the chain is broken, as the bond angle of the groups surrounding the double $C = C$ bond changes. Furthermore, the tail can not rotate around the bond. Therefore, a double bond can be thought of as a kink in the tail. Such a kink in the otherwise very regular aliphatic chain perturbs the packing of the chains in the ordered gel and subgel-phases and decreases the main transition temperature significantly. Branched tail chains have an effect very similar to unsaturated tail chains. However, branched tails also increase the average cross section of the tail and consequently change the packing parameter S.

The nature of the head group has a profound influence on the ordered gel and subgel phase structures and the subtransition temperature, as these strongly depend on the head group interaction. Additionally, the head group controls whether the gel phase is tilted or untilted and therefore whether a ripple phase exists or not. Also, the interdigitated gel phase $L_{\beta I}$ can be observed in bilayers of lipids that have sufficiently large heads, *e.g.* dialkyl phosphatidylcholines.

Finally, the linkage between the tails and the glycerol backbone has an influence on the phase behaviour. When the tails are linked to the backbone by alkyl groups instead of acyl groups, the gel phase L_β is replaced by the interdigitated gel phase $L_{\beta I}$. This can be explained by the fact that while PCs with acyl linkage can build hydrogen bonds between the acyl oxygens that usually stabilises the gel phase L_β, this can not happen in PCs with alkyl linkage.

4.4. Conclusions

In this chapter, the basics of the chemistry and physics of lipids have been discussed. Lipids are amphiphilic molecules with a hyrdophilic head and a hydrophobic tail moiety. In watery environment, depending on the shape of the lipid molecules, they form various aggregates, like micelles and bilayers. Furthermore, lipid bilayers can occur in several different phases.

5. Modelling a lipid bilayer

This chapter is devoted to describing the lipid bilayer model used in this work. First, a very brief introduction to two basic approaches to system modelling are given (section 5.1). Second, computer models of lipid bilayers used for other purposes are investigated (section 5.2). The bilayer model used in this work consists of two subparts. The model of the lipids that form the bilayer and the model of the watery solvent environment are described in sections 5.3 and 5.4, respectively, and have been published in [LS05, SDLW07]. The combination of the phantom solvent model and the bilayer model is referred to as the *bilayer reference model* and it is summarised in section 5.5 on page 60.

5.1. Top-down and bottom-up modelling

The goal of modelling a system is to reproduce a specific behaviour of the system. In general, two very different approaches to modelling can be used.

Bottom-up models try to reproduce the behaviour of a specific system by mimicking the system's degrees of freedom and the interactions between these as exactly as possible. These kind of models can be used to answer diverse, specific questions on the modelled system. However, as bottom-up models typically employ complex interactions and a vast number of model parameters, results can not easily be transferred to other, related systems. Furthermore, it is difficult to deduce the properties of the system that actually led to the observed behaviour. Bottom-up models can be coarse-grained by reducing the number of degrees of freedom to yield *effective* degrees of freedom. This process is referred to as *systematic coarse-graining* and requires great care and a very good understanding of the system, as only those degrees of freedom should be removed that do not influence the system's behaviour.

On the other hand, *top-down models* try to reproduce the system's behaviour with as few model parameters as possible. Because it only has a few model parameters, the effect of every single parameter on the system's behaviour can be systematically checked. The results of the model are generic and can be transferred to many different systems, as long as the model parameters can be mapped onto the system's parameters. However, the results are naturally not as detailed and exact as the results of bottom-up models.

To summarise, while bottom-up models can *predict* the quantitative behaviour, top-down models can help to *understand* the qualitative behaviour of a system.

5.2. Other bilayer models

Very different bilayer models have been employed for many different length and time scales. In the following, several models that have been used on the relevant length scale given length scale of a few tens of nanometers are described. As the phenomena examined in this work depend on the translational and conformational degrees of freedom of the lipids, only *off-lattice models* that possess explicit representations of lipid molecules moving in continuous space and interacting with each other via interaction potentials are taken into account. To simulate these models, both the molecular dynamics and the Monte-Carlo method can be used.

In the group of off-lattice models, *bead-spring models* are the most prevalent. In these models, mostly spherical beads act as single atoms or groups of atoms. Several beads are bound to other beads via spring-like potentials that mimick molecular bonds to build chains or networks that represent molecules. The interaction potentials between the different beads types can be of varying complexity. Especially in bottom-up-models, these interactions can become very complex.

5.2.1. Bottom-up bead-spring models

Atomistic bead-spring models are bottom-up models. In *all-atom models*, every single atom is represented by a bead that interacts with the other beads via complex potentials that model the atomic interactions. This includes the lipid molecules as well as the water molecules. In a *united-atom model*, the degrees of freedom of hydrogen atoms are typically removed and included into the interactions of the heavier atoms. During the recent years, many properties of specific lipid bilayers have been investigated by atomistic simulations (see the reviews in [Sco02, TTK97, TMB97]). The largest systems contain a few hundreds of lipids and thousands of water molecules and have been simulated over a time of a few tens of nanoseconds.

In *bottom-up coarse-grained bead-spring models*, each of the beads represents a specific group of atoms of the modelled lipid that interacts via carefully chosen interaction potentials with the other beads. Likewise, non-bonded beads act as a group of water molecules [KNS04, SSR+01a, MVM04]. This type of models was mainly applied to exploring the phase behaviour and structure of lipid bilayers [KVS03a, KVS03b, KS04, KLS04, SSR+01b].

5.2.2. Top-down bead-spring models

Top-down coarse-grained bead-spring models are usually less detailed than bottom-up bead-spring models. They contain fewer parameters per bead type, fewer bead types and fewer beads per lipid than corresponding bottom-up models. Also, it is normally not possible to map a bead onto a specific group of atoms. Several different levels of coarse-graining have been used. In some of the top-down models, a single bead chain is even thought to represent a group of a few lipid molecules [NT01b, CD05].

Top-down coarse-grained bead-spring models have been applied to a broad range of problems, from investigations of fluctuations, defects and pores in bilayers [LMKS03, LMS04, LMS05], to fusion, deformation, adhesion and self-assembly of vesicles [NT01a, NT01b, NT02b, NT02a, Nog02], to number only a few.

The basic coarse-grained off-lattice bead-spring model of a lipid that produces a stable bilayer is composed of two bead types, *head beads* and *tail beads*, that represent the lipid head and tail groups, respectively. Several of these beads are bound together to form a lipid chain. Non-bonded beads interact via a repulsive soft-core interaction to model the steric repulsion. To model the hydrophobic attraction between two tail beads, a potential that has an attractive tail is used. This simple lipid model was first employed by Götz and Lipowsky [GL98]. Almost all coarse-grained bead-spring lipid models are relatives of this basic model that vary in the number of tail and head beads, the exact potentials and the parameters of the potentials.

A system consisting of a number of model lipids without any solvent model already produces a stable bilayer in the gel phase L_β or $L_{\beta'}$. However, the fluid phase L_α of the bilayer is only stable in a narrow temperature range and only when the bead chains are long enough. To obtain a stable bilayer in the fluid phase as well as the main transition between these phases, a *solvent environment model* for the watery surroundings of the bilayer is required, or more precisely a representation for the hydrophobic interaction, that keeps the lipid tails together. In the following, a number of such models are introduced.

Explicit solvent models

In many coarse-grained bead-spring top-down models, the watery environment is represented by additional, non-bonded *explicit solvent beads*.

These models extend the basic model described above by non-bonded head beads that act as the solvent environment (for one of the first examples, see [GL98]). Unfortunately, this simple approach has a number of disadvantages.

On the one hand, the model is too simple to describe any interesting hydrodynamic effects of the solvent, as it completely neglects the structure of fluid water. In fact, the model may even exhibit some unwanted artefacts in the solvent structure caused by the packing properties of the beads. Also, a huge number of solvent beads is required to inhibit any indirect interaction of the periodic images of the bilayer with itself or of the bilayer with the walls of the system box. On the other hand, explicit solvent beads are computationally very expensive as they add a huge number of degrees of freedom and interactions between these degrees of freedom that need to be simulated. In many cases, the simulation of the explicit solvent almost takes half of the computing time required by the model, even though its behaviour is not only of no interest, but also has a number of unwanted characteristics.

Surface potential solvent models

A very simple approach to overcome the disadvantages of the explicit solvent model is to use a *surface potential solvent*. This is done by adding a fixed potential to the model that forces the head beads to remain at the bilayer-water interface and the tail beads to reside inside the bilayer [SB98]. This method is extremely cheap from a computational point of view. However, the method can not be used to investigate phenomena like the hydrophobic mismatch or surface undulations as the bilayer surface is completely rigid.

Implicit solvent models

A more elaborate solution to the problem are so-called *implicit solvent models* or *solvent-free models*. These models incorporate the hydrophobic interaction into the interactions of the lipid beads. The first of these models has been devised by Noguchi and Takasu [NT01b], who use an additional multi-body term in the interaction potential of the head beads. In the model by Farago [Far03], non-additive potentials between the different bead types are employed. Cooke, Deserno and Kremer [CKD05, CD05] use a tail-tail interaction potential that has a broader attractive tail than the standard 12-6-Lennard-Jones potential. Brannigan, Phillips and Brown [BPB05] add an additional *interfacial bead* type to each lipid that has a long range attractive potential.

All of these models have been shown to exhibit a stable fluid phase and for most of them it has also been shown that a system of free lipids self-assembles into a bilayer. It should also be noted that all of these models incorporate interactions that have a longer range than the models with explicit solvent, and are therefore not necessarily computationally more efficient than models with a computationally very cheap explicit solvent model.

5.2.3. Molecular models

Molecular models are less detailed than bead-spring models. In these models, the lipid molecules are represented by non-spherical objects, like spherocylinders or spherocones [DML91, BB04]. This type of model is especially well suited to investigate the effect of the molecular shape on the phase behaviour of lipids [BTB04].

5.3. Lipid model

The lipid model employed in this work was created for the investigation of lipid-mediated interactions of membrane-integral proteins on intermediate length scales of a few tens of nanometers or more.

The structure and dynamics of biomembranes on these length scales is not yet very well understood, as the length scale is still difficult to access experimentally [MJ97],

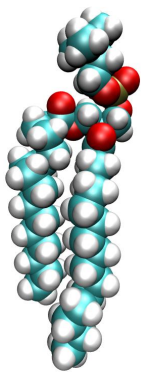
and atomistic models are still computationally too costly to be feasible. Therefore, a coarse-grained top-down model had to be developed.

The focus of the model was put on the influence of the bilayer's phase behaviour close to the main transition on these lipid-mediated interactions. As the main transition (also called lipid tail order-disorder transition) of a bilayer is mainly driven by the ordering and packing properties of the hydrophobic lipid tails, it was necessary to model the conformational and translational degrees of freedom of the lipid tails as well as the overall lipid shape. All further molecular details have been neglected, like the nature and shape of the hydrophilic head group, the zigzagged structure of the hydrocarbon tails or even the fact that the lipids in biomembranes usually possess two hydrophobic tails.

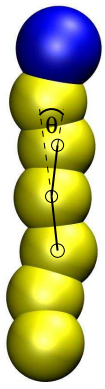
The coarse-grained model type best suited to match these requirements are bead-spring models. A great number of models of this type have been used before with great success, for example by Stadler, Lange, Düchs and Schmid [SL97, Sta98, SSL99, SS99, DS01] for the investigation of the phase behaviour of Langmuir monolayers. Therefore, the lipid model used in this work is a direct descendant of the basic model described above ans is, in fact, identical to the model used by Stadler, Lange, Düchs and Schmid.

In the following, we will define the *bilayer reference model* used in this work. When not explicitly noted otherwise, all measurements and observed facts refer to this model.

A single bead-spring chain is thought to roughly correspond to one DPPC molecule (see figure 5.1a). In the reference model, a model lipid consists of six tail beads and one slightly larger head bead (figure 5.1b). The interactions in the model have been chosen to be as simple as possible.

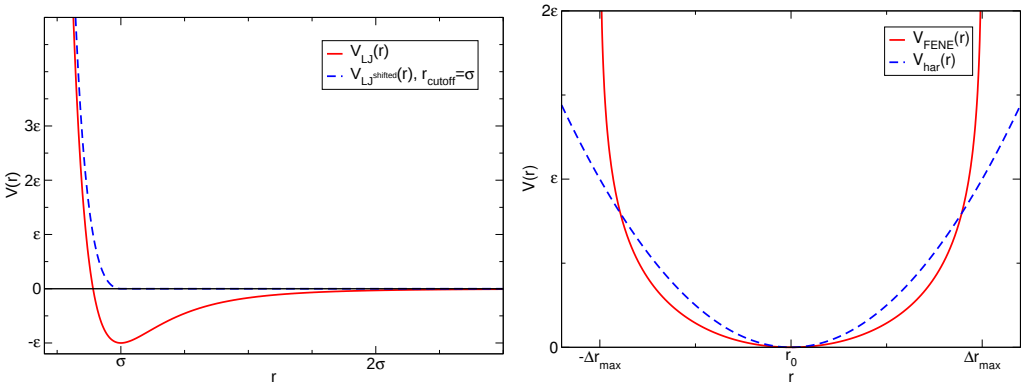


(a) All-atom model of DPPC



(b) Coarse-grained bead-spring model used in this work

Figure 5.1: Lipid models on different levels of coarse-graining.



(a) Nonbonded potentials: pure 12-6-Lennard-Jones potential V_{LJ} (equation 5.1) and the soft-core potential, *i.e.* the truncated Lennard-Jones potential $V_{\mathrm{LJ}}^{\mathrm{shifted}}$ (equation 5.2) with $r_{\mathrm{cutoff}} = \sigma$.

(b) Bonded interaction potentials: FENE potential V_{FENE} and the harmonic spring potential V_{har}.

Figure 5.2: Potentials used in the model

5.3.1. Nonbonded interactions

Non-bonded beads interact via a *12-6-Lennard-Jones potential* V_{LJ} (equation 5.1, plot see figure 5.2a).

$$V_{\mathrm{LJ}}(r) = \epsilon \left(\left(\frac{\sigma}{r} \right)^{12} - 2 \left(\frac{\sigma}{r} \right)^{6} \right) \tag{5.1}$$

The potential is repulsive for $r < \sigma$ and it has a soft core that goes to infinity for $r \to 0$, *i.e.* it serves as an excluded volume interaction. For distances of $r > \sigma$, the potential is attractive but the value and its derivative vanish with growing distance. In practice, one uses a variant of the pure Lennard-Jones-potential V_{LJ}, the *truncated 12-6-Lennard-Jones potential* $V_{\mathrm{LJ}}^{\mathrm{shifted}}$ (equation 5.2).

$$V_{\mathrm{LJ}}^{\mathrm{shifted}}(r) = \begin{cases} V_{\mathrm{LJ}}(r) - V_{\mathrm{LJ}}(r_{\mathrm{cutoff}}) & , \text{if } r < r_{\mathrm{cutoff}} \\ 0 & , \text{otherwise} \end{cases} \tag{5.2}$$

In this variant, the potential is cut off at the distance r_{cutoff}, giving the potential a finite range. When it is chosen to be $r_{\mathrm{cutoff}} = 2\sigma$, the potential is already very close to the pure Lennard-Jones potentials. However, when the cutoff is chosen to be $r_{\mathrm{cutoff}} = \sigma$, only the repulsive core of the potential is used. This variant is sometimes called the *soft-core potential* (plot see figure 5.2a).

In the lipid reference model, only the interaction between non-bonded tail beads includes the attractive tail if the Lennard-Jones potential ($r_{\mathrm{cutoff}} = 2\sigma$). The interactions between two non-bonded head beads and the interaction between a head and a tail bead is the purely repulsive soft-core of the potential with $r_{\mathrm{cutoff}} = \sigma$.

Throughout all of this work, the values of the Lennard-Jones σ for the different interactions have been chosen such that they are pairwise additive, *i.e.* a well-defined radius can be assigned to each bead type.

5.3.2. Bonded interactions

The adjacent beads of a lipid chain are bound to each. The bond is modelled by a FENE type spring potential V_{FENE} (equation 5.3, plot see figure 5.2b).

$$V_{\mathrm{FENE}}(r) = -\frac{1}{2}\epsilon(\Delta r_{\max})^2 \log\left(1 - \frac{r - r_0}{\Delta r_{\max}}^2\right) \tag{5.3}$$

Close to the equilibrium length where $r = r_0$, this potential is very similar to a simple harmonic spring potential (equation 5.4), but it limits the maximal and minimal bond length so that $V_{\mathrm{FENE}}(r) \to \infty$ for $r < r_0 - \Delta r_{\max}$ and $r > r_0 + \Delta r_{\max}$.

$$V_{\mathrm{har}}(r) = \epsilon(r - r_0)^2 \tag{5.4}$$

In the lipid reference model, adjacent beads interact *only* via the bond-length FENE potential, and there is no Lennard-Jones interaction between them. Only during the first studies done for this work, a lipid model is used that additionally employs the Lennard-Jones interaction between neighbouring beads. This model differs from the lipid reference model mainly in a slightly different equilibrium bond length between two bonded beads and consequently the overall lipid chain length. While the length of a single bond is exactly equal to r_0 in the definition of V_{FENE} in the first variant, it is slightly larger in the second. This has some influences on the phase behaviour of the models (see section 7.3 on page 84).

Models with fully flexible bead-spring chains poorly reproduce some of the properties of bilayers, like the average area per lipid A. Therefore, a bond-angle potential V_{BA} was added that favours stretched chain conformations:

$$V_{\mathrm{BA}}(\theta) = \epsilon\,(1 - \cos\theta) \tag{5.5}$$

5.4. Solvent environment model

As depicted above, a solvent environment model is required to stabilise the fluid phase L_α of the lipid bilayer. As the modelling of the head groups of the lipids, *i.e.* the interface between the solvent environment and the lipid bilayer, is very coarse, it was not deemed necessary to model the solvent environment in great detail. Therefore, the solvent models were designed to be very efficient from a computational point of view.

In this work, two different solvent models were used. However, the first of these models, the *surface potential model*, lacks a few of the basic features that are required for the examination of the phenomena that were the original subject of this thesis.

Therefore it was only used in a few simulation experiments to determine the model's phase diagram, while most of the simulations employed the second model, the *phantom solvent beads*. Therefore, the bilayer reference model uses the phantom solvent beads.

5.4.1. Surface potential solvent environment model

In the first solvent environment model, the bilayer is constrained by two parallel planes. The lower plane is the x-y-plane itself, the upper plane is shifted by $z_{\mathrm{upper}} \geq 0$. The tail beads of the bilayer are confined between the planes by the surface potential V_{ST} (equation 5.6), while the head beads are forced to stay above the upper plane respectively below the lower plane by V_{SH} (equation 5.7). The parameters have been chosen arbitrarily to be $\epsilon = 10$, $r_0 = 0$ and $\Delta r_{\max} = 0.5$).

$$V_{\mathrm{ST}}(r) = \begin{cases} V_{\mathrm{FENE}}(z) & ,\text{if } z < 0 \\ V_{\mathrm{FENE}}(z - z_{\mathrm{upper}}) & ,\text{if } z > z_{\mathrm{upper}} \\ 0 & ,\text{otherwise} \end{cases} \tag{5.6}$$

$$V_{\mathrm{SH}}(z) = \begin{cases} V_{\mathrm{FENE}}(z) & ,\text{if } 0 < z < \frac{1}{2} z_{\mathrm{upper}} \\ V_{\mathrm{FENE}}(z - z_{\mathrm{upper}}) & ,\text{if } \frac{1}{2} z_{\mathrm{upper}} < z < z_{\mathrm{upper}} \\ 0 & ,\text{otherwise} \end{cases} \tag{5.7}$$

This solvent environment model is very simple to implement and is also very efficient when it comes to computing. Unfortunately, the bilayer is not flexible in this model, *i.e.* it can not undulate or deform on longer length scales. Therefore, it is useless for the simulation of phenomena that involve any membrane deformations, such as undulations or hydrophobic mismatch effects of membrane integral proteins. As these phenomena are the main goal addressed by this work, the surface potential model was used only during the first simulation studies performed for this work and is seen as a variant of the bilayer reference model (see section 7.3 on page 84).

Note, that the upper plane position z_{upper} can drop to 0, as the soft potentials associated with the planes allow for beads to permeate up to a certain depth into the disallowed regions. When the volume of the system is defined to be $V = L_x L_y z_{\mathrm{upper}}$, this would actually allow the volume to drop to 0.

5.4.2. Phantom solvent beads

The phantom solvent bead model was developed to retain the full membrane flexibility, while still adding only a small computational overhead. In this model, the solvent is represented by explicit solvent beads. These beads behave exactly like additional, non-bonded head beads, *i.e.* they have a purely repulsive soft-core interaction with the lipid beads. What is special about these beads, however, is that they do not interact with each other.

The phantom solvent beads have two effects on the lipid bilayer. The main effect of the solvent is, that it mediates the external pressure onto the bilayer by means

of an excluded volume interaction: the phantom solvent probes the lipid-bead free volume, gouverned by the ideal gas law, thus excerting pressure onto the bilayer, which keeps the bilayer together and gives the system a preference for well-packed lipid configurations. This is an advantage over implicit solvent models, where external pressure can not be applied.

Another, more subtle effect of the phantom solvent is, that it creates an attractive depletion interaction between the lipid beads at the interface between the phantom solvent and the the lipid, yielding in a surface tension. As in the most cases, only head beads can be found at the interface, this results in an effective attractive interaction between the head beads. The strength of the interaction is controlled by the external pressure, while the range of the interaction is gouverned by the size of the solvent beads.

Note, that the depletion interaction only affects the beads at the solvent–lipid interface, *i.e.* the head beads, while the effect of the external pressure to minimise the system's volume in general affects the whole bilayer. Therefore, the relation between the head group attraction and the tail group attraction can be finely tuned by the choice of the pressure. This influences the general phase behaviour of the model, and the ripple phase $P_{\beta'}$ and interdigitated phase $L_{\beta I}$ in particular (see chapter 7 on page 73).

Using the phantom solvent in a computer simulation of lipids has a number of advantages. First of all, the model is still computationally very efficient: phantom solvent beads that are far from the bilayer do not have any interaction partner. In a Monte-Carlo simulation together with the cell-lists algorithm, this means that the energy computation for these beads can be skipped. Only those solvent beads that are actually close to the bilayer significantly contribute to the computing time. In fact, simple considerations show, that the efficiency of the phantom solvent environment model at low to intermediate pressures is comparable to the efficiency of the so-called solvent-free models that use an implicit solvent, as all of these models require longer ranged potentials.

In contrast to the surface potential model, the phantom solvent model does not put any constraints on the shape of the bilayer: the bilayer is fully flexible and can undulate or bulge. However, the surface tension induced by the depletion interaction between the interfacial beads leads to a preference for assemblies of the lipids.

Compared to a model of explicit, interacting solvent beads, the phantom solvent has the advantage, that the phantom solvent can not develop an internal structure. This rules out any artefacts in the bilayer behaviour caused by the solvent. Furthermore, only a relatively thin layer of solvent is required to screen the bilayer from interacting with either the system's wall or the bilayer's periodic image, depending on the type of boundary conditions used. If a thicker layer of phantom solvent beads is used in a constant volume simulation, the pressure of the system can be measured by measuring the solvent density far from the bilayer, as the ideal gas equation of state holds for the phantom solvent beads there.

It is also possible to use the phantom solvent environment model in a molecular dynamics simulation. In that case, it should even be possible to use a dissipative

particle dynamics (DPD) thermostat to study the solvent's hydrodynamics.

To summarise, the phantom solvent model is a simple way to excert external pressure onto the bilayer without putting any constraints onto the bilayers flexibility, without the disadvantages of solvent artefacts and with a computational effort comparable to that of implicit solvent models.

5.5. Bilayer reference model

The interaction potentials and parameters of the reference model are summarised in table 5.1. The tail-tail interaction potential was chosen as a reference to define the fundamental units of the system: all lengths in the system are expressed in units of the Lennard-Jones parameter σ, while all energies are expressed in units of the parameter ϵ. Refer to section 6.3 on page 66 for more details on the units.

The solvent employed in the bilayer reference model are phantom solvent beads of the same size as the head beads. The length of the chains (*i.e.* the number of tail beads) and the interaction parameter ϵ of the bond-angle potential V_{BA} were adapted to roughly correspond to DPPC molecules [Sta98, SSL99, DS01].

	potential	**parameters**
tail–tail	$V_{\mathrm{LJ}}^{\mathrm{shifted}}(\lvert\vec{r}\rvert)$	$\epsilon = 1$, $\sigma = 1$, $\frac{r_{\mathrm{cutoff}}}{\sigma} = 2$
head–tail	$V_{\mathrm{LJ}}^{\mathrm{shifted}}(\lvert\vec{r}\rvert)$	$\epsilon = 1$, $\sigma = 1.05$, $\frac{r_{\mathrm{cutoff}}}{\sigma} = 1$
head–head	$V_{\mathrm{LJ}}^{\mathrm{shifted}}(\lvert\vec{r}\rvert)$	$\epsilon = 1$, $\sigma = 1.1$, $\frac{r_{\mathrm{cutoff}}}{\sigma} = 1$
head–solvent	$V_{\mathrm{LJ}}^{\mathrm{shifted}}(\lvert\vec{r}\rvert)$	$\epsilon = 1$, $\sigma = 1.1$, $\frac{r_{\mathrm{cutoff}}}{\sigma} = 1$
tail–solvent	$V_{\mathrm{LJ}}^{\mathrm{shifted}}(\lvert\vec{r}\rvert)$	$\epsilon = 1$, $\sigma = 1.05$, $\frac{r_{\mathrm{cutoff}}}{\sigma} = 1$
solvent–solvent	0	(no interaction)
bond-length	$V_{\mathrm{FENE}}(\lvert\vec{r}\rvert)$	$\epsilon = 100$, $r_0 = 0.7$, $\Delta r_{\max} = 0.2$
bond-angle	$V_{\mathrm{BA}}(\theta)$	$\epsilon = 4.7$

Table 5.1: Summary of the interaction potentials in the reference model. $\vec{r}$ is the centre-to-centre distance vector of two beads, θ is the bond angle of three adjacent beads (as depicted in figure 5.1b).

5.6. Conclusions

This chapter was devoted to defining the lipid bilayer model that was used in this work. Initially, the difference between bottom-up and top-down modelling was clarified, to allow for the classification of the model used in this work. Then, bilayer models employed in other works were shortly revisited. The lipid model used in this work bases on a model employed previously by Schmid, Düchs, Lange and Stadler for the simulation of the phase behaviour of monolayers. To be able to form bilayers from the model lipids, two alternative solvent environment models were discussed. Amongst these is a new type of solvent model, the so-called *phantom solvent model*,

that has a number of attractive advantages over other models. Finally, the bilayer reference model used in the rest of this work was defined.

6. Simulation details

In this chapter, the details of the simulations that were performed in the frame of this work are described. The simulation algorithm is specified in section 6.1, while section 6.2 on the next page gives an overview of the typical parameters of the simulation runs that were performed. The observables that were measured in the system are discussed in section 6.3 on page 66.

6.1. Algorithms

The lipid bilayer model described in the last chapter was simulated using the well-known Metropolis Monte-Carlo algorithm in the constant pressure (N, p, T) ensemble. The system uses three-dimensional sheared rectangular periodic boundary conditions as described in section 3.2 on page 32.

A *Monte-Carlo step* (or *sweep*) consisted of a sequence of N *single bead moves* of a randomly chosen bead, with N being the total number of beads in the system. This ensured, that on average every bead was moved once per step.

For theoretical reasons, it is often demanded that lipid bilayer simulations maintain a neglectable surface tension. To reduce the surface tension and to maintain constant pressure, every few Monte-Carlo sweeps, a *volume move* in one of the three dimensions was done as described in appendix A on page 123. As a focus of this work lies on the more ordered lipid bilayer phases, where the lipid heads form lattice-like structures, special care was taken to reduce any shear stress resulting from a mismatch between the geometry of the simulation box and the lattice-like structure of the more ordered phases. Therefore, the simulation used sheared rectangular periodic boundary conditions as well as *shearing moves*[Sta98].

In many cases, the main simulation run was prepended by a *simulation prerun* that used *Loeding's acceptance rate adaption* to adapt the maximal move ranges and the acceptance rate [Sta98]. To speed up the simulation, a sheared cell lists data structure (see section 3.3 on page 33) was used. As the interaction range of the different bead types was very different, a separate cell lists data structure was used for every pair of bead types. When the system was simulated on a parallel computer, a parallel Monte-Carlo algorithm[ULA+02] was employed. The simulation was implemented in the `C++` programming language, using the Off-Lattice Template Library OLTL described in chapter 3 on page 29.

6.2. Simulation runs

When the simulation runs were not started from a configuration of another run, the system was set up artificially as a bilayer with untilted lipid chains (see figure 6.1). The head beads of the lipids were set up on a triangular grid. During a simulation, the average number of lipids per side does not change significantly – the relaxation time for this process is beyond the simulation run times – therefore, the number of lipids per side can be used to characterise the size of the system. This means, that a 12x12-lipid-system is a system were the lipids of one leaflet of the bilayer were set up on a grid with 12 lipids on each side, summing up to $144 \times 2 = 288$ lipids in total. When the phantom solvent model was used, the phantom solvent beads were placed randomly on both sides of the bilayer.

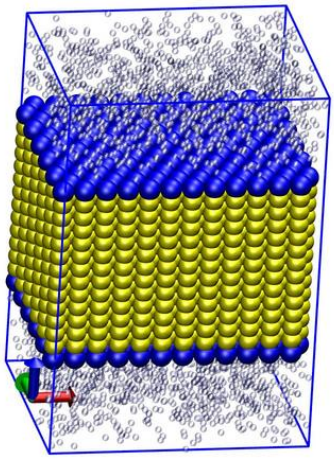


Figure 6.1: Configuration snapshot of an artificially setup 12x12-lipid-system.

Typical simulation runs employed between 288 lipids (12x12) for determining the phase behaviour of the system (see chapter 7) up to 3,200 lipids (40x40) for simulations of large rippled systems.

Before the main simulation run was done, a simulation prerun of a few thousand Monte-Carlo steps was performed to adapt the maximal move ranges of the different move types to an acceptance rate of 0.3. However, the maximal move range was capped to 0.5 to avoid very large maximal move ranges. This was necessary especially in the case of the phantom solvent bead moves. Figure 6.2 shows the adaption of the move ranges and the acceptance rates of a typical run.

An important parameter for the simulation runs was the number of Monte-Carlo steps required by the system to reach equilibrium. The system was said to be equilibrated when all observables of the system fluctuated about the equilibrium value, and none of the observables shows a trend. To be able to establish whether the system

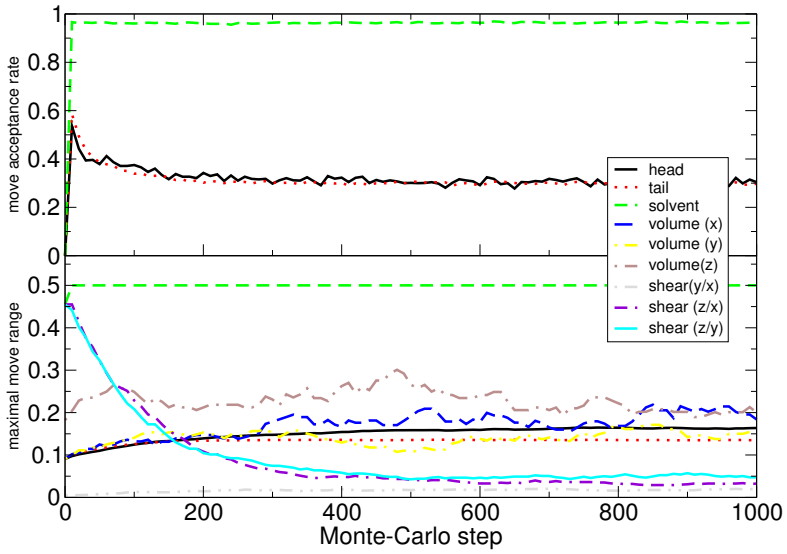


Figure 6.2: Plot of the adaption of the different acceptance rates and maximal move ranges over the number of Monte-Carlo steps performed during a typical simulation prerun.

was equilibrated or not, the observables with the *slowest* dynamics, *i.e.* the observable with the longest *relaxation times* hat to be watched. In the bilayer systems used in this work, the chain order parameter S_z and the total energy of the system E_{tot} turned out to have the longest relaxation times. A plot of the development of the total energy and the chain order parameter S_z of a typical simulation run is shown in figure 6.3.

Equilibration of the bilayer state typically took between 1,000,000 and 3,000,000 Monte-Carlo steps. Because of the *critical slowing down* close to the phase transitions, in some cases simulation runs with up to 7,000,000 Monte-Carlo steps were required. After equilibration, the simulations were run for at least 500,000 more steps to perform measurements of the different observables.

In some simulation runs that involve a phase transition to one of the more ordered gel phases, the otherwise very well-ordered bilayer shows defects, where a number of lipid chains are not incorporated into the bilayer, but form drop like assemblies that clinge to the bilayer (see for example figure 6.4 on page 67. If these defects do not change and grow within a few million Monte-Carlo steps, it is assumed that the states are metastable and represent local minima of the free energy, with an energy barrier that is too high, so that it would take too long for the system to reinsert the lipids into the bilayer. In this case, the simulation runs are discarded and a new run with the same system parameters is started, as the defects would strongly influence the value of several observables.

The collective moves (volume and shearing moves) were very expensive from a computational point of view. Therefore, the frequency of the collective steps was gradually decreased. It turned out that the total number of Monte-Carlo steps required for the

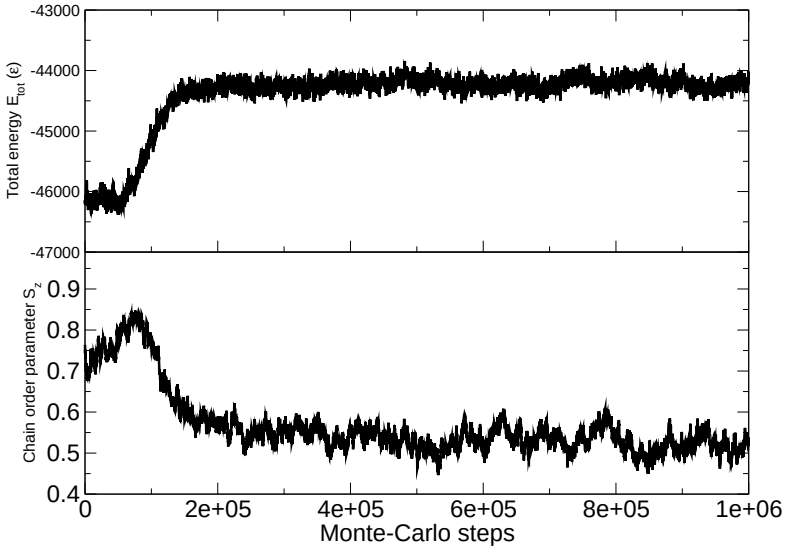


Figure 6.3: Plot of the total energy E_{tot} and the chain order parameter S_z over the number of Monte-Carlo steps in a typical simulation run.

equilibration of the system did not significantly increase when the collective moves were performed only every 50 Monte-Carlo steps. During the prerun, however, the collective moves were done in every Monte-Carlo step to get better statistics for estimating the acceptance rate required for the adaption of the maximal move range.

6.3. Observables

In the following, the different observables measured in the bilayer simulations are described and how they relate to experimental observables.

Wherever units are used, the dimensions are given in terms of the two fundamental simulation units: the length unit σ and the unit of energy ϵ, that equal the interaction parameters σ and ϵ of the tail-tail interaction potential, respectively. The conversion factors of these units to standard SI units are estimated in section 7.4.1 on page 87.

Throughout this work, in many cases the unitless reduced temperature T^* and the reduced solvent pressure p^* are used, which are defined by

$$T^* = \frac{T}{\mathrm{k_B}^{-1}\epsilon} \tag{6.1}$$

$$p^* = \frac{p}{\epsilon\sigma^{-3}} \tag{6.2}$$

Keep in mind, that despite the sheared box geometry, the bilayer plane is always parallel to the x-y-plane. Therefore, it can be safely assumed that the bilayer normal is parallel to the z-axis.

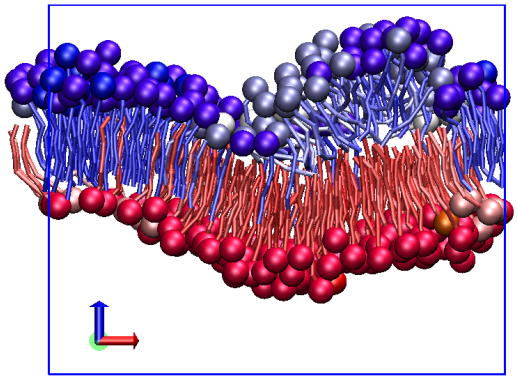


Figure 6.4: Snapshot of a bilayer at solvent pressure $p = 0.1\,\epsilon\sigma^{-3}$ and temperature $T = 1.05\,\mathrm{k_B}^{-1}\epsilon$ that was set up from the tilted gel phase $L_{\beta'}$. The system underwent a phase transition and got stuck in the process of transforming to the interdigitated gel phase.

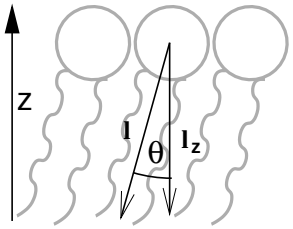


Figure 6.5: Lipid chain length l, l_z and tilt angle θ.

6.3.1. End-to-end vector, chain length and tilt

The end-to-end vector of a lipid chain is the vector from the head bead to the last tail bead of a lipid chain (see figure 6.5). The length of this vector is the *chain length* l.

In some cases, it is interesting to differentiate between the beads of the upper and the lower bilayer leaflets. On this behalf, the z-component of the end-to-end vector l_z can be measured, that is equal to the component along the bilayer normal. If l_z is positive, the lipid is defined to belong to the lower bilayer, otherwise it belongs to the upper bilayer.

The *average chain length* $\langle l \rangle$ is well suited for measuring the degree of chain stretching in the bilayer. When the average chain length is large, more chains are stretched, and there's less entropy in the chain conformations. This is typically a sign of the well ordered gel phases L_β, $L_{\beta'}$, while in the fluid phase L_α, the average chain length is smaller.

The tilt θ of a lipid chain towards the bilayer normal is defined by

$$\cos\theta = \frac{l_z}{l} \tag{6.3}$$

The *average tilt* $\langle\theta\rangle$ of all lipids in a system can be used to distinguish between the tilted and untilted gel phases $L_{\beta'}$ and L_{β}. The *local average tilt* along one of the axis can be measured by splitting the axis into bins, sorting the lipids into the bins and computing $\langle\theta\rangle$ for every bin separately.

6.3.2. Bilayer thickness and area per lipid

The *bilayer thickness* d is defined to be the difference between the average z-position of the head beads that belong to the upper bilayer leaflet and the z-position of those that belong to the lower leaflet:

$$d = \langle z\rangle_{\mathrm{head,upper}} - \langle z\rangle_{\mathrm{head,lower}} \tag{6.4}$$

In some cases, the *local bilayer thickness* d in the bilayer plane was measured by splitting the plane into bins, sorting the beads into the bins and computing equation 6.4 for every bin.

The average area per lipid A in the bilayer plane is the total area of the system in the x-y-plane divided by the number of lipids per leaflet.

Both the bilayer thickness and the area per lipid are quantities that can be measured experimentally. They are suitable for comparison between experimental and simulation results and were used to estimate the length unit conversion factor in section 7.4.1 on page 87.

6.3.3. Chain order parameter and nematic order parameter

The *chain order parameter* S_z is defined by

$$S_z = \frac{1}{2}\langle 3\cos^2\theta - 1\rangle = \frac{1}{2}\langle 3\left(\frac{l_z}{l}\right)^2 - 1\rangle \tag{6.5}$$

The chain order parameter measures how good the lipid end-to-end vectors are aligned with the z-axis. When all end-to-end vectors are parallel to the z-axis, S_z is equal to 1. When the end-to-end vectors have an isotropic distribution, S_z is equal to 0. The minimum value of $-\frac{1}{2}$ is reached when all lipids are orthogonal to the z-axis. As long as the lipids are part of a bilayer that lies in the x-y-plane, S_z is expected to be always greater than 0.

Another way to measure the lipid tail ordering is the nematic order parameter S. It is closely related to the chain order parameter S_z: while the chain order parameter measures the alignment of the chains with the z-axis, the nematic order parameter measures the alignment of the chains with the *director* of the chains, *i.e.* with the "main direction" of the chains. To compute both the director and the nematic order parameter, one can use the matrix

$$S_{ij} = \frac{1}{2N}\sum_{n=1}^{N_{\mathrm{lipids}}}(3x_i^{(n)}x_j^{(n)} - \delta_{ij}) \tag{6.6}$$

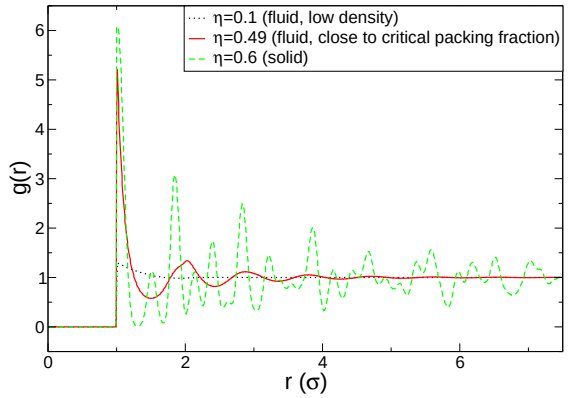


Figure 6.6: The radial distribution function $g(r)$ (RDF) of hard-spheres at different packing fractions. The radius of the spheres is σ. The system has a fluid – solid phase transition at a packing fraction of $\eta = 0.495$.

where x_i is i'th component of the end-to-end vector. The biggest eigenvector of S_{ij} is the director of the chains, and the corresponding eigenvalue is the nematic order parameter S [dGP93].

6.3.4. Radial distribution function

The *radial distribution function* (RDF) (or pair correlation function) $g(r)$ of two particle types A and B is a function of the distance r from a particle. The value of $g(r)$ at a given distance r to a particle of the type A is the density of particle type B at that distance, divided by the average density of the particle type B.

Figure 6.6 shows the radial distribution function of hard spheres in a hard-sphere system at different packing fractions η, obtained from canonical Monte-Carlo simulations.

The RDF reveals some information about the short-range structure of the system. For small distances, $g(r)$ typically approaches 0 because of excluded volume interactions between the particles. At contact distance, the value jumps to a finite value. Thereafter, the function decays in several oscillations. As can be seen in the figure, the range of the oscillations are different for systems in the fluid and the solid phase. In the solid phase, the amplitude of the oscillations decays only slowly with growing distance. In the fluid phase, the oscillations decay fast, while a clear structure at lower distances remains. Thus, the RDF gives evidence for the fact that fluids only show a short range order, whereas solids (i.e. crystal structures) also have a distinct long range order. The RDF contains all information on the pair structure of the system, although it cannot fully account for higher order contributions. All observables of the system that do only depend on pair correlations can be calculated when the RDF of the system is known.

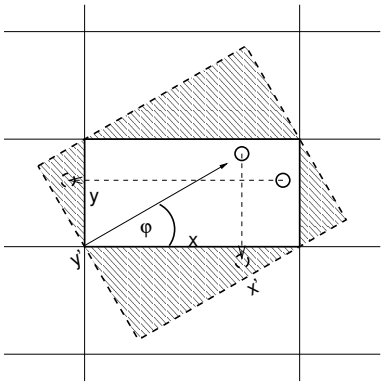


Figure 6.7: Sketch of the rotated system. The bead coordinates (x, y) are transformed in to the coordinate system (x', y') that is rotated by the angle ϕ. The beads in the "empty" corners are complemented by periodic images of the beads in the central image.

6.3.5. Density profiles

Plots of density profiles of the different bead types against one or two axes are useful to understand the overall structure and distribution of beads in the different bilayer phases.

Lateral density profiles show the density of the different bead types along the bilayer normal. They help to understand the lateral structure of the bilayer. How far can the solvent penetrate into the bilayer? Do the heads form a clearly defined layer, or do they intermingle with the tail groups? Do both leaflets overlap, or are they clearly dinstinguishable?

Measuring such density profiles in a simulation requires a number of steps. To compensate the net movement that the bilayer may make between different configurations, the system is first normalised, *i.e.* the origin of the system is shifted to the bilayer's center of mass. Otherwise, the density profile would be blurred by the net movement of the whole bilayer. Then, the used axes are split into bins. Each bin counts the number of beads of the appropriate bead type that fall into the bin, over a number of configurations. The density of a bead type can then be approximated by dividing the count per bin by the volume of the bin times the number of configurations.

Note, that in particular in the ripple phase $P_{\beta'}$ (see chapter 8 on page 93), two-dimensional density profiles in x'-z-direction were measured, where x' denotes an arbitrary direction in the plane of the bilayer, *e.g.* perpendicular to the ripple direction. On that behalf, the coordinates of the central image of all beads were transformed into a system rotated in the x-y plane by a given angle of ϕ. By itself, this would yield in a non-rectangular system, therefore the beads were complemented by periodic images to fill the resulting rectangular box (see figure 6.7).

6.3.6. Structure factor

The *structure factor* $Q(\vec{q})$ can be measured by

$$Q(\vec{q}) = \frac{1}{N}\left|\sum_{j=0}^{N} e^{i\vec{q}\vec{x_j}}\right| \tag{6.7}$$

where N is the number of beads and the $\vec{x_j}$ are the bead positions. It is closely related to the density measurements, and in fact $Q(\vec{q})$ can be obtained by Fourier transforming the density and taking the absolute value. However, the process is only reversible, when the phase information of the different $\vec{q}$-vectors can be recovered.

The structure factor is feasible for understanding the long-range structure of the lipid bilayer. When the structure factor at a given value of the wave vector $\vec{q}$ is large, this indicates the existence of a lattice-like periodic structure with periodicity $\vec{q}$, while there is no such structure when $Q(\vec{q})$ is close to 0.

Note, that when the structure factor is measured, it has to include all periodic images of the simulation box, and not just the central image. Effectively, this leads to an annihilation of the value of $Q(\vec{q})$ for most values of $\vec{q}$. $Q(\vec{q})$ has a finite value only for vectors on the *reciprocal lattice* of the simulation box size $\vec{L}$ ($\vec{q} \in \{\frac{\vec{i}}{\vec{L}} | \vec{i} \in \mathbf{N}^3\}$).

For visualisation purposes, projections of the structure factor on a plane are used, instead of the full three-dimensional structure factor $Q(\vec{q})$. Within the scope of this work, two kinds of projections have been used. The projection of the structure factor onto the bilayer plane, the so-called in-plane structure factor $Q_{xy}(\vec{q_{xy}})$, is useful to recognise crystal lattices in condensed phases, which show up as peaks in the structure factor. Weak cocentric rings, the so-called *Debye-Scherrer-rings* indicate weakly correlated, isotropic structures with a preferred distance between the particles.

The seconds type of projection of the structure factor used in this work is the *powder average* $Q_{rz}(\vec{q_{rz}})$. The advantage of the powder average is that it can be directly measured in certain types of scattering experiments. The scattered intensity of X-ray radiation at a given value of the wave vector $\vec{q}$ (which is referred to as the *momentum transfer*) is proportional to $Q(\vec{q})^2$. However, it is normally not possible to measure the full three-dimensional structure factor $Q(\vec{q})$. Small-angle X-ray scattering experiments (SAXS) obtain a one-dimensional projection of the structure factor onto the bilayer normal $Q_z(q_z)$, while wide-angle X-ray scattering experiments (WAXS) measure the powder average $Q_{rz}(\vec{q_{rz}})$, a two-dimensional projection of the structure factor.

To understand the powder average, it is necessary to understand that even the bilayer in one of the gel or subgel phases usually consists of domains with a constant tilt direction that are not much larger than a few μm. The focussing of an X-ray beam used in WAXS experiments is rather larger than that. Therefore, what is measured in the experiments is an average of the structure factor over a large number of possible directions with a fixed component of the momentum transfer vector in the bilayer plane, and the direction along the bilayer normal. To get the powder average in the simulations performed for this work, the structure factor was simply averaged over all possible directions in the plane.

Consequently, the powder average of the structure factor exhibits the detailed structure of the bilayer along the z-axis, while it can only show preferred distances in the bilayer plane.

Under certain circumstances, *electron density maps* (EDM) can be constructed from the structure factor (see section 6.3.6) obtained in scattering experiments [STNSN96]. As the electron density of the lipid head group is much higher than that of the lipid tails, EDMs can be roughly compared to density maps of the head groups obtained from the simulation.

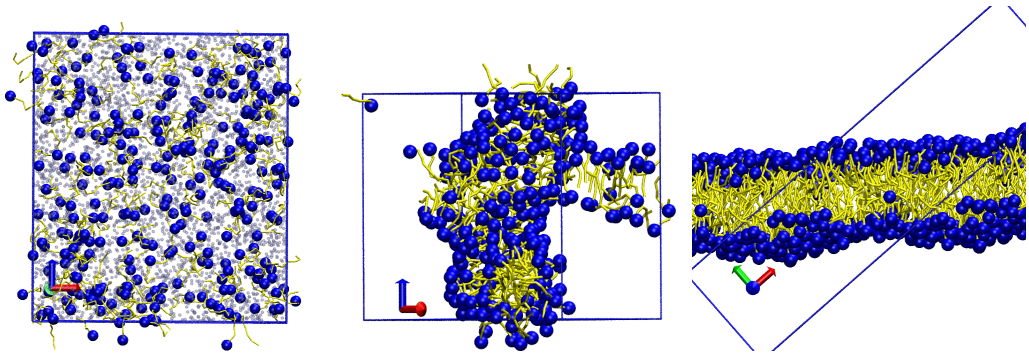
6.4. Conclusions

In this chapter, the details of the simulations performed in the frame of this work were discussed. The bilayer reference model is simulated using a constant-pressure Monte-Carlo simulation with sheared boundary conditions, which allows for simulations of a few thousand lipid molecules for a few million simulation steps. Finally, the various observables that were measured in the system were described.

7. Phase behaviour of the model

This chapter discusses the phase behaviour of the lipid model that was described in chapter 5. In the first section, it will be shown that a system of disordered lipids and phantom solvent beads self-assembles into a stable fluid bilayer. Section 7.2 on the following page discusses the phase behaviour of the bilayer reference model in detail, while the phases of some model variants are described in section 7.3 on page 84. In the discussion section 7.4 on page 87 the role of the phantom solvent and the head-head interaction in the phase diagram is focussed and the results are compared to experimental evidence from the literature. The results of this chapter have been partly published in [SDLL04].

7.1. Self-assembly



(a) Initial setup.

(b) After 300,000 MC steps, the lipids and the phantom solvent have demixed.

(c) After 3,000,000 MC steps, a stable fluid bilayer has formed.

Figure 7.1: Self-assembly of 288 model lipids at $T^* = 1.3$, $p^* = 2.0$, setup from a system of disordered lipids. The tail beads are replaced by tubes connecting their centers, the solvent beads are drawn only in plot 7.1(a), where they are scaled down.

Self-assembly of a fluid bilayer from a system of completely disordered lipids (see figure 7.1) is often seen as an important test for the feasability of a lipid model. Therefore, a reference model system of 288 completely disordered lipids has been prepared by heating a bilayer system to a very high temperature of $T^* = 5.0$. After a few thousand MC steps, the lipids are completely disordered.

When cooling down the system to $T^* = 1.3$, the lipids and the phantom solvent at

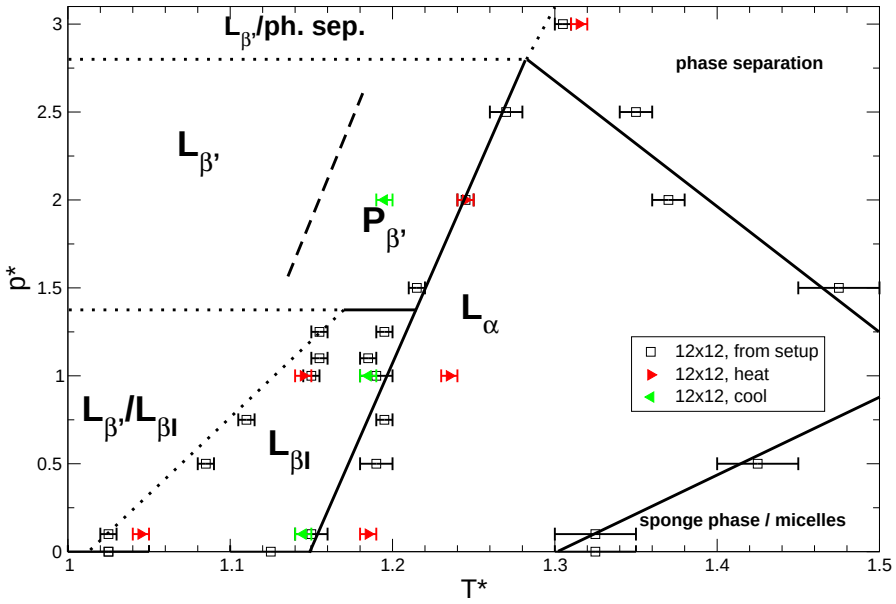


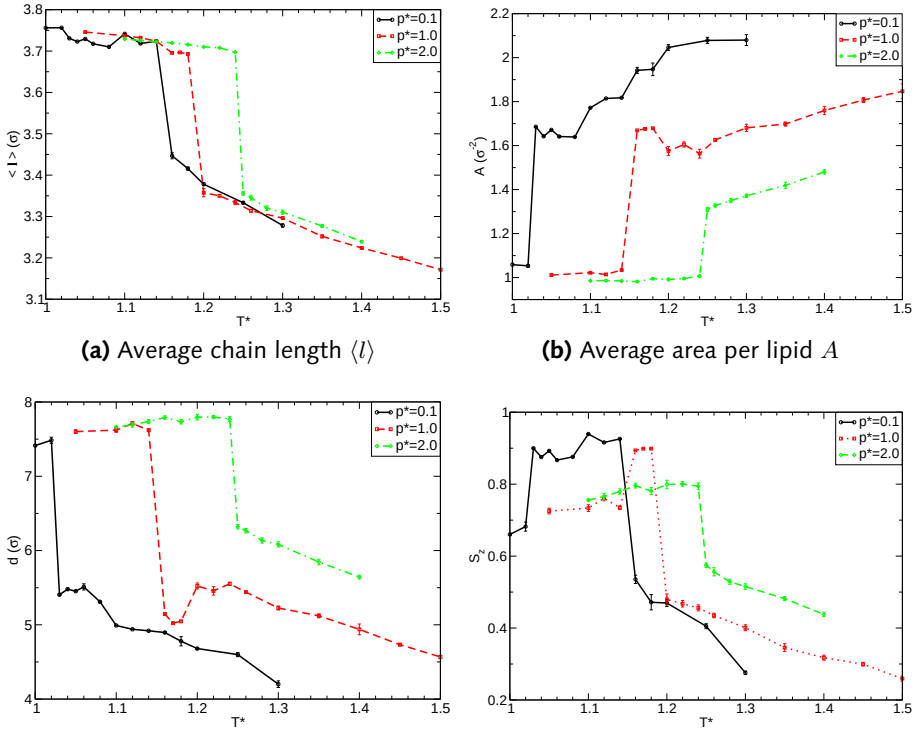
Figure 7.2: Phase diagram of the bilayer reference model. The lines are the estimated phase boundaries. See the text for an explanation of the different line types.

first demix within a few $100,000$ MC steps. Within a few million MC steps, the system has self-assembled into a stable, fluid bilayer.

7.2. Phases of the reference model

The phase behaviour of the system mainly depends on the temperature and the pressure. Figure 7.2 shows the phase diagram of the bilayer reference model with 7-bead lipids and phantom solvent beads. Figure 7.3 on the facing page holds plots of different observables against the (reduced) temperature T^* at a given (reduced) external pressure p^*, namely the average chain length $\langle l \rangle$, the average area per lipid A, the bilayer thickness d and the chain order parameter S_z. In the phase diagram, five phases of the model system can be distinguished, snapshots of which are depicted in figure 7.4 on page 76.

At intermediate pressures and low temperatures, the model system forms a well-ordered bilayer phase, that we will identify with the experimentally known tilted gel phase $L_{\beta'}$, as the characteristics of the phase match very well. A snapshot of the phase is shown in figure 7.4(a). In the phase, the lipids are mostly stretched, parallel to each other and well aligned with the bilayer normal, which is indicated by the relatively high average chain length $\langle l \rangle$ and chain order parameter S_z. The lipids have a distinct tilt towards the bilayer normal. Both bilayer leaflets are clearly distinguishable, and no overlap between the both leaflets occurs. Accordingly, the thickness of the bilayer is also high and almost two times the length of a stretched lipid. In the bilayer plane,



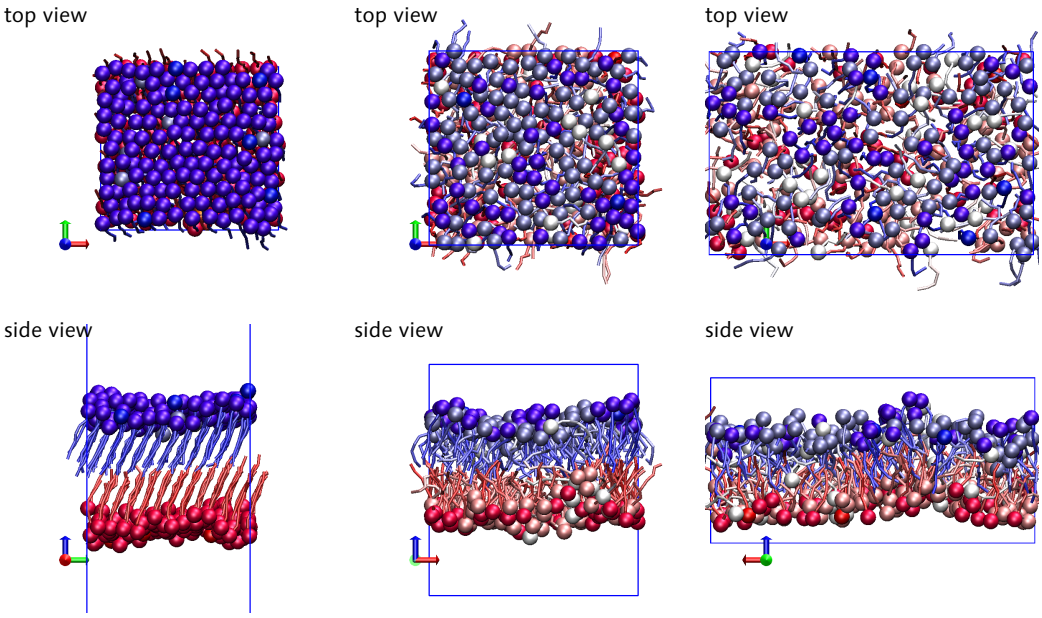
(a) Average chain length $\langle l \rangle$

(b) Average area per lipid A

(c) Bilayer thickness d

(d) Chain order parameter S_z

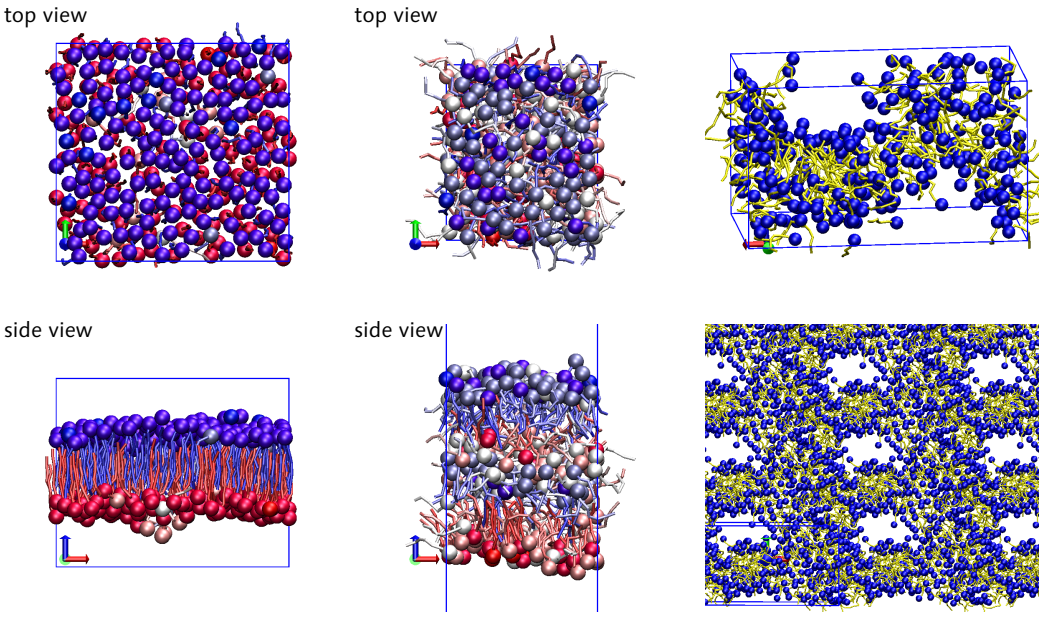
Figure 7.3: Equilibrium averages of different observables aainst the temperature T^* at different external pressures p^*



(a) tilted gel phase $L_{\beta'}$ ($p^* = 2.0$, $T^* = 1.16$)

(b) fluid phase L_α ($p^* = 2.0$, $T^* = 1.25$)

(c) fluid phase L_α ($p^* = 0.1$, $T^* = 1.25$)



(d) interdigitated phase $L_{\beta I}$ ($p^* = 0.5$, $T^* = 1.16$)

(e) phase separation ($p^* = 2.0$, $T^* = 1.45$)

(f) sponge phase L_3/ micelles ($p^* = 0.1$, $T^* = 1.4$)

Figure 7.4: Snapshots of the phases of 12x12-lipid reference model systems. The solvent beads are not drawn, the tails are replaced by tubes. In figures 7.4a to 7.4e, the colour of the lipids encodes the z-component of the end-to-end vector: Blue lipids point downwards, red lipids point upwards, white lipids have a small z-component. The ripple phase $P_{\beta'}$ is not shown here, but described in detail in chapter 8.

the lipid heads are relatively well ordered and packed, which is confirmed by the low average area per lipid A. The tail groups are very well ordered and densely packed.

Note, that in the side view of the snapshot in figure 7.4(a) the background can be seen shining through a patch of 12 lipids. This is an artefact of the regular lattice, where the direction of view is aligned with one of the main lattice directions. In fact, a corresponding direction can be found in all of the ordered systems, although it is not always aligned with the view direction of the snapshot and therefore not always apparent.

Between the tilted gel phase $L_{\beta'}$ and the main transition to the L_α-phase, another phase occurs under certain circumstances. This phase will be described in greater detail in chapter 8 on page 93 and it will be identified with the ripple phase $P_{\beta'}$. Here, it is sufficient to state that the phase is closely related to the titled gel phase $L_{\beta'}$ and the values of the most observables are comparable – the lipid chains are mostly well ordered and stretched, most of them are closely packed on a hexatic lattice and have a low average area per lipid.

Low pressures and low temperatures bring forth another well-ordered bilayer phase, that resembles the interdigitated gel phase $L_{\beta I}$ known from experiment (see figure 7.4d on the preceding page). Also in this phase, the lipids are stretched, well-packed and parallel to each other, which is suggested by the hight average chain length $\langle l \rangle$ and the high chain order parameter S_z. However, in contrast to the tilted gel phase $L_{\beta'}$, the bilayer leaflets overlap completely, and the bilayer has a low thickness d. The lipids are not tilted but parallel to the bilayer normal (which is reflected in the fact that the value of S_z is even higher than in the tilted gel phase $L_{\beta'}$). Although the lipid tails are very densely packed, the lipid heads have a relatively low density and a low average area per lipid A.

At higher temperatures, a less ordered phase is found, that matches the fluid phase L_α. In this phase, the lipids have a significantly lower average length $\langle l \rangle$, and they are not as well-packed as in the gel phase $L_{\beta'}$, indicated by the low value of the chain order parameter S_z. Instead, they are disordered and fluid-like, and on average not tilted. The thickness d of the bilayer is lower, and the average are per lipid A higher, than in the tilted gel phase, while being comparable to the interdigitated gel phase $L_{\beta I}$. When comparing both snapshots of the L_α-phase at two different values of the pressure, some differences meet the eye: at high pressure (see figure 7.4b on the facing page), both leaflets can still be clearly distinguished, and they seem not to overlap, while at lower pressure (see figure 7.4c on the preceding page), the leaflets seem to overlap and are not clearly distinguishable.

The separate regions at high temperatures and low respectively high pressures denote the occurence of non-bilayer phases. In the *sponge phase*, the lipids form interconnected assemblies, and not a flat bilayer anymore. From visual evidence, it is assumed that the phase corresponds to the experimentally known sponge phase L_3, but it may also be a phase of cylindrical micelles or as the cubic phase. In the regions labelled *phase separation*, the solvent beads and the lipids demix. As the lipids shield the tail beads from the phantom solvent beads, one can still recognise two lipid monolayers. However, a significant fraction of the lipids has left either of the

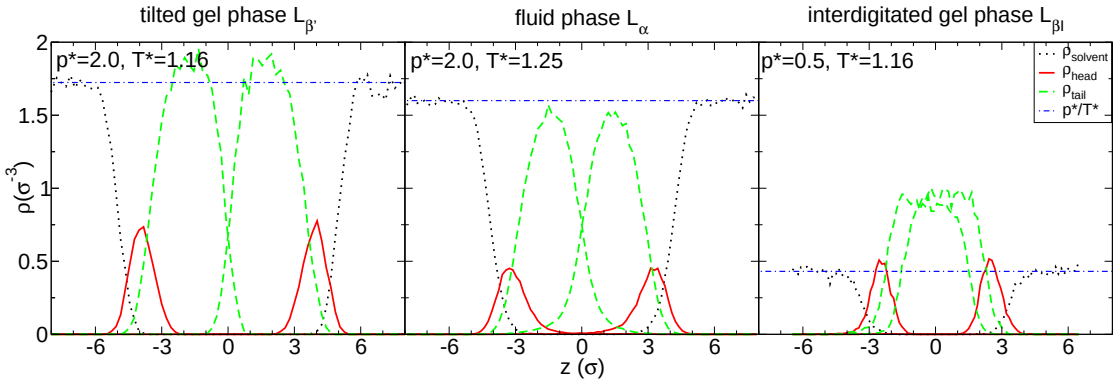


Figure 7.5: Transverse density profiles of the system components in the different bilayer phases. The dash-dotted blue line denotes the ideal gas pressure p^*/T^*.

monolayers and forms a disordered fluid of lipids between the two monolayers. As both of these phases were of no greater interest for this work, they have not been characterised in any detail.

When examining the dependency of the different observables within one phase in figure 7.3, a number of interesting observances can be made.

In general, within the gel phases $L_{\beta'}$ and $L_{\beta I}$, the pressure has no and the temperature only has a minor effect on the observables, while both strongly affect the observables within the fluid phase L_α.

In all phases, the average chain length $\langle l \rangle$ decreases with increasing temperature, which can be attributed to higher entropic demands. Consequently, the external pressure has almost no effect on the average chain length.

Within the tilted gel phase $L_{\beta'}$, it stands out, that the chain order parameter S_z slightly increases with increasing temperature, although an increase in S_z usually is associated with a higher alignment of the chains, which implies more order in the phase. However, note that the chain order parameter S_z measures the alignment of the chains *with the z-axis*. Therefore, the increase in S_z can also be explained by a drop in the average tilt angle $\langle \theta \rangle$ towards the z-axis. This assumption is confirmed by the fact, that the bilayer thickness d increases, while the average area per lipid A is constant.

Within the fluid phase L_α, the bilayer thickness and the chain order parameter decrease with increasing temperature, while the average area per lipid grows. This is consistent with the decrease in average chain length caused by the higher entropic demands. The effect of the pressure is exactly the reverse of the effect of the temperature - with increasing pressure, the average area per lipid is decreased, while the bilayer thickness and the chain order parameter increase.

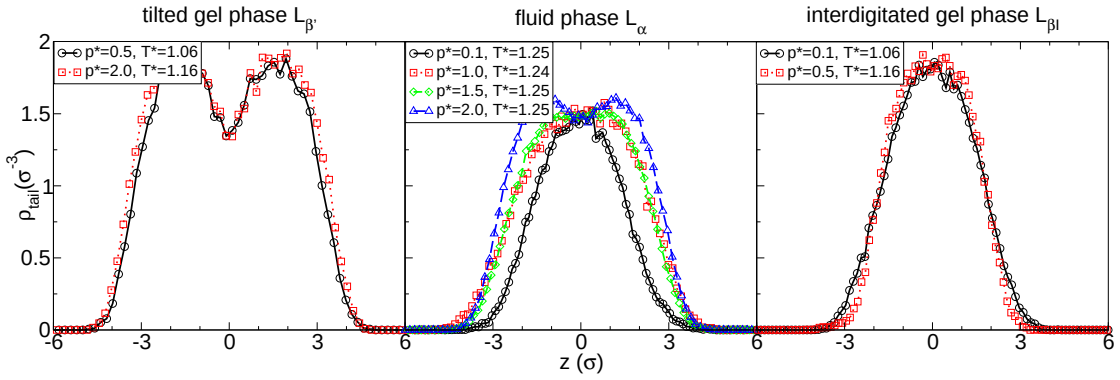


Figure 7.6: Transverse tail bead density profiles of different bilayer phases at different reduced temperatures T^* and pressures p^*.

7.2.1. Transverse bilayer structure

Figure 7.5 on the facing page shows the transverse density profiles of the different components and the different leaflets in the three bilayer phases. This leads to a number of observances. In all three phases, the solvent beads are clearly expelled from the tail region. The density of the solvent beads inside the bilayer is 0. Note also, that the solvent density has converged towards the ideal gas density $\frac{p^*}{T^*}$ at a distance of a bit more than 1 σ to the bilayer. Furthermore, although it is smeared out in the fluid phase L_α, the head bead region separates the solvent beads from the tail beads in all phases, and there is no significant overlap of the density profiles of the heads and the solvent beads.

The plot of the transverse tail bead density profile in figure 7.6 sheds some more light onto the structure in the tail region. First, it is apparent that the maximal tail bead density in the gel phases $L_{\beta'}$ and $L_{\beta I}$ is comparable and relatively high, while the maximal tail density of the fluid phase L_α is significantly lower, indicating the phase's fluidity.

In the tilted gel phase $L_{\beta'}$, the tail bead density profile has a minimum in the middle of the bilayer, which signifies two clearly separated leaflets with no overlap. In contrast to that, the profile of the interdigitated gel phase $L_{\beta I}$ has a maximum in the middle of the bilayer and strongly overlapping leaflets. In both gel phase, the tail bead density profiles do not differ significantly for different pressures and temperatures.

In the fluid phase L_α, however, there is a transition from high overlap of the leaflets at low pressures to no overlap at high pressures. This fits well to the observation, that at low temperatures and low pressures, the overlapping interdigitated gel phase prevails, while at higher solvent densities, only the tilted gel phase with two distinct leaflets occurs. Apparently, the pressure strongly influences whether the leaflets overlap or not.

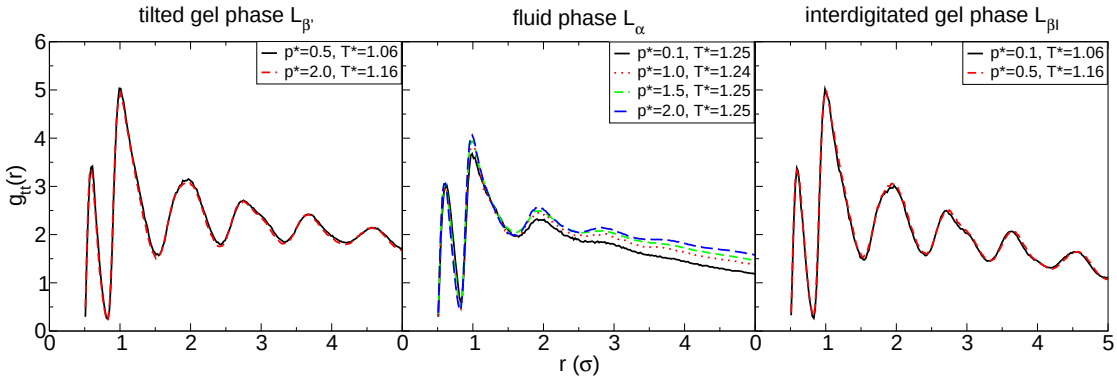


Figure 7.7: Tail-tail radial distribution function $g_{\mathrm{tt}}(r)$ of different bilayer phases at different reduced temperatures T^* and pressures p^*.

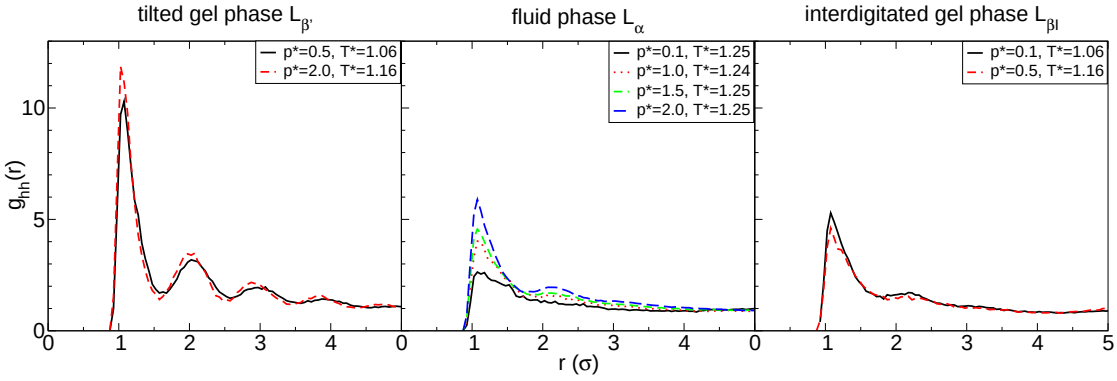
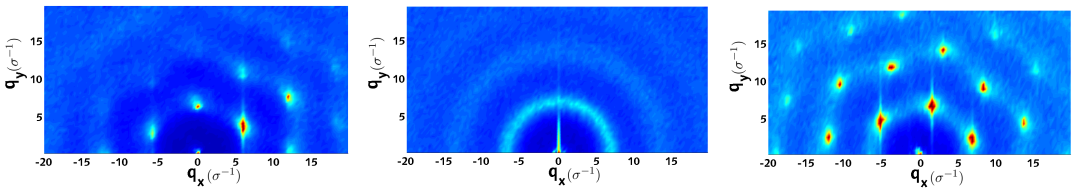


Figure 7.8: Head-head radial distribution function $g_{\mathrm{hh}}(r)$ of different bilayer phases at different reduced temperatures T^* and pressures p^*.



(a) $L_{\beta'}$ ($p^* = 2.0$, $T^* = 1.16$) **(b)** L_{α} ($p^* = 2.0$, $T^* = 1.25$) **(c)** $L_{\beta I}$ ($p^* = 0.5$, $T^* = 1.16$)

Figure 7.9: In-plane structure factor Q_{xy} of the different bilayer phases.

7.2.2. Short-range order

The plot of the tail-tail radial distribution function $g_{tt}(r)$ in figure 7.7 on the facing page shows the highly correlated fluid structure of the tail beads in the gel phases $L_{\beta'}$ and $L_{\beta I}$, and the much less correlated fluid structure in the fluid phase L_α. Note, that the first peak in the RDF at $r = 0.7$ corresponds to the bonded tail beads, and is of comparable height for all phases, while the peak at contact distance is lower for the less dense fluid phase L_α. Interestingly, the RDF is very much the same for different solvent densities and temperature values, as long as the phase boundary is not crossed. Only in the fluid phase a difference is seen for large distances. However, this can be explained by the changes in the thickness of the fluid phase at different solvent densities. Note, that the plots show the full three-dimensional RDF. As a bilayer is a sheet with a thickness of only a few σ, the RDF shows the structure known from RDFs of isotropic fluids only for small distances r. At larger distances, the two-dimensional structure of the bilayer can be observed in the linear decrease of the three-dimensional RDF. When the bilayer is thin, the decrease occurs more clearly, as in the fluid phase L_α or the interdigitatd gel phase $L_{\beta I}$.

The structure of the head region in the different phases is depicted in the plot of the head-head radial distribution function in figure 7.8 on the preceding page. While in the tiled gel phase $L_{\beta'}$, the head beads form a strongly correlated fluid, the correlations are suppressed in the fluid phase L_α and the interdigitated gel phase $L_{\beta I}$. The height of the peak at contact distance seems to depend mainly on the solvent density.

7.2.3. Long-range order

The in-plane long-range order of the lipid beads in the phases can be seen best in the in-plane structure factor Q_{xy} of the phases (see figure 7.9 on the facing page). In the gel phases $L_{\beta'}$ and $L_{\beta I}$, distinct peaks on a hexagonal lattice can be recognised, that give evidence of an underlying hexagonal lattice of the beads in real space. Note that only in the untilted interdigitated gel phase $L_{\beta I}$ all reflections in the hexagonal lattice are obvious, while in the tilted gel phase $L_{\beta'}$, not all reflections in the hexagon show up, because the peaks in the tilt direction do not lie in the plane of the bilayer. In the fluid phase L_α, no such peaks can be seen. Instead, weak Debye-Scherrer rings indicate the fluid structure of the phase.

7.2.4. Phase transitions and hysteresis

To understand the properties of the model system's phase transitions, different types of simulation runs close to the transitions have been performed. To be able to quickly sample the T^*-p^* plane, simulation runs of 12x12-lipid-systems have been started with artificial initial configurations that were setup by the procedure decribed in section 6.2 on page 64, and the temperature and pressure were directly set to the target values of T^* and p^*. After equilibration, the phase was determined. The open squares

in the phase diagram in figure 7.2 on page 74 mark the transition temperatures obtained by this method. In all simulations, the system quickly forms an $L_{\beta'}$-like state within a few 10,000 steps, with high chain ordering, high density and a tilt towards the bilayer plane. Therefore, the results can be interpreted as if the bilayer had been set up in the tilted gel phase $L_{\beta'}$ directly and was rapidly heated to the target temperature and pressure.

Unfortunately, the system shows distinct *hysteresis* effects, *i.e.* the transition temperatures vary depending on whether the system is heated up from a low-temperature phase or cooled down from a high-temperature phase, and also depending on the rate of cooling or heating. Therefore, in some cases additional simulations at selected temperature and pressure ranges have been performed. In these simulations, the initial configurations were equilibrated at temperatures well inside one phase. Then, the (reduced) temperature T^* was increased or decreased by small steps of 0.01, and the system was again equilibrated. These steps were repeated, until the phase transition occured. In the phase diagram, the freezing temperature of these simualtions is marked by triangles that point to the left, while the melting temperature is marked by triangles that point to the right.

Furthermore, *finite size effects* could influence the position of the phase transitions. In some cases, simulation runs with other system sizes have been performed. These runs have shown, that with the exception of the ripple phase $P_{\beta'}$, finite size effects seem to be of less importance than the above hysteresis effects.

The phase transitions to the non-bilayer phases at high temperatures for low respectively high pressures have not been studied.

Pretransition $L_{\beta'} \leftrightarrow P_{\beta'}$

The pretransition, where the bilayer transforms from the tilted gel phase $L_{\beta'}$ to the ripple phase $P_{\beta'}$, is usually ignored in coarse-grained simulations, as the ripple phase is not seen in these models and the exact nature of the phase is not known, anyway. The model used in this work is the first such model where the pretransition and the ripple phase was identified. The structure of the phase is described in detail in chapter 8 on page 93.

However, also in this work, it was not possible to accurately determine the position of the pretransition, and in the phase diagram in figure 7.2 on page 74, it is indicated only by a dashed line that refers to no data points. The main reason for this is the structure of the ripple phase itself. Because of the size of the ripple structure of a few tens of σ, the exact determination of the phase boundary would require the simulation of very large systems, that are computationally expensive. Furthermore, long simulation runs would be required, as the pretransition requires a highly cooperative transition between two ordered phases that yields large relaxation times. In the following, only a rough outline of the transition is presented.

In the standard simulations of 12x12-lipid-systems, the heating pretransition from $L_{\beta'}$ to $P_{\beta'}$ was never observed. Instead, the titled gel phase $L_{\beta'}$ directly transforms into the fluid phase L_α at the main transition. However, when the system is heated

gently, the ripple structure can be observed as an instable transitional state in the transition. In a sufficiently large system (*e.g.* 12x30 lipids) and very close to the main transition, an incomplete transformation of the tilted gel phase to the asymmetric ripple phase was observed (see chapter 8).

In none of the simulations performed for this work, the cooling transition from the ripple phase $P_{\beta'}$ to the tilted gel phase $L_{\beta'}$ occured. It can be assumed that the relaxation times for the transition are too long and the energy barrier between the phases is too high for such an event to occur within feasible simulation times. Note however, that simulation runs that were set up with an initial configuration in the fluid L_α-phase directly transformed into the tilted gel phase $L_{\beta'}$ when the cooling rate was high, *i.e.* the temperature was directly set to a temperature in the $L_{\beta'}$-domain. Therefore, the existance of the $L_{\beta'}$-phase in the phase diagram can safely be assumed.

Main transition $P_{\beta'} \leftrightarrow L_\alpha$

Of special interest for this work is the bilayer main transition (or chain order/disorder transition) between the mostly ordered ripple phase $P_{\beta'}$ and the fluid phase L_α at pressures of $p^* > 1$.

As was to be expected, at higher external pressures, the transition temperature increases, as higher external pressure favours the gel-like $P_{\beta'}$ state with higher density over the lower density L_α states. Many observables of the bilayer, *e.g.* the thickness d, the area per lipid A, the chain order parameter S_z, the average chain length l etc. undergo significant changes at the phase transition, even when the temperature step is very small, while they do almost not change within a phase, indicating that the transition is a first order phase transition.

The chain melting is very rapid, *i.e.* when the first remnants of disorder are visible in a completely ordered gel system, within a few 100,000 MC steps the whole system has transformed to a completely disordered fluid . Although the freezing process is slower and requires about 1,000,000 MC steps from the first signs of order, it also proceeds continuously without intermediate stages that mark transitional states.

Hysteresis was observed at the transition. At all examined pressures, the melting transition upon heating occurs at temperatures that were about $0.04\,\mathrm{k_B^{-1}}\epsilon$ higher, than the reverse freezing transition.

It should be stressed, that the heating simulations used initial configurations in the tilted gel phase $L_{\beta'}$, not in the ripple phase $P_{\beta'}$. However, some simulation runs suggested that this did not change the transition temperatures. The transition temperature in this case is very close to the transition temperature observed when the system is set up artificially and directly set to the target temperature. The reason for this is, that the artificially set up system first forms a transitional phase that corresponds to the tilted gel phase $L_{\beta'}$ and only then transforms to the fluid phase L_α. Therefore, the transition basically is identical to the main transition when the initial phase is the tilted gel phase.

The system size also influences the transition temperature: Larger systems have a melting temperature that is slightly lower than that of smaller systems, and a freezing

temperature that is identical or slightly higher. This can be explained by the fact that small system sizes suppress long-ranged undulations of the bilayer that destabilise the bilayer and lead to a lower melting temperature.

Transition $L_{\beta'} \leftrightarrow L_{\beta I}$

The boundaries of the order-order transition from $L_{\beta'}$ to $L_{\beta I}$ could not be clearly determined. In the region labelled $L_{\beta I}$, the system transforms from the titled gel phase $L_{\beta'}$ into the interdigitated $L_{\beta I}$-phase only via an intermediate, relatively stable L_α-like transitional state. The transformation to this intermediate state is as fast as the main transition, and occurs within a few 100,000 MC steps. The reordering transformation from the fluid-like state to the interdigitated gel phase $L_{\beta I}$ is much slower and typically takes about 1,000,000 MC steps. Also, the reordering often starts only after some million MC steps. Consequently, long simulation runs of some million MC steps were required to establish the correct phase. In the domain labelled $L_{\beta'}$/$L_{\beta I}$, the system does not do this transformation upon heating.

The heating transformation seems to be irreversible: when a system in the $L_{\beta I}$-domain is cooled down, the tilted gel phase $L_{\beta'}$ could not be recovered, instead the interdigitated phase is (meta-)stable. It should be noted that the low temperatures increase the relexation times.

Therefore, in the domain labelled $L_{\beta'}$/$L_{\beta I}$, the equilibrium phase of the system is not clear, and the exact line of the phase boundary could not be determined.

Neither finite size effects nor hysteresis of the transition have been studied. Again, the transition is of no greater interest for this work and has therefore not been investigated in greater detail.

Transition $L_{\beta I} \leftrightarrow L_\alpha$

The transition from the interdigitated gel phase $L_{\beta I}$ to the fluid phase L_α is very similar to the main transition above: it is a very rapid, very sharp transition and it shows a comparable hysteresis. Finite size effects in this case have not been studied, as the phase was of minor interest to this work.

In contrast to the main transition, the transition temperature for the artifically set up and tempered system in this case is mostly identical with the freezing transition temperature, and not to the melting temperature. The reason for this is, that the transition from the articifially set up system to the interdigitated phase has a transitional L_α-like state with a low chain order parameter S_z.

7.3. Model variants

Two variants of the bilayer reference model have been studied.

7.3.1. Longer bonds

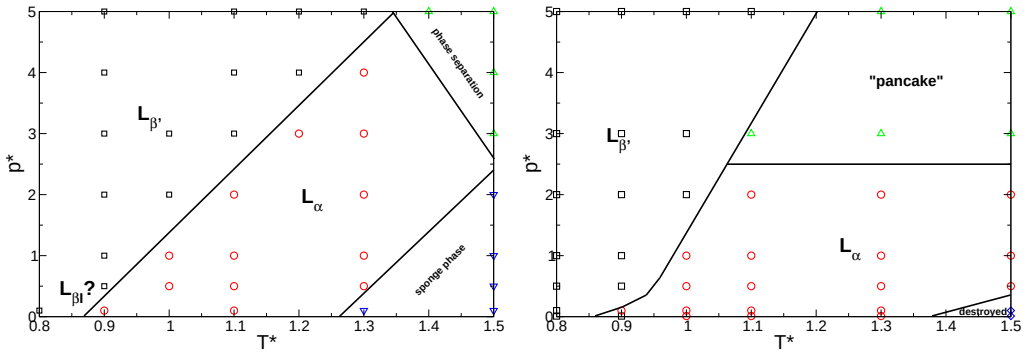
Figure 7.10(a) depicts the phase diagram of a model that differs from the bilayer reference model only in the fact that it takes into account the Lennard-Jones interaction between neighbouring lipid beads, additionally to the FENE bond-length potential. This results in a greater equilibrium bond length of $l_{\mathrm{bond}} \approx 0.847\sigma$ for tail-tail bonds. Keep in mind, however, that although the total length of a lipid is about 20% greater than in the reference model, the number of beads is the same. Consequently, the average interaction strength per unit of lipid length is lower, as the beads act as interaction sites. Furthermore, because the average bead distances are slightly higher, the average interaction per lipid bead is also decreased.

This lower binding energy between lipid chains results in the observed decrease of the transition temperatures compared to the reference model diagram in figure 7.2 on page 74. Qualitatively, the phase diagrams match pretty well. At high pressures, the low-temperature tilted gel phase $L_{\beta'}$ directly transforms into the fluid gel phase L_α when the temperature is increased. Even though it is not presented in this work, in this model the first remnants of the ripple phase $P_{\beta'}$ have been observed close to the main transition. At even higher temperatures, the system is instable and phase separates. At low pressures, upon heating the fluid L_α phase tranforms into a phase with micellar aggregates, forming the sponge phase.

The largest difference between the phase diagrams is the apparent disappearance of the low pressure low temperature interdigitated gel phase $L_{\beta I}$. The reason for this might be twofold: First, only a few simulation runs have been performed in the p^*-T^*-region where the phase might be expected. Second, the simulations performed for the phase diagram typically were run only for up to 1,000,000 Monte-Carlo steps. The formation of the tilted gel phase from the artificially setup bilayer occurs within a few 10,000 MC steps, and the order-disorder-transition to the fluid L_α phase also is of the order of 100,000 MC steps. On the other hand, the reordering transition from the L_α-like transitional state to the ordered $L_{\beta I}$-state is much slower, and its set in may take up to a few million MC steps. Both facts together might mean that the phase simply has not been observed in the simulations.

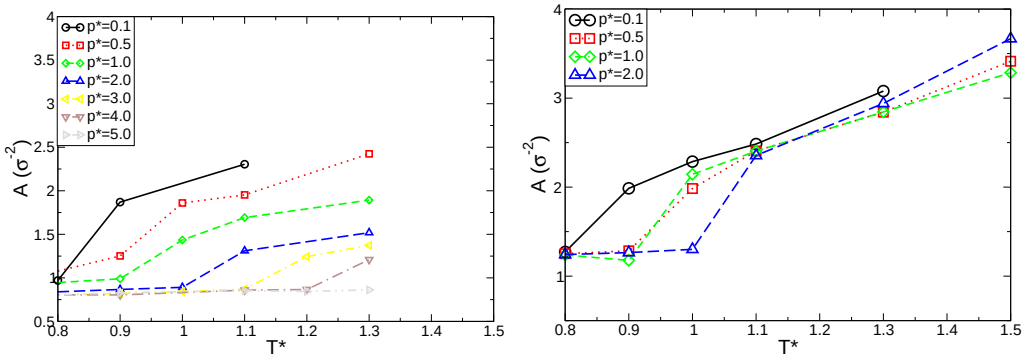
The temperature and pressure dependence of the area per lipid A (see figure 7.11(a)) is qualitatively comparable to the corresponding plots of the reference model. Note, that in this model the nematic order parameter S has been measured, instead of the chain order parameter S_z. Even though both observables are closely related, they yield different results in the tilted gel phase $L_{\beta'}$: while the nematic order parameter S is very close to 1 because all of the chains are very well aligned, the chain order parameter S_z would be lower, as the alignment of the chains with the bilayer normal is not so high. When this is kept in mind, the plot of the nematic order parameter S in figure 7.11(c) of this model compares well to the chain order parameter plot in the reference model.

The results of the model variant were published in [LS05].

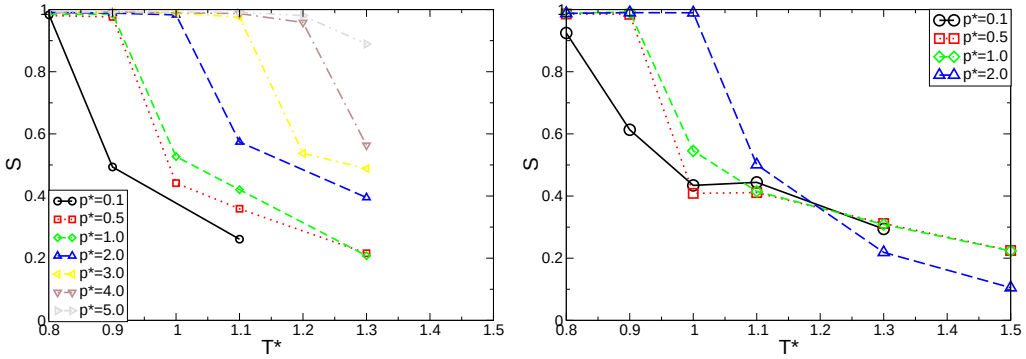


(a) phantom solvent model **(b)** surface potential model

Figure 7.10: Phase diagrams of the lipid model variant with longer bonds for two solvent environment models. Each data point represents one simulation run. The lines represent the estimated phase boundaries.



(a) Area per lipid A, phantom solvent model **(b)** Area per lipid A, surface potential model



(c) Nematic order parameter S, phantom solvent model **(d)** Nematic order parameter S, surface potential model

Figure 7.11: Equilibrium averages of different observables against the temperature T^* at different external pressures p^* in the lipid model variant with longer bonds for two solvent environment models.

7.3.2. Longer bonds, surface potential solvent environment

In a second variant of the model, the lipid model with the longer bonds was used, and furthermore, the solvent environment was replaced by the computationally very cheap surface potential environment model described in section 5.4.1 on page 58. The phase diagram (figure 7.10(b)) of the model and the plots of the observables (figures 7.11(b) and 7.11(d)) show fundamental differences to the reference model.

At high pressures, the fluid phase L_α is replaced by a *pancake phase*, where the position of the upper plane z_{upper} drops to 0 and the lipids lie on the planes and form a quasi-two-dimensional gas phase. This non-physical phase is caused by difficulties in the definition of the volume of the system required by the constant pressure algorithm, as described in section 5.4.1 on page 58.

Another non-physical phase occurs at high temperatures and very low pressures, in the domain labelled *destroyed*. In this phase, the bilayer breaks up into two dissociated monolayers and the lipids form an expanded, gaseous phase in the planes.

In general, the model seems to show less pressure dependence: the phase boundary of the main transition in the phase diagram is steeper, than in the phantom solvent model, and the observables have only a minor pressure dependence within the same phase. The temperature dependence of the observables is also comparable to the reference model.

The average area per lipid A in this model is higher than in the reference model, which stems from the fact that the heads are forced into the two-dimensional plane and can not to avoid each other in the third dimension.

The results of the model variant were published in [LS05].

7.4. Discussion

To check the validity of a coarse-grained model, it is crucial to compare the results of the simulations to experimental data. Therefore, in a first step, the conversion factor between standard SI units and the simulation units is estimated in section 7.4.1. In section 7.4.2 on the next page, the structure factor Q_{rz} of the model is compared to the experimental structure factor obtained from wide-angle X-ray grazing incidence scattering. Finally the model phase behaviour is compared to the phase behaviour of DPPC and DMPC in section 7.4.3 on page 90.

7.4.1. Conversion of simulation units to standard units

Even though the goal of this work is to model generic properties of lipid bilayers, it is sometimes instructive to be able to compare experimental and simulation results quantitatively. On this behalf, it is necessary to estimate a conversion factor between the simulation units σ and ϵ and standard SI units. Then, it is possible to convert the values of different observables to standard units. In coarse-grained simulations,

observable	simulation	exp.	conversion factor
bilayer thickness d ($L_{\beta'}$)	$7.75 \pm 0.1\,\sigma$	$42.8\,\text{Å}$	$1\sigma \approx 5.5 \pm 0.1\,\text{Å}$
bilayer thickness d (L_α)	$5 \pm 0.4\,\sigma$	39.2Å	$1\sigma \approx 6.5 \pm 0.5\,\text{Å}$
area per lipid A ($L_{\beta'}$)	$0.99 \pm 0.02\,\sigma^2$	$47.3\,\text{Å}^2$	$1\sigma \approx 6.9 \pm 0.1\,\text{Å}$
area per lipid A (L_α)	$1.4 \pm 0.1\,\sigma^2$	$62.9\,\text{Å}^2$	$1\sigma \approx 6.7 \pm 0.2\,\text{Å}$

Table 7.1: Conversion between the simulation length unit σ and standard SI units.

the best method to estimate the conversion factor is to compare the values of well-defined observables from the simulations with their experimental counterparts. As this work is mainly concerned with the structure of the bilayer, only the conversion factor for the length unit σ has been determined, and no attempt to estimate the energy conversion factor ϵ was made.

As the modelling of the system was performed with the phospholipid DPPC in mind, the simulation results were compared to experimental observables obtained for DPPC bilayers from X-ray-scattering experiments in the different phases, namely the bilayer thickness d and the area per lipid A in different phases. The experimental data for the fluid phase L_α was obtained from [STNSN96], for the tilted gel phase $L_{\beta'}$ from [NZTN$^+$96]. The simulation values were taken from figure 7.3 on page 75 for a pressure of $p^* = 2.0$ at different temperatures T. The results of this comparison are given in table 7.1. This comparison yields in a conversion factor estimation of $1\,\sigma \approx 6\,\text{Å}$.

Note, that this value does not match very well with the value of $\sigma = 3.79\,\text{Å}$ given by Stadler [Sta98], which is based on the chain modelling. In that model, a bead is thought to represent 2 CH_2 groups. Note, however, that Stadler used a bond-angle parameter of $\epsilon = 10$, while in this model, the value $\epsilon = 4.7$ taken from a later work [SSL99] was used, where the beads are assumed to represent 2-3 CH_2 groups, and the value of ϵ was adapted to match DPPC. When following these assumptions, a bead corresponds to 2.66 carboxyl groups, and the estimation based on the chain modelling would change to $\sigma \approx 5.1\,\text{Å}$, which better matches the conversion factor estimated from the observables.

7.4.2. WAXS structure factor

Figure 7.12 on the facing page compares the structure factors obtained from grazing incidence wide-angle X-ray scattering (WAXS) experiments in two different bilayer phases with the corresponding powder averages of the structure factor in the model system. The graphs compare well: in the fluid phase L_α, a smeared-out, weak "banana" shows the weak but nonetheless existant long-range structure in the phase, indicating a preferred lipid-lipid distance. In the more ordered ripple phase $P_{\beta'}$, the structure is much more pronounced and more concentrated, indicating a much more pronounced long-range structure in the phase.

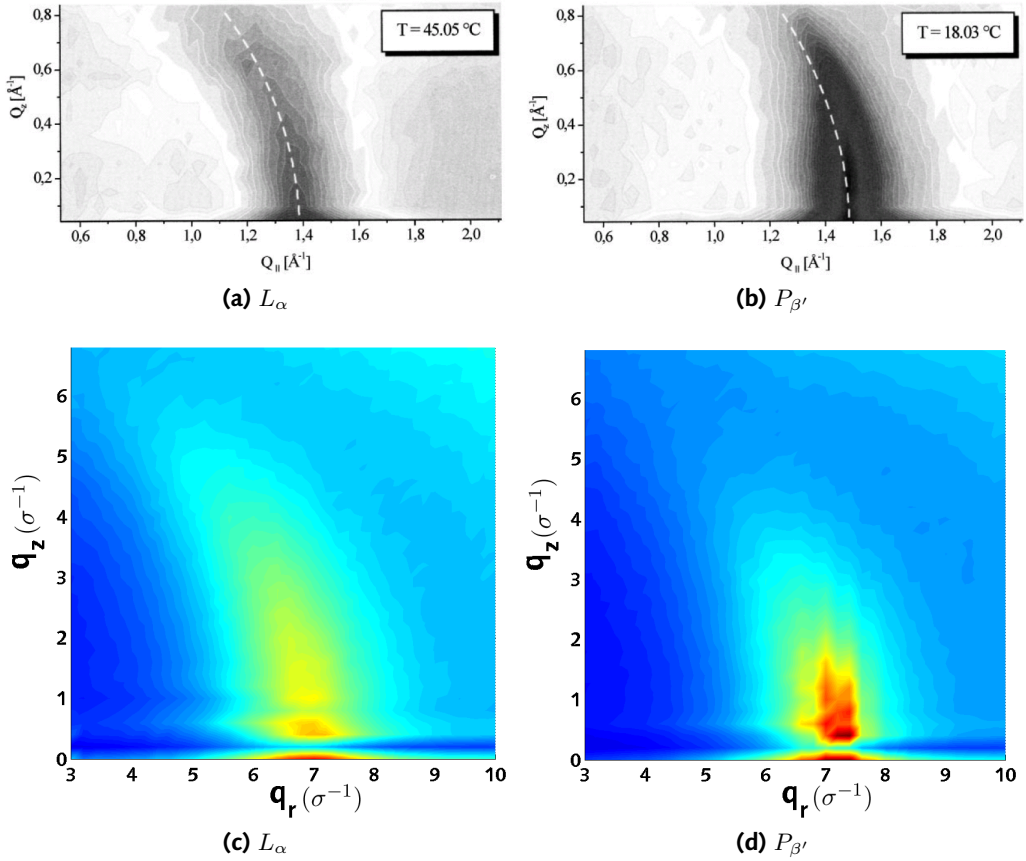
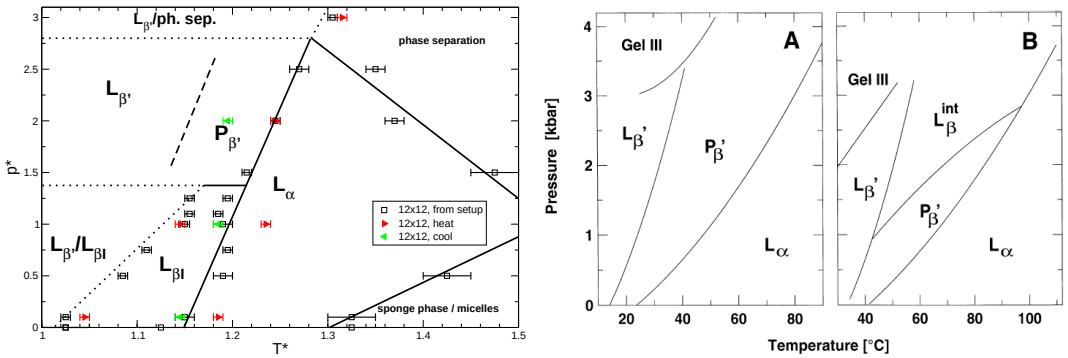


Figure 7.12: (a) and (b): WAXS structure factors from [Sal00] of DMPC bilayers in different phases; (c) and (d): Powder averages of the structure factor Q_{rz} of the model in different phases.

7.4.3. Comparison of the phase behaviour



(a) p^*-T^* phase diagram of the reference model.

(b) p-T phase diagram of DMPC bilayers (A) and DPPC bilayers (B) from [KC98].

Figure 7.13: Model and experimental phase diagrams.

Figure 7.13 shows the phase diagram of the reference model and experimentally determined phase diagrams of real DMPC and DPPC bilayers side-by-side. When comparing the diagrams, a few major differences meet the eye.

First, in experimental bilayers, a number of low-temperature subgel phases (for example *Gel III* or L_c, which is not shown in the phase diagrams) can be distinguished from the tilted gel phase $L_{\beta'}$, that do not occur in the model phase diagram. As explained in section 4.3.1 on page 46, the transitions between the different gel and subgel phases are mainly driven by the head-head interaction, *i.e.* the hydration of the lipid head region, the orientation of the head groups, and the acyl chain packing[KC98, 97]. In the lipid model, the head groups consist of a single, isotropic bead, that does not have any orientational degree of freedom and the size of the head group was chosen such that it roughly corresponds to the size of the hydrated head group. Furthermore, the two zigzagged acyl chains that make up the tail groups in real phospholipids are replaced by a single, straight bead-spring chain. Therefore, it is not surprising, that the low-temperature subgel phases can not be distinguished in the model system, and only the tilted gel phase $L_{\beta'}$ can be observed, that is characterised by fully hydrated head groups that lie on a hexatic lattice and whose long axis is aligned with the bilayer normal.

Real phospholipid bilayers at normal pressure undergo the well-known phase transition sequence $L_{\beta'} \rightarrow P_{\beta'} \rightarrow L_\alpha$. In the model, this phase behaviour is found at model pressures of $p^* \approx 2.0$, while it is different at higher respectively lower pressures. Most striking is the location of the interdigitated gel phase $L_{\beta I}$: in the model, it occurs at low pressures, while in the phase diagram of real phospholipid bilayers, it occurs at high pressures.

Note, that the tail packing and interaction between the tails in the interdigitated phase $L_{\beta I}$ is comparable to that of the gel phase $L_{\beta'}$. Consequently, the difference

between both phases has to be in the head group region.

Ultimately, all the differences can be traced back to the effect of the phantom solvent. As described in section 5.4.2 on page 58, the phantom solvent model has two effects on the lipid bilayers: on the one hand, it mediates the pressure onto the bilayer by means of an excluded volume interaction, that limits the amount of free volume in the system and therefore leads to a preference for compact configurations. On the other hand, it induces a pressure-dependent attractive depletion interaction between the lipid beads at the lipid–solvent interface, *i.e.* between the head beads. The range of this interaction is controlled by the size of the solvent beads: for distances below the phantom solvent bead size, the interaction is attractive.

This means, that at higher pressures in the model, two effects suppress the interdigitated phase: the attractive depletion interaction between the heads would prefer a lower average distance between the interfacial beads, and the general effect of the pressure punishes the additional free volume between the head beads. Therefore, the interdigitated phase can not occur at these pressures. At lower pressures, both these effects disappear.

In real phospholipids higher pressure more or less has the opposite effect. Note, that the size of the solvent water molecules is much smaller than the size of the lipid head group, and that the head groups are polar and can be hydrated, *i.e.* closely covered by water molecules. This means, that in the interdigitated phase, solvent molecules can enter in the spaces between the lipid head groups. Consequently, other than in the model system, there is no disadvantageous free volume between the head groups, and there is no attractive head-head interaction of the given range.

In the model system, very high pressures lead to phase separation between the lipids and the solvent beads. This is caused by the depletion interaction between the lipid beads that gets stronger at higher pressures. At these pressures, even non-bonded head beads demix[Sch06].

In general, it seems that in the model system between the gel phase $L_{\beta'}$ and the fluid phase L_α, at lower pressures the interdigitated phase $L_{\beta I}$ is found, while at higher pressures the ripple phase $P_{\beta'}$ occurs. Therefore, it can be assumed that the phases $L_{\beta I}$ and $P_{\beta'}$ are somehow related. In section 8.4 on page 111, a hypothesis will be formulated that offers an explanation for the phase diagram in detail.

7.5. Conclusions

In this chapter, the results of the bilayer model were discussed. It was shown, that the bilayer model readily self-assembles to form a stable bilayer, and that the bilayer has a rich phase diagram, which qualitatively resembles well the generic phase diagram of lipid bilayers. It was concluded, that the phase behaviour is significantly influenced by the head-tail mismatch and by the interaction between the head groups.

8. Microscopic structure of the ripple phases

In the phase diagram of phosphatidylcholine bilayer systems close to the main transition from the tilted gel phase $L_{\beta'}$ to the liquid phase L_α, the ripple phase $P_{\beta'}$ and $P_\beta^{(mst)}$ are observed. In these phases, the bilayer is not flat as in the other phases, but exhibits a height-varying ripple structure.

More than 30 years after the first observation of these peculiar phases [TLR73], many experiments have explored the characteristics of the rippled phases (see section 8.1.1) and a large number of theories (see section 8.1.2) have been developed. However, the microscopic structure of the phase has still not been understood. As Tristam-Nagle and Nagle state in 2004, "[...] we still do not know how the molecules are arranged in the rippled bilayers, whether they are tilted or some are melted. This is also an outstanding structural problem in lipid biophysics." [TNN04] Only very recently, rippled phases were observed in simulation studies in molecular detail (see section 8.1.3). These have shed some light on the microscopic structure of the *asymmetric* ripple phase $P_{\beta'}$.

In the previous chapter it was shown, that the phantom solvent bilayer model possesses a rippled phase close to the main transition. Section 8.2 will describe this phase in greater detail, and will identify it with the asymmetric ripple phase $P_{\beta'}$. Depending on the system size, the model phase shows a second occurence, that will be described in section 8.3 on page 107 and that will be identified with the symmetric ripple phase $P_\beta^{(mst)}$. To the authors knowledge, this is the first time that the microscopic structure of the symmetric ripple phase is explained. The structure of both ripple phase occurences is closely related. In section 8.4 on page 111, the structures of the model phases will be discussed and explained in detail, and it will be compared to experimental evidence, theory and simulational studies.

The results of this chapter have been published in [LS07].

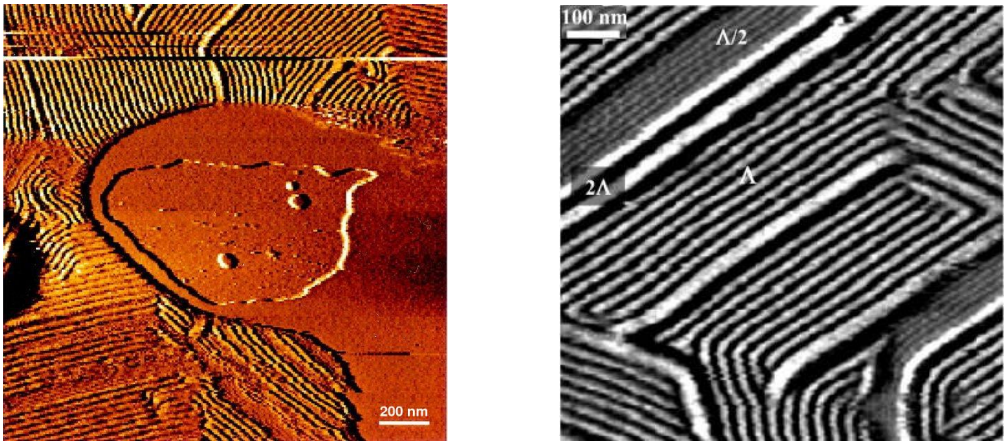
8.1. Research to date

8.1.1. Experimental evidence

The most apparent evidence of the ripple phase $P_{\beta'}$ can be observed in atomic force microscopy (AFM) images of lipid bilayers (see figure 8.1a on the next page). The same structure is also visible in tunnel electron microscopy (TEM) images (for example [WZ96]). In these images, ripples with a repeat distance of a few tens of nanometers

and a length of at least a few micrometers can be clearly recognised.

Depending on the thermal history of the sample (see section 8.1.1 on page 97), a second occurence of the ripple phase can be seen. The ripple phase $P_\beta^{(mst)}$ often occurs in coexistence with the ripple phase $P_{\beta'}$ and has roughly the double repeat distance, see figure 8.1b [YMTH91].



(a) (b)

Figure 8.1: AFM images of a DPPC double bilayer on mica support (a) A hole was scratched into the bilayers, so that the lower bilayer and the mica surface is visible in the middle of the image. Only the upper bilayer is rippled (from [LKC+02]). (b) Coexistence of the ripple phases $P_{\beta'}$ ("$\Lambda/2$"), $P_\beta^{(mst)}$ ("Λ") and a macroripple ("2Λ") (from [KLC+03]).

Ripple characteristics

To obtain more detailed information on the structure of the rippled bilayers, a great number X-ray scattering experiments (wide-angle X-ray scattering WAXS and small-angle X-ray scattering SAXS) have been carried out (*e.g.* [WW89]). The results of these experiments were used to construct electron density maps (EDMs) [STNSN96, SRK00, SRK03] of both ripple phases as shown in figure 8.2 on the next page.

From the electron density maps, an interesting characteristic of the ripple phases can be inferred: the phase $P_{\beta'}$ has an asymmetric height profile of varying bilayer thickness, the phase $P_\beta^{(mst)}$ has a symmetric height profile. Therefore, the both phases are usually denoted asymmetric and symmetric ripple phases, respectively.

While asymmetric ripples exhibit a repeat distance of about 13 – 15nm (DPPC at 39^0 C), roughly corresponding to about 20 lipid diameters, the symmetric ripple phase has about the double repeat distance of about 25 – 27 nm [KTNL+00]. Also the amplitude of the symmetric ripple is much larger than that of the asymmetric ripple.

Generally, the structure of the asymmetric ripple is described to consist of a "thicker" and a "thinner" arm with very different characteristics, connected by a kink.

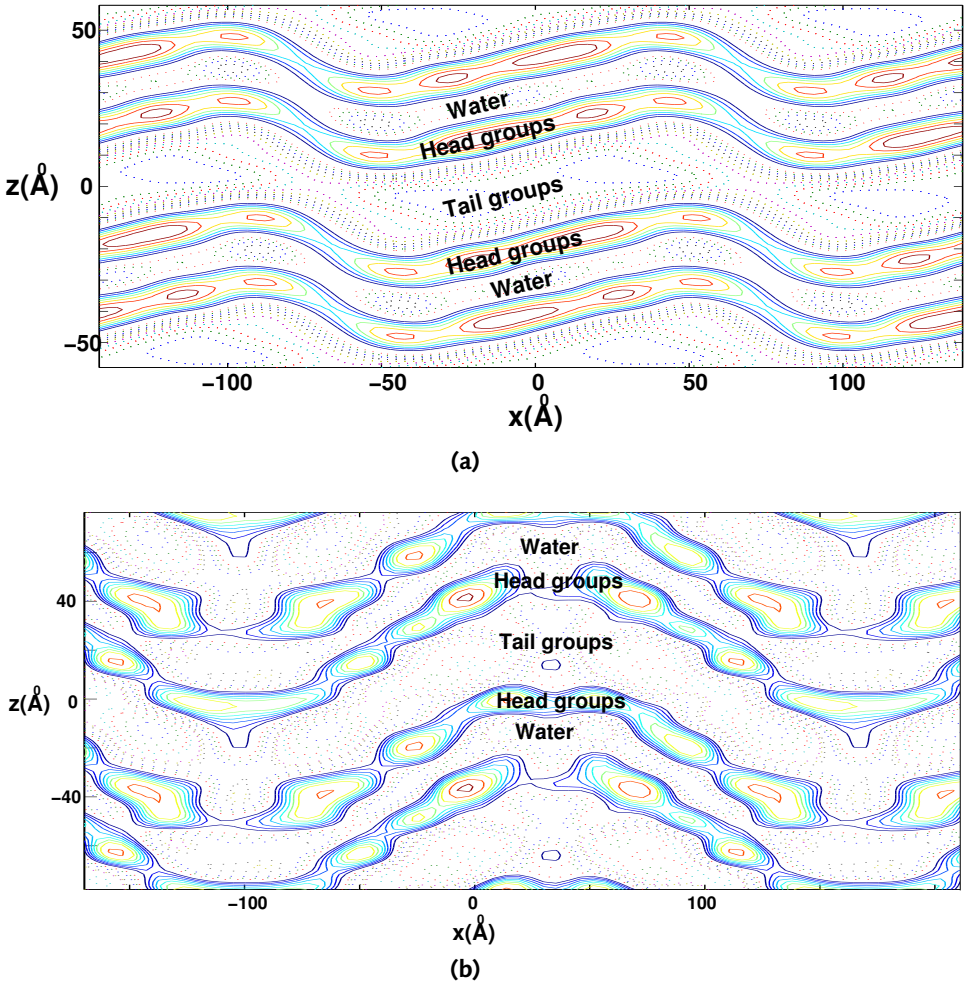


Figure 8.2: EDM of (a) the asymmetric ripple phase $P_{\beta'}$ of DMPC and (b) the symmetric ripple phase $P_{\beta}^{(mst)}$ of DPPC. Both EDMs have been published in [SRK03] and have been stretched in x direction, so that the x and z axes use the same scale. Note that (b) is described as being a bad fit to the data and not reliable.

Otherwise, the two types of ripples appear to be very similar: they exhibit the same type of peaks in X-ray scattering experiments [KTNL$^+$00] and they have almost the same volumetric requirements in scanning densiometry experiments [KKT96]. Together with the fact that both ripple types seem to be able to coexist, it is reasonable to assume that they are closely related.

Recently, a third type of ripple has been observed. In AFM images of double bilayers on solid support, "macroripples" with the double repeat distance of the symmetric ripple phase have been observed (see figure 8.1(a)). These seem to occur mainly as single macroripples in areas, where the symmetric ripple phase prevails [KLC$^+$03]. So far, only very few is known about these macroripples.

Chain ordering

There has been considerable debate going on over the question whether the lipids in the ripple phases are ordered like in the gel phase $L_{\beta'}$, or disordered like in the fluid phase L_α. Both hypothesis are supported by a number of experimental observances. This has led to the assumption, that the ripple phases might be a coexistence of gel-state and fluid-state lipids.

On the one hand, the x-ray scattering peaks of the intermolecular distance in the ripple phases are much stronger than the corresponding peaks in the fluid phase L_α. This clearly shows, that there is a high degree of order in the phase. However, the peaks are also not as pronounced as in the gel phase $L_{\beta'}$, which points towards a certain amount of disorder in the phase.

Furthermore, the different ripple domains typically join in discrete angles of 60^0 or 120^0 (see for example in figure 8.1 on page 94). This is typically interpreted as a hint to the underlying, hexagonal lattice of a well ordered phase.

On the other hand, a significant amount of disorder in the structure has been shown in NMR experiments [WSG81] and lipid self-diffusion experiments: the lipid self-diffusion is a few orders of magnitude higher than in the ordered gel phase. What is most interesting is the fact, that the self-diffusion of the lipids in the ripple phase is highly anisotropic [SCW83]. This can be interpreteted in favour of the hypothesis that the ripple phases are a fluid-gel state lipid coexistence and that the fluid phase lipids can easily diffuse along the ripples.

Further experimental support for this hypothesis is seen in time-resolved X-ray scattering experiments [RPR$^+$00] and in electron density maps of the asymmetric ripple phase $P_{\beta'}$ as seen in figure 8.2a. While the electron density of the head groups area in the "thicker" arm of the ripple matches the ordered gel phase L_β, the "thinner" arm has the characteristics of the fluid phase L_α.

As the thickness of the thick and thin arms also match the thickness of the gel respectively the fluid phase, this has led to the assumption, that the thin arm consists of fluid phase lipids while the thick arm consists of gel phase lipids [STNSN96]. Unfortunately, a number of experiments contradict this hypothesis: first, although the lipid self-diffusion in the ripple phase is much higher than in the ordered gel phase, it is also much lower than in the fluid phase [SCW83]. Finally, calorimetric studies of

the pretransition show, that at maximum about $\frac{1}{10}$ of the chains actually melt in the transition, while the thin arm of the bilayer contains about $\frac{1}{3}$ of the lipids [SRK03].

Single vs. multiple bilayers

There has been some discussion whether or not the rippled phase exists in single lipid bilayers, or if the rippled phase can only be observed in a stack of bilayers and consequently is caused by bilayer–bilayer interactions. The question is difficult to settle experimentally, as it is so far not possible to directly image a single bilayer in solution.

AFM images of double bilayers on solid support (see for example figure 8.1) seem to support the hypothesis that ripples can only occur in multiple bilayers, as the rippled structure is only observed in the double bilayer area and not in the regions where only a single bilayer prevails [FY96, LKC+02]. However, it has been argued that the solid support in this setup suppresses the ripple formation in the lower bilayer [MGE+99], so the question seems to be unsettled in this case.

On the other hand, the pretransition, although broader and less sharp, can also be found in single bilayers [LMDB84, Mey96]. Furthermore, it has been shown, that the ripple does not change its characteristics in bilayer stacks when the distance between the bilayers is increased [TUK+95]. This speaks against the hypothesis that the ripple phase is a result of the interaction between bilayers.

Thermal history dependence

At normal pressure, the ripple phases exist in a small stripe of a few degrees below the main transition. However, whether the symmetric or asymmetric ripples form delicately depends on the thermal history of the bilayer.

When the bilayer is prepared by heating up from the tilted gel phase $L_{\beta'}$, the asymmetric ripple phase $P_{\beta'}$ occurs. When the bilayer is prepared by cooling down the fluid phase L_α, the situation is much less clear. In general, a mixture of symmetric and asymmetric ripples seems to form, where the fraction of the ripple types depends on the rate of cooling [MYKH93]. The coexistence of the both ripple types can be directly visualised in AFM images (see for example figure 8.1(b)). When the sample is cooled sufficiently slow and remains at the transition temperature for a long time, a high content of symmetric ripples is achieved, while the fraction drops down to 0, when the rate of cooling is large[KTNL+00].

It is usually assumed that the symmetric ripple phase is metastable (hence the label $P_\beta^{(mst)}$) and very slowly transforms into the stable asymmetric ripple phase $P_{\beta'}$. This hypothesis is supported by observances from X-ray scattering experiments [KTNL+00] and AFM images[KLC+03]. However, the issue does not seem to be finally settled.

Lipid type dependence

Experimentally, the asymmetric ripple phase $P_{\beta'}$ occurs in bilayers of lipids that exhibit a tilted gel phase $L_{\beta'}$ (*e.g.* in lecithin bilayers), while it does not occur in bilayers of lipids that only have an untilted gel phase (*e.g.* in PE bilayers). Furthermore, the chain length of the lipid seems to play an important role: only PCs with intermediate chain lengths (12–22 CH^2 groups) exhibit the asymmetric ripple phase $P_{\beta'}$ [KC98].

The symmetric ripple phase $P_{\beta}^{(mst)}$ occurs in the same bilayers that exhibit the asymmetric ripple phase [KKT96]. The only exception to this rule is dimystriol phosphatidylcholine (DMPC), which appears not to have a symmetric ripple phase.

Typically, whether the gel phase is tilted or untilted is attributed to the size mismatch of the heads and tails: when the heads have a larger diameter than the tails, the gel phase needs to be tilted to compensate the size difference. Therefore it is often assumed that the ripple phases depend on the head-tail size mismatch.

8.1.2. Theories

Particularly during the first two decades since its discovery, a huge number of theories for the ripple phase have been developed. Here, only those theories will be focussed that can be applied to the model used in this work.

A large fraction of the theories are mean-field or Landau type theories that rely on elastic constants and other more macroscopic properties of the bilayer (see [Hei00] for a review). These theories can not be applied to the length scales that were studied in this work and will therefore not be considered. Also, theories that depend on interbilayer interactions, *e.g.* [CŽP81], are ignored in this work, as it has been shown experimentally, that the pretransition also occurs in single bilayers.

The more microscopic theories use different ingredients to obtain a rippled phase. Some focus on the chain melting[Hei00] or chain melting coupled to the spontaneous curvature of the bilayer[FSFM82, MFLM84], as chain melting does supposedly play a role in the pretransition.

Caused by the observation, that the ripple phases only occur in bilayers of lipids that have a tilted gel phase $L_{\beta'}$ as opposed to the untilted gel phase L_{β}, many other theories base on the packing competition between the head groups and the tails (*e.g.* [Lar77, PS82]) within a leaflet. The interaction between the lipids in the different leaflets of the bilayer is neglected in these theories. Furthermore, most of the theories actually predict symmetric ripple phases and either neglect the asymmetry of the ripple phase $P_{\beta'}$ or attribute it to asymmetries within the lipid molecule itself that the theory does not cover.

Of these theories, the theory of Carlson and Sethna [CS87] should be emphasised here, as their basic argument matches well the findings of this work. Therefore, the results of the theory will be presented in greater detail in section 8.4 on page 111.

8.1.3. Simulation studies

Not many simulation studies exist that show rippled phases in a model that has a representation of the conformational and translational degrees of freedom of the single lipid molecules.

In 2004, Kranenburg, Laforge and Smit have seen a rippled phase in a coarse-grained lipid bilayer model [KLS04]. The bead-spring model is closely related to the model used in this work. However, there are a few major differences. On the one hand, a single lipid in the model is more complicated: it has two tails and a head group that consists of three head beads. On the other hand, the model was simulated using the DPD method and employs the very soft bead interaction potentials connected with this algorithm. While all beads have the same radius, the interaction strength is varied. Using these soft potentials has a profound influence on the phase transitions in the model: while the transistions in the model used in this work are typically very sharp, they are much broader in the mentioned model.

The rippled phase seen in the model has a number of characteristics that match the experimental evidence: It occurs in the right temperature range at the phase transition between the gel phase L_β and the fluid phase L_α, and the repeat distance of the ripples is of the order of a few tens of lipids which fits to the evidence of the ripple phase $P_{\beta'}$. The phase is described as a coexistence of fluid-state and gel-state like lipids and exists at a temperature where half of the lipids are expected to be molten. An interesting fact is, that the lipids in the gel-state domain are tilted perpendicular to the direction of the ripple.

Usually, the interfacial area between the two phases in a phase coexistence is minimised. Apparently, this is not the case in the observed phase. The occurence of the phase is therefore explained by the competition between the head surface area and the average cross sectional area of the tail groups: in the gel-like domains, the tail groups can pack densly. However, this creates a frustration in the order of the head groups that can be compensated by the higher cross-sectional area of the tails in the disordered fluid-like domains.

However, the ripple phase in the model also has some characteristics that stand in contrast to the experimental evidence of the ripple phase. For example, the phase has a symmetric height profile, while the ripple phase $P_{\beta'}$ in experiment clearly has an asymmetric profile. The symmetric, metastable ripple phase $P_\beta^{(mst)}$, however, is much more emphasised than the ripples in the model phase [SRK03]. Furthermore, about half of the lipids in the model phase are in the fluid-like state, while calorimetric studies show that only about $\frac{1}{10}$ of the lipids melt in the ripple phases in experiment.

In 2005, de Vries, Yefimov, Mark and Marrink have observed a rippled phase in an atomistic simulation of a DPPC bilayer that was cooled down from the fluid phase L_α [VYMM05]. The structure that was observed in their work is similar to the structure described in this work, and the differences will be explained in detail in section 8.4.

8.2. Asymmetric ripple structures in the model

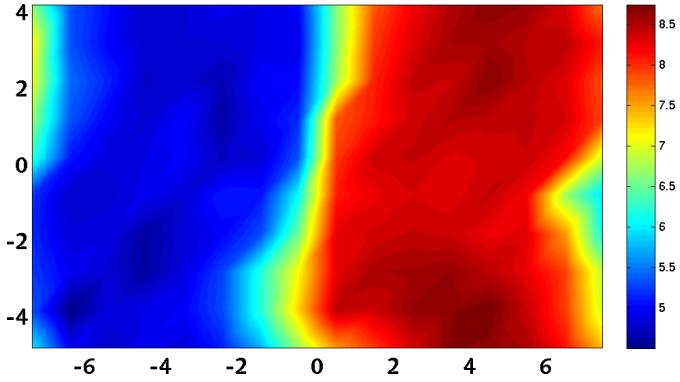
When a model bilayer in the fluid phase L_α at an intermediate pressure of about $p^* = 2.0$ is cooled down sufficiently, the curious structure pictured in figure 8.3 on the facing page or a closely related structure evolves. It is characterised by variations of the bilayer thickness (see subfigure 8.3a) that lead to a rippled appearance of the bilayer when it is periodically repeated. In the following, we will identify the structure as the origin of the asymmetric ripple structure $P_{\beta'}$. The direction of periodic repetition of the thickness variation will be referred to as the *ripple direction*.

The thickness variations originate from the coexistance of domains with lipids in very different states. While in the thicker "arm" of the structure (domain A), the lipids resemble the lipids in the gel phase L_β, the lipids in the thinner part (domain B) are highly interdigitated, as in the interdigitated phase $L_{\beta I}$. At the interface between these regions, one of the leaflets features a domain of disordered lipids comparable to the fluid phase L_α (domain C), while the other leaflet can maintain its gel-like state and has a kink that bends it around the disordered region. Note, that the gel-like domain A of the lower leaflet on the left side of the B-domain in the snapshot (red lipids) continuously goes over into the interdigitated domain B (red and blue lipids) and further on into the gel-like A-domain of the upper leaflet (blue lipids) to form a single, continuous domain of highly ordered lipids, where the orientation of the lipid end-to-end vectors swap from pointing upwards to pointing downwards, while the corresponding other leaflets end in the disordered C-domains.

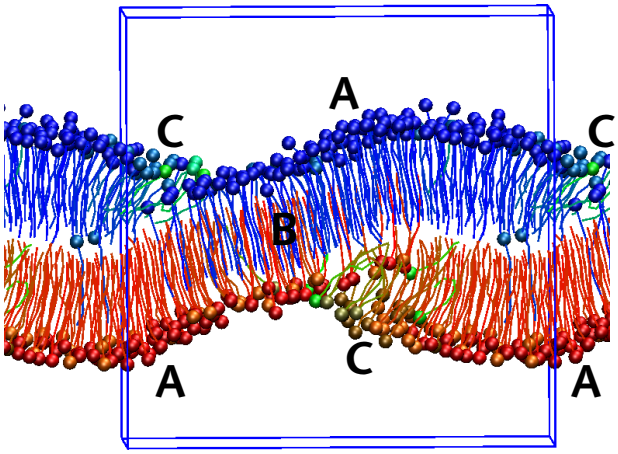
The rest of this section is devoted to describing the features of this structure in detail, that we will refer to as the *asymmetric ripple structure*, and that we will identify with the prevalent structure in the asymmetric ripple phase $P_{\beta'}$. Depending on the thermal history and the system size, a number of (meta-)stable variants of the structure occur, that differ in the exact arrangement and shape of the different domains, both in the domain structure of the leaflets and in the plane of the bilayer. These variants will be described in the following sections.

8.2.1. Bilayer thickness

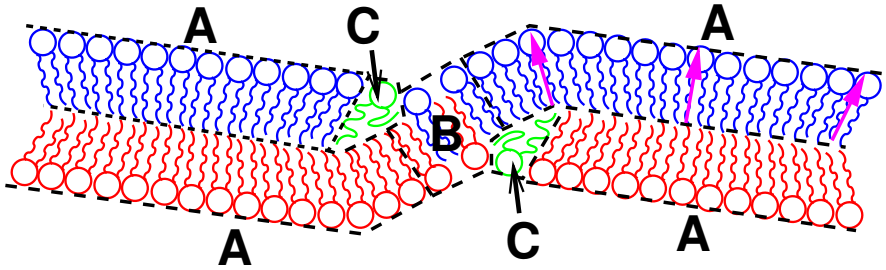
Plots of the bilayer thickness d give evidence of the rippled structure in the bilayer plane. Figure 8.4 on page 102 shows the thickness of different occurences of the ripple structure at different system sizes. Pronounced variations in the thickness that form an elongated ripple structure can be clearly recognised. The thicker domains have an average thickness which is slightly larger than the thickness in the gel phase L_β ($d = 7.6\sigma \pm 0.3\sigma$), while the thickness in the thin part goes down to a value close to that of the interdigitated gel phase $L_{\beta I}$ ($d = 5.25\sigma \pm 0.25\sigma$). In all cases, the area of the thinner domain is approximately similar to the area of the thicker part, with a small interfacial region.



(a) Local thickness of the bilayer. The horizontal and vertical axes lie in the plane of the bilayer.

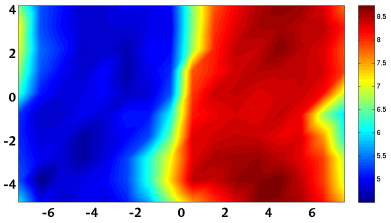


(b) Simulation snapshot of the structure. The horizontal axis lies in the bilayer plane, the vertical axis is parallel to the bilayer normal. To stress the structure, the particle coordinates are smoothed over 10 adjacent configurations. The colour of the lipid chains encodes l_z: red lipid chains point upwards, while blue chains point downwards and green lipids are in between. The head beads are depicted as spheres. For an explanation of the domain labels, see the text.

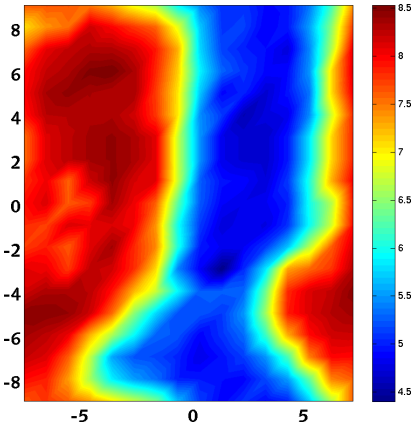


(c) Sketch of the structure. The purple arrows indicate the lipid tilt. For an explanation of the domain labels, see the text.

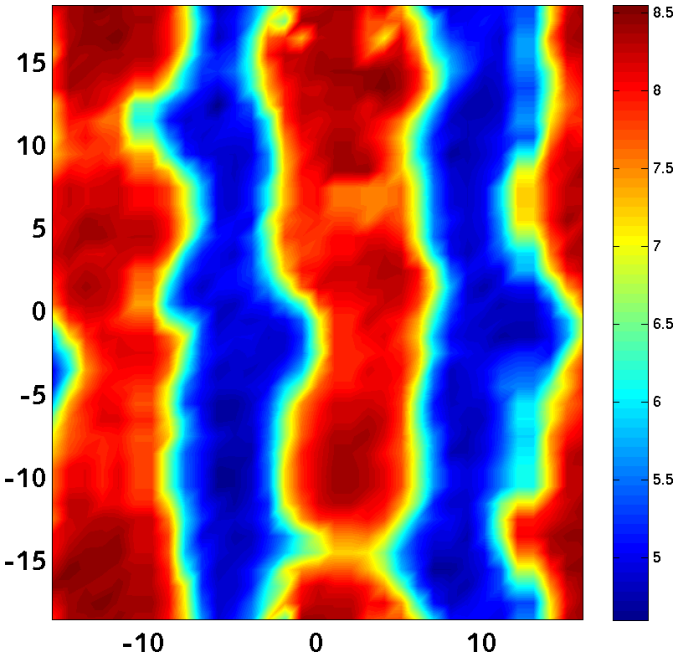
Figure 8.3: Structure obtained upon cooling down a 12x12-lipid-system from the L_α-phase.



(a) 12x12-lipid-system at $T = 1.1$, slowly cooled down from $T = 1.2$ (L_α)



(b) 15x15-lipid-system at $T = 1.19$, cooled down from $T = 1.2$ (L_α)



(c) 30x30-lipid-system at $T = 1.18$, resized from a rippled 15x15-lipid-system

Figure 8.4: Thickness in the x-y-plane for different occurences of the asymmetric ripple structure.

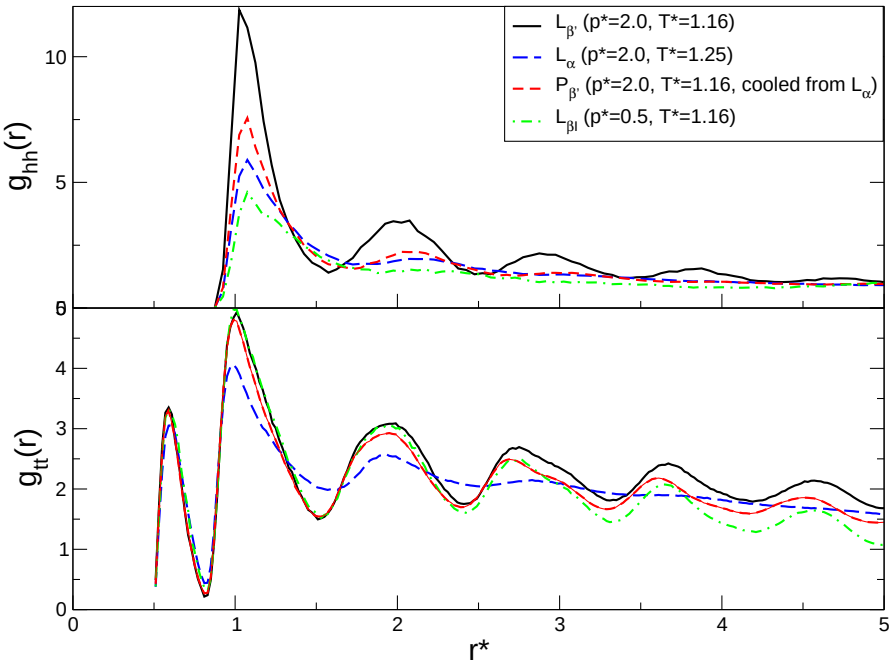


Figure 8.5: Head-head radial distribution functions $g_{\mathrm{hh}}(r)$ and tail-tail radial distribution functions $g_{\mathrm{tt}}(r)$ of the different bilayer phases.

8.2.2. Lipid ordering

In the snapshots of the structure in figure 8.8 on page 105, it can be seen that the major part of the lipid chains in the structure are well ordered and densly packed, while there are only comparatively small domains that consist of disordered, fluid-like lipids. This observation can be confirmed by the fact, that the radial distribution function of the tail beads $g_{\mathrm{tt}}(r)$ in the $P_{\beta'}$ structure in plot 8.5 compares best to the tail-tail radial distribution function of the gel phase $L_{\beta'}$ and the interdigitated phase $L_{\beta I}$, while the RDF of the fluid phase L_{α} shows much shorter-ranged correlations.

In contrast, the head-head radial distribution function $g_{\mathrm{hh}}(r)$ shows much weaker correlations than in the gel phase and is comparable to the RDF of the fluid phase L_{α}. This indicates relatively low ordering in the head region. The implications of this observation will be discussed in section 8.4 on page 111.

However, the ripple structure does not only show strong short-range correlations, but the lipids in the ripple even seem to lie on a long-ranged, hexagonal lattice. This can be quantified by the in-plane structure factor of the phase that is shown in figure 8.6. When comparing this to the structure factors of the other phases (see figure 7.9 on page 80), the hexagonal lattice structure occurs in the interdigitated phase $L_{\beta I}$ as well as in the tilted gel phase $L_{\beta'}$. However, both in the ripple structure factor and in the structure factor of the interdigitated phase, all of the peaks of the hexagonal structure are visible, while they are shifted out of the plane in the tilted gel phase. This indicates, that the average tilt of the lipids in the ripple phase is close to 0.

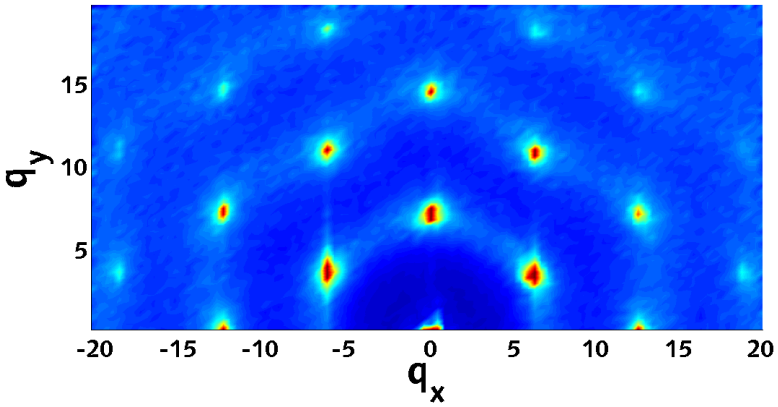


Figure 8.6: In-plane structure factor of the asymmetric ripple structure.

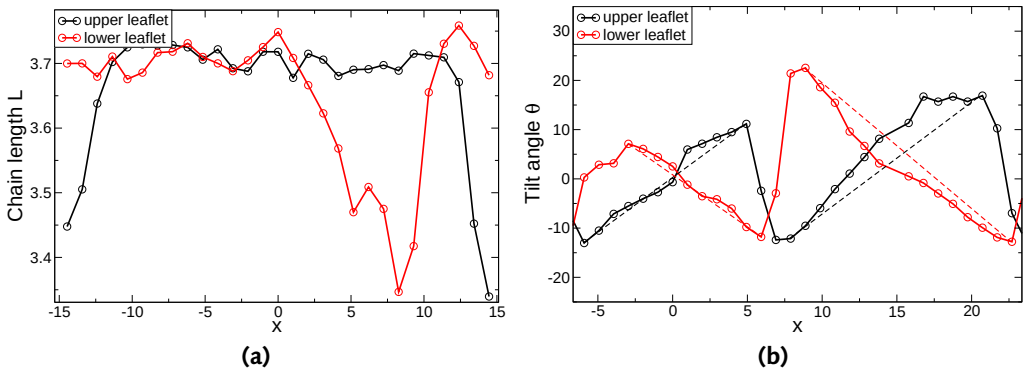


Figure 8.7: (a) Local average lipid chain length L and (b) local average tilt θ along the ripple direction in the asymmetric ripple structure.

The local variations of the lipid ordering can be observed in a plot of the average chain length along the x-axis (figure 8.7a). In the A and B domains of the rippled structure, the average chain length corresponds to the average chain length in the gel phase L_β, indicating stretched, well-ordered lipids, while in the thin domain it drops sharply to the much lower chain length of the fluid phase L_α.

8.2.3. Chain tilt and splay

As already observed above, the lipids in the rippled structure have on average no tilt towards the bilayer plane, in contrast to the tilted the gel phase. Instead, to compensate for the head-tail mismatch, the lipids exhibit *splay*, where the long axes of adjacent lipids are slightly tilted towards each other, so that the tilt towards the bilayer plane continuously changes, as is indicated by the purple arrows in figure 8.3c on page 101.

To quantise the splay, the local tilt of the lipids in both leaflets along the ripple

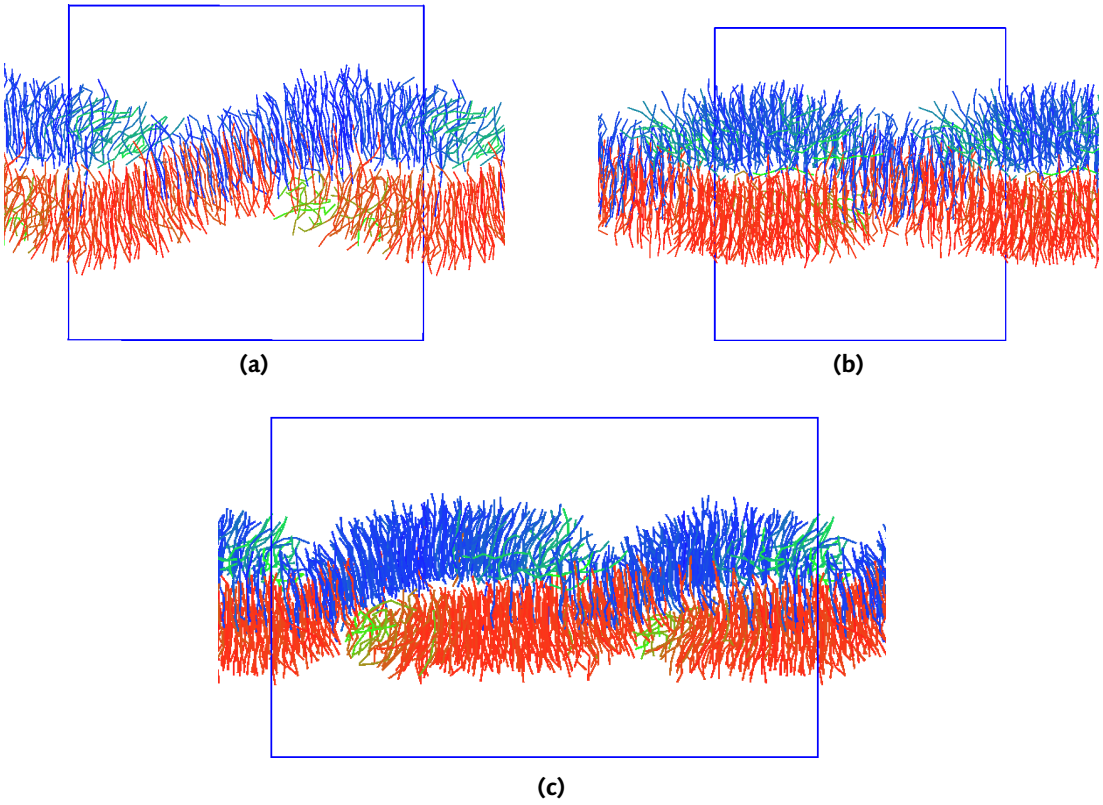


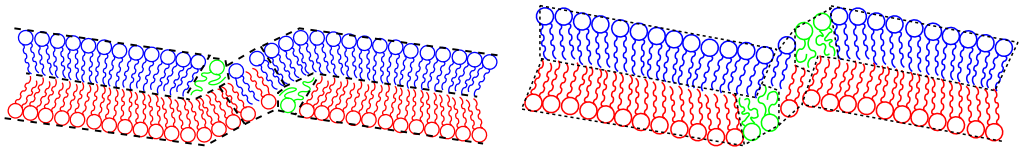
Figure 8.8: Snapshots of different occurences of the asymmetric ripple structure $P_{\beta'}$. The horizontal axis lies in the bilayer plane and is parallel to the ripple direction, the vertical axis is parallel to the bilayer normal. The colour of the lipid chains encodes l_z: red lipid chains point upwards, while blue chains point downwards and green lipids are in between. The systems correspond to the systems in figure 8.4 on page 102.

direction is plotted in figure 8.7(b). The plots show an asymmetric sawtooth shape, where the tilt changes continuously from negative to positive values or vice versa in the ordered domain, and rapidly returns to the original value at the disordered interface region. However, it can also be seen, that although the tilt graphs in both leaflets are discontinuous at the ripple, the tilts of the opposite leaflets match and could be smoothly continued. This indicates that both leaflets are interlocked and form an interdigitated domain.

Measuring the slope of the tilt graph in the ordered domains yields a value for the splay of $\frac{d\theta}{dx} = 2.3 \pm 0.2$ for all systems. Note, that perpendicular to the ripple direction, neither splay nor tilt can be observed.

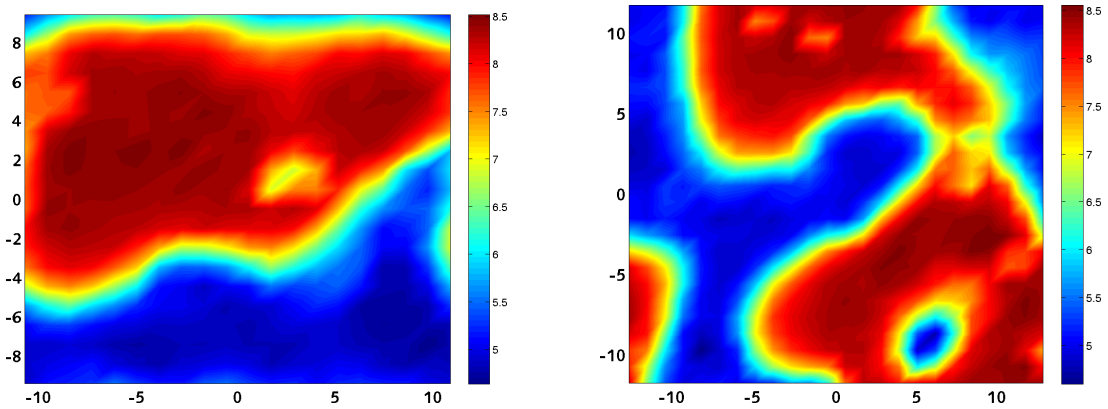
8.2.4. Domain structure and ripple repeat distance

A major problem when studying the ripple structures in the simulations are finite size effects. The largest possible systems that can be simulated in a feasible time only allow



(a) **(b)**

Figure 8.9: Sketches of different variants of the asymmetric ripple structure $P_{\beta'}$. Note, that the size of the interdigitated domain and the kink angle of the splayed domain seem to vary strongly.



(a) 20x20-lipid-system at $T = 1.16$ **(b)** 24x24-lipid-system at $T = 1.18$

Figure 8.10: Thickness in the x-y-plane for different systems with a box size that does not fit to the ripple repeat distance.

for a few ripples to fit in the simulation box. Furthermore, as the ripple structure is well ordered, the relaxation time of the structure and of the simulation box geometry in the plane of the bilayer is larger than the maximal possible simulation time.

As a consequence, it was impossible to accurately measure the equilibrium ripple repeat distance for the ripple structures from the simulations in the frame of this work. Also, the exact equilibrium structure of the different domains could not be determined, whether the interdigitated domain and the kink actually exist in equilibrium, or whether they are an expression of finite size effects.

However, it is possible to estimate limits for the repeat distance from the structures formed in systems of different sizes.

When cooling down a bilayer in the fluid phase L_α, single, asymmetric ripple structures were found in 12x12-lipid-systems and 15x15-lipid-systems, double ripples where found along the long axis of 30x12-lipid-systems.

In systems with intermediate sizes (20x20 and 24x24), no clear elongated ripples evolved. Instead, branched and interconnected structures of the thicker and thinner parts of the bilayer as shown in figure 8.10 on the facing page are formed. Still, the arrangement of the leaflet domains very much resembles the asymmetric ripple structure: the bilayer exhibits thicker, gel-like, well ordered domains as well as interfacial interdigitated domains were the upper and the lower leaflets of two gel-like domains form a joint interdigitated domain, with the corresponding opposite leaflets end in disordered, fluid-like domains.

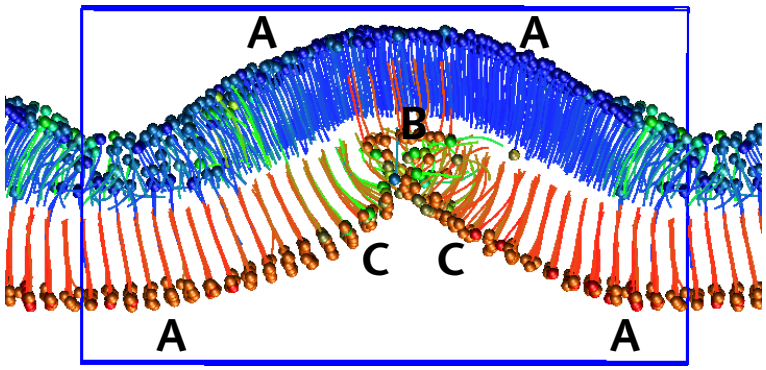
The different ripple structures formed in systems of different sizes differ mainly in the size of the interdigitated domains (labelled "B" in figure 8.3b on page 101) and the kink angle, as can be seen in the simulation snapshots of three different occurences of the ripple structure in figure 8.8 on page 105. While it is a relatively large domain in subfigure 8.8a, it is much smaller in the other subfigures, down to its complete disappearence in some cases. Figure 8.9 gives sketches of two alternatives of the asymmetric ripple structures, with either no kink and a small interdigitated region to an extended interdigitated region with a strong kink.

It can be assumed, that by varying between these two extremes, the system can compensate the mismatch between the simulation box geometry and the equilibrium ripple repeat distance to a certain amount. Unfortunately, caused by the slow relaxation of the box geometry, it was not possible to establish the equilibrium the domain in the frame of this work.

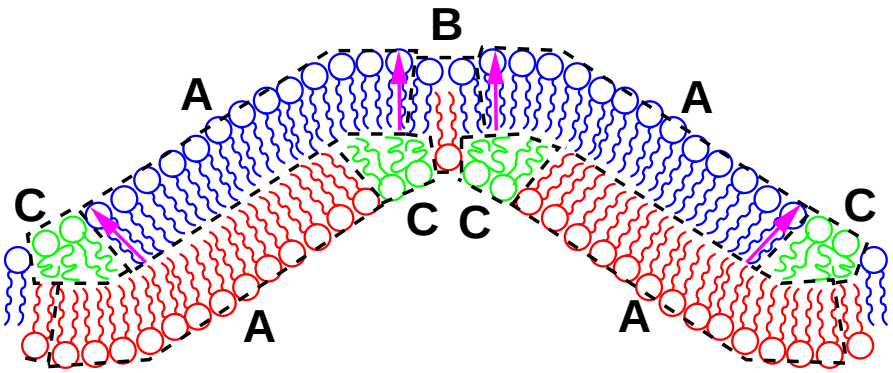
From these facts, the equilibrium ripple repeat distance can be estimated to be between 12 and 15 lipid diameters. From the behaviour of the systems, the author assumes that it is closer to 15 lipid diameters, and that the interdigitated domain is minimal, *i.e.* that there is direct contact between both leaflets at the ripple.

8.3. Symmetric ripple structures in the model

When cooling down a sufficiently large patch of a model bilayer at pressure $p^* = 2.0$ in the fluid phase L_α, in some cases, the structure depicted in figure 8.11 is formed. In

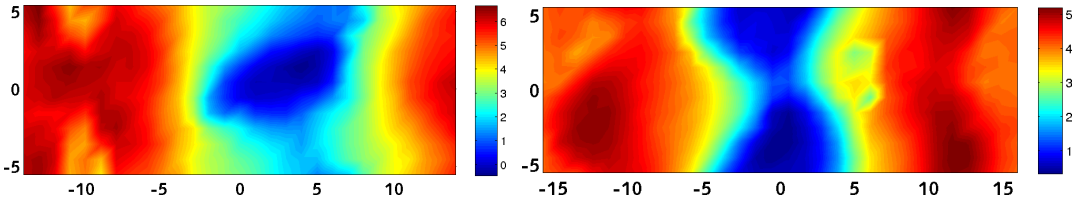


(a) Simulation snapshot of the structure. The horizontal axis lies in the bilayer plane and is parallel to the ripple direction, the vertical axis is parallel to the bilayer normal. To stress the structure, the particle coordinates are smoothed over 10 adjacent configurations. The colour of the lipid chains encodes l_z: red lipid chains point upwards, while blue chains point downwards and green lipids are in between. The head beads are depicted as spheres. For an explanation of the domain labels, see the text.



(b) Sketch of the structure. The purple arrows indicate the lipid tilt θ. For an explanation of the domain names, see the text.

Figure 8.11: Symmetric ripple structure obtained when slowly cooling down a 30x12-lipid-system from the L_α-phase.



(a) 12x30-lipid-system at $T = 1.18$, slowly cooled down from $T = 1.19$ (L_α)

(b) 12x30-lipid-system at $T = 1.16$, cooled down from $T = 1.2$ (L_α)

Figure 8.12: Height of the bilayer in the x-y-plane for different occurences of the symmetric ripple structure.

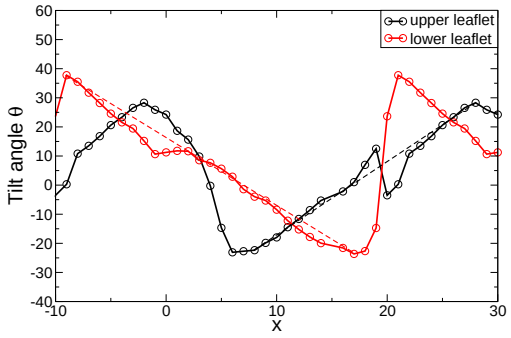


Figure 8.13: Average lipid tilt θ againts x-coordinate in the symmetric ripple structure.

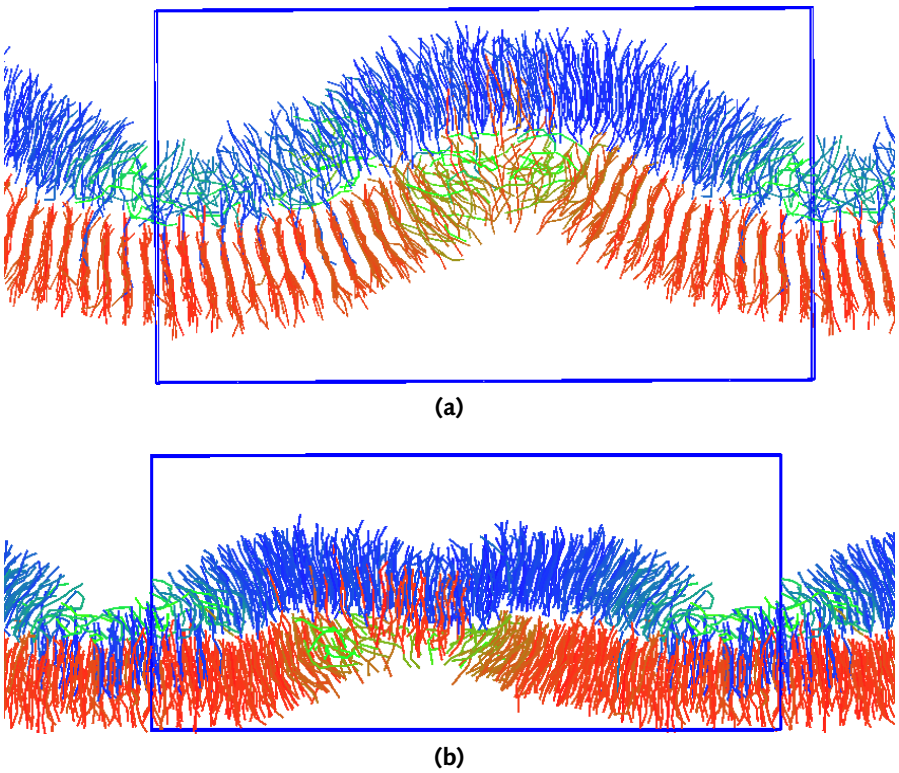


Figure 8.14: Snapshots of different occurences of the symmetric ripple structure. The vertical axis is parallel to the bilayer normal, the horizontal axis is parallel to the ripple direcion. The systems correspond to the systems in figure 8.12 on the facing page.

the following, this structure is identified with the prevalent structure in the metastable symmetric ripple phase $P_\beta^{(mst)}$.

The structure shares most of the characteristics of the asymmetric ripple structure described in the previous section. Firstly, the structure consists of large, well-ordered gel-like domains (labelled A), interdigitated domains of varying size (labelled B) and smaller, disordered, fluid-like domains (labelled C).

On average, the lipids in the structure are mostly ordered and reside on a hexagonal lattice – the radial distribution functions and structure factors of the structure are virtually indistinguishable from those of the asymmetric ripple structure and are therefore not shown. Instead of being tilted towards the bilayer normal on average, the lipids in the ordered domains exhibit *splay* that continuously changes the tilt (see figure 8.13 on the previous page). The value of the splay is $\frac{d\theta}{dx} = 2.3 \pm 0.2$, as in the case of the asymmetric ripple structure.

However, there are a number of significant differences between the structures. First of all, the arrangement of the domains is different from the asymmetric ripple structure described in the previous section, leading to a symmetric height profile which has roughly double the repeat distance of the asymmetric structure. This can be seen in figure 8.12 on page 108, where the height of the bilayer, *i.e.* the maximal z coordinate in the plane of the bilayer is visualised. Note that in the figure, the bilayer height h is plotted instead of the bilayer thickness d, as the thickness of the bilayer varies with the same repeat distance, as in the asymmetric phase, while the height has roughly the double repeat distance.

As in the asymmetric ripple structure, the lipids of two splayed domains on both sides of the ripple are interconnected via an interdigitated intermediate domain. In the symmetric ripple structure, the upper leaflet on one side interconnects with the upper leaflet on the other side. To allow for a continuous, splayed, ordered leaflet, the system has to locally tilt the plane of the bilayer, which results in the height variations of the symmetric ripple phase. In the cavity that is formed in the opposite leaflet, fluid-like disordered domains form that end the corresponding ordered domains, where the tilt of the leaflets is reset.

Again, only limits for the equilibrium repeat distance and size of the interdigitated domain can be given, as the relaxation times of the box geometry are very slow. The largest systems that were simulated only fit a single symmetric ripple structure. Different occurences of the structure are shown in figure 8.14, the repeat distances of which were between 28 and $40\,\sigma$. In section 8.4 on the next page, it will be discussed, why the repeat distance is about the double repeat distance of the asymmetric phase. In general, the amplitude of the ripple profile is larger than in the asymmetric ripple, and the bilayer has a distinct curvature.

Note, that a transition from the asymmetric ripple structure to the symmetric ripple structure or vice versa has never been observed. It can be assumed, that the energy barrier between both structures is too high for such an event to occur within feasible simulation times.

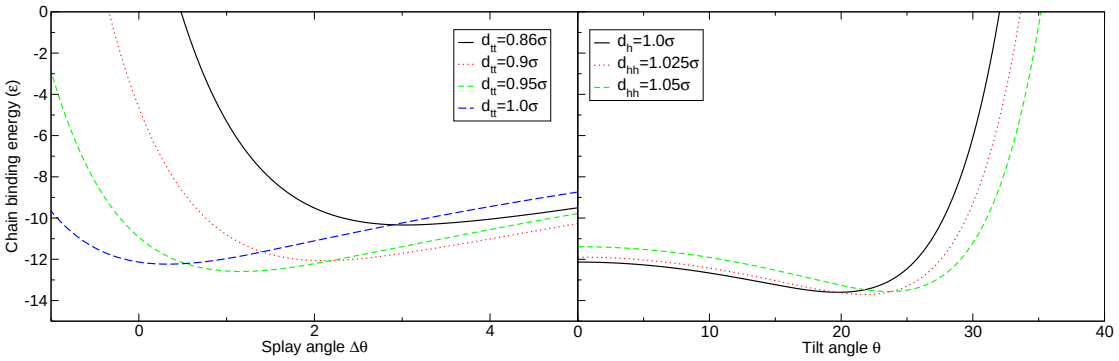


Figure 8.15: Comparison of the binding energies of two adjacent fully stretched lipids against the splay angle $\Delta\theta$ and tilt angle θ. d_{tt} denotes the distance between the last tail beads and d_{hh} between the head beads. Note, that the minimal binding energies are of comparable magnitude in both cases.

8.4. Discussion

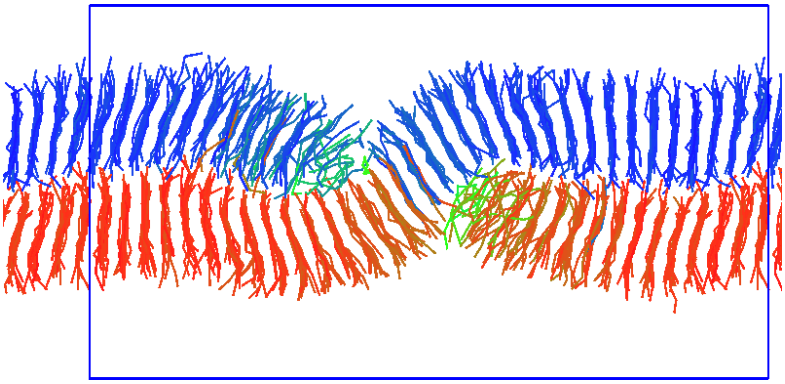
As the lipid model is relatively simple and has only a few free parameters, it can be expected that the observed or at least closely related structures actually occur in real lipid bilayers. Furthermore, it is reasonable to assume that they can be identified with the ripple phases $P_{\beta'}$ and $P_{\beta}^{(mst)}$, as was shown above.

Using computer simulations of the microscopic structure of the ripple phases, it is now possible to study problems related to the structure. Why do the ripple phases occur between the tilted gel phase $L_{\beta'}$ and the fluid L_{α} phase? Why are the phases stable?

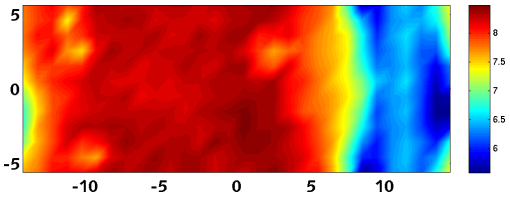
8.4.1. Splay instead of tilt

The studies seem to suggest, that the central element of both rippled structures in the model is the *splay* of the lipids in the ordered leaflet domains, in contrast to the constant tilt found in the tilted gel phase. Both tilt and splay fulfil the same basic function within the phase – they maximise the contacts between the tail groups to get into the energetic minimum of the Lennard-Jones potential, while on the other hand, both allow for the distance between the heads to be large enough to avoid steric clash. Like this, both counteract the frustration in the bilayer caused by the head-tail mismatch of the lipids. The binding energies between the lipids in both cases are of comparable magnitude, which is depicted in figure 8.15, which shows the interaction energy between two adjacent lipids for different values of the tilt and splay angles respectively.

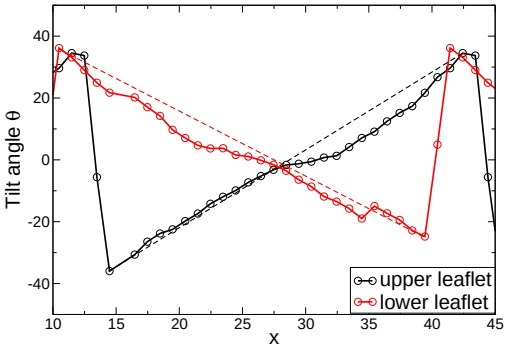
If it is assumed, that the high ordering in the ripple phases originates from domains with splayed lipids and not from tilted domains, this can also explain the existance of melted lipid chains in the ripple phases. In the structures described above, these disordered domains are required to end some of the splayed domains, and to reset



(a) Simulation snapshot. The colour of the lipid chains encodes l_z: red lipid chains point upwards, while blue chains point downwards and green lipids are in between.



(b) Local thickness in the bilayer plane.



(c) Local average tilt θ along the ripple direction in the asymmetric ripple structure. The measured splay corresponds to the slope of the straight lines.

Figure 8.16: 12x30-lipid-system at $T = 1.21$, heated up from $T = 1.2$ ($L_{\beta'}$).

the tilt of the lipids, so that it can form a periodically repeated structure that contains splayed leaflet domains. Otherwise, the existance of disordered lipids in a temperature range below the order-disorder transition of the lipid chains would be surprising, as they are energetically unfavourable in an overall energy-dominated phase.

Another fact that supports the importance of the splay for the ripples is the structure that was formed in the incomplete transformation to the ripple phase upon heating of a 12x30-lipid-system from the tilted gel phase $L_{\beta'}$ (see figure 8.16)[1]. Even though there is only a single asymmetric ripple structure, the ordered leaflet domains are

[1] At the beginning of the simulation, the local tilt of the lipids in the bilayer started to fluctuate very strongly, until after a few million Monte-Carlo steps a defect occured and a row of lipids in the upper leaflet "slid" over to the lower leaflet and formed a single asymmetric ripple as can be seen in the snapshot. Afterwards, the fluctuation of the local tilt were much weaker, so that the formation of a second ripple that would fit the simulation box is suppressed. Therefore, the structure can be assumed to be metastable.

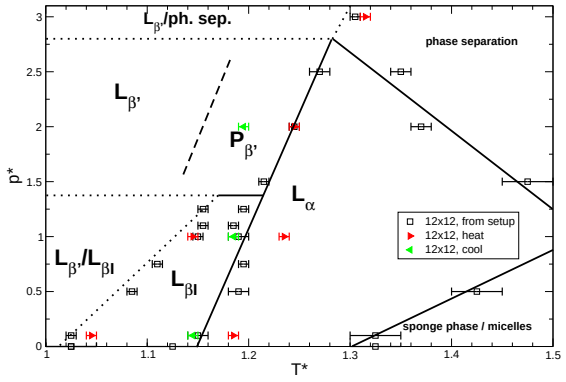


Figure 8.17: Phase diagram of the bilayer reference model. The lines are the estimated phase boundaries.

splayed, and the whole structure has arranged such, that the splay within the ordered leaflet domains is $\frac{d\theta}{dx} = 2.3 \pm 0.2$, as it was in all of the observed ripple structures. That the value of $\frac{d\theta}{dx}$ is constant in all systems suggests, that this particular splay value is a basic property of the system and not just an artifact caused by finite size effects.

8.4.2. Explaining the phase diagram

When tilting the bilayer lipids can reduce the head-tail mismatch in a periodically repeating system without the need for disordered regions, why does splay occur in the first place? In the following, a hypothesis will be presented, that explains the phase diagram of the reference model including the ripple phase in terms of the preferred phases of the different lipid components: each of the bilayer phases from the phase diagram in figure 8.17 can be interpreted as a combination of the state of the tail groups and the state of the head groups. As the head and tail groups are bound to each other, they are of course heavily interrelated, and the system has to arrange such that a stable phase forms. The hypothesis is supported by the plot of radial distribution functions of the head and tail groups in figure 8.5 on page 103, respectively.

At low temperatures, both the head and tail groups prefer to form a solid phase with high packing, such as the tilted and untilted gel phases $L_{\beta'}$ and L_β. In this phase, both components show strong short- and long-ranged correlations (see chapter 7 on page 73). At high temperatures above the main transition, both head and tail groups unbind and form the fluid phase L_α of the bilayer.

As has been discussed in section 5.4.2 on page 58, the pressure p^* induces an attractive depletion interaction between the head groups, which grows with increasing pressure. At low to intermediate pressures, this attraction is relatively weak, and in particular it is weaker than the tail-tail attraction, so that the order-disorder transition of the head beads occurs at a lower temperature, than the order-disorder transition of the tails. Consequently, at temperatures between the head group transition and the tail group transition, the system has to arrange with the situation, that the head

groups would prefer to unbind, while the tail groups still want to form a densly packed solid.

At low pressures, there is almost no attractive interaction between the head groups, so that at the order-disorder transition of the heads, the heads unbind completely and would prefer a gaseous state. This combination is represented by the interdigitated phase $L_{\beta I}$, where the head groups show only very weak correlations, while the tail groups maintain a solid, highly correlated state.

At intermediate pressures, the attractive depletion interaction between the heads gets stronger, caused by the increasing number of phantom solvent beads. When increasing the temperature, the head groups therefore do not completely dissociate as in the interdigitated phase above, but instead prefer to form a weakly correlated liquid. The splayed domains in the ripple phases $P_{\beta'}$ and $P_{\beta}^{(mst)}$ carry all the characteristics of this combination: while the tail groups are still highly correlated and as densly packed as in the gel phase, the head groups are weakly correlated, but much less than in the gel phases.

At high pressures, the induced head-head attraction becomes so strong, that even heads that are not bound to a lipid would demix[Sch06]. Therefore, the phantom solvent and the lipids phase separate, and no bilayer is maintained.

8.4.3. Ripple structures

Under the assumption, that the splay is the main property of the ripple phase, the curious asymmetric and symmetric ripple structures can be explained.

For a monolayer, the splay of the lipids is weakly coupled to the curvature of the monolayer [LM93]: a convex curvature would be favourable for the formation of splay, while a concave curvature would suppress it. However, as a bilayer consist of two coupled monolayers (the leaflets) with opposite curvature, it will be locally flat on average.

In a flat bilayer, the allowed tilt of a lipid in each of the leaflets has an upper bound: when the tilt grows greater than a certain value, the interaction energy with the adjacent lipids grows rapidly, thus making larger tilts unfavourable. Therefore, the size of a bilayer segment that contains ordered, splayed lipids in both leaflets has an upper bound.

On the other hand it was shown above, that the system is still below the order–disorder transition of the lipid tails, which makes the the splayed regions with their well-ordered tails energetically favourable. Consequently, one can assume that the system will try to make the splayed domain as large as possible. This results in a characteristic size of the splayed domains for a given lipid type, which should only weakly depend on the temperature.

To allow for a periodically repeated structure that includes flat domains with splayed lipids in both leaflets, the system somehow has to reset the tilt at both ends of these domains. The most obvious solution to this problem would be domains of disordered lipids between the splayed domains, which would result in the structure depicted in

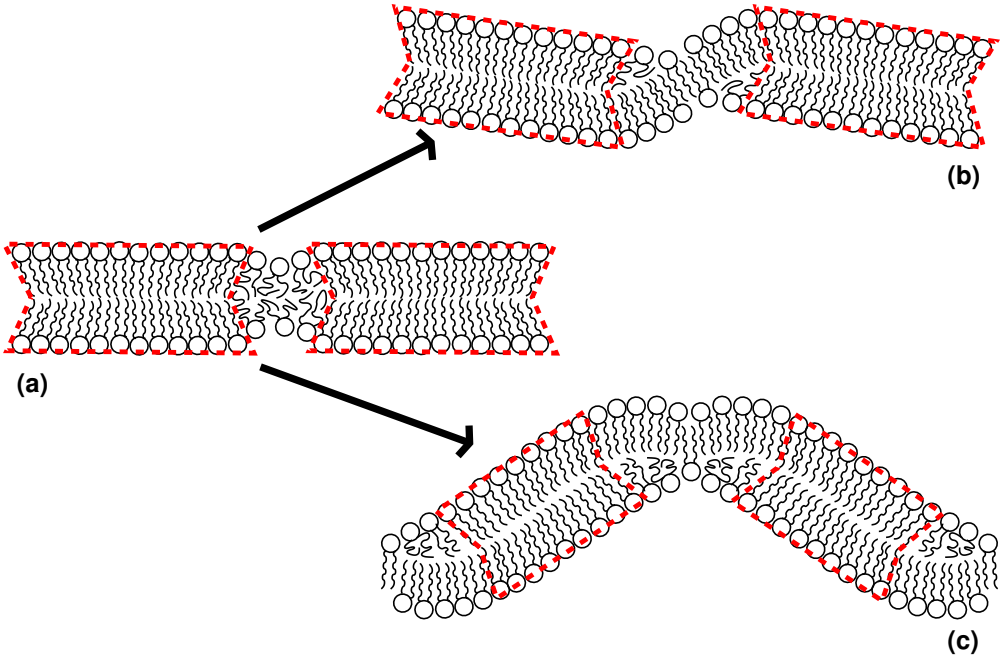


Figure 8.18: Sketch of possible ripple structures: (a) Suggested ripple structure from [CS87] (b) Asymmetric and (c) symmetric ripple structures as obtained in this model. The splayed domains are visually highlighted.

figure 8.18(a). The structure has been proposed earlier [CS87] as the origin of the ripple phases. However, it has a few shortcomings, as it does not explain why there are two ripple phases $P_{\beta'}$ and $P_{\beta}^{(mst)}$, nor does it satisfactorily explain the asymmetry of the asymmetric ripple phase phase.

The structures found in this work provide alternative solutions to the problem of periodically repeating structures that contain domains of splayed lipids. Instead of forming domains of disordered lipids, two of the leaflets interconnect via an interdigitated domain such, that a single, well-ordered and splayed monolayer is formed. Only the two remaining leaflets need to be ended by disordered lipids.

In the asymmetric ripple structure (see figure 8.18(b)), the opposite leaflets of both splayed bilayer domains form a single, continuous, well-ordered monolayer, where the lipid orientation swaps from pointing upwards to pointing downwards or vice versa at the interfacial interdigitated domain. What is interesting about this structure is, that it unveils the origins of the asymmetry in the height profile of the phase, and that it does not require any kind of asymmetry in the lipid molecules itself. Also note, that the local plane of the splayed bilayer domains is inclined towards the bilayer plane.

In the case of the symmetric ripple structure (see figure 8.18(c)), the splayed domains are inclined towards each other, such that the same leaflets of both bilayer domains form an single, continuous, well-ordered monolayer. This results in a convex bilayer segment, with the required disordered lipids in the cavity. To make the whole

structure continuous, the splayed, extended leaflet domains alternate in each ripple.

If it is assumed, that the interfacial interdigitated domains between the splayed domains are significantly smaller than the splayed domains itself, the construction of both structures provides a good explanation of the fact that the repeat distance of the symmetric ripple phase is roughly the double repeat distance of the asymmetric ripple phase.

In summary, the presented ripples seem to be good solutions to the problem of forming periodic structures that contain large, well-ordered domains of splayed lipids on the one hand, while on the other hand the number of required, energetically unfavourable disordered lipids is small.

8.4.4. Comparison to other studies

Under the assumption, that the central element in the formation of the ripple phases $P_{\beta'}$ and $P_{\beta}^{(mst)}$ is the splay of the lipids in the ordered leaflet domains in contrast to the constant tilt in the tilted gel phase $L_{\beta'}$, the ripple structures can be compared to experimental findings and to those of other simulational studies.

Experiment

In the following, the characteristics of the ripple structures observed in the model will be compared to the experimentally determined properties of the ripple phases $P_{\beta'}$ and $P_{\beta}^{(mst)}$. Together, they will provide evidence that the structures are indeed identical to the prevalent structure of the ripple phases, and that they can decribes the long-sought-after microscopic structures of the phases.

In general, the splay hypothesis matches the observation, that the ripple phase occurs only in bilayers of those phospholipids that form a tilted gel phase $L_{\beta'}$ in contrast to an untilted gel phase L_{β}. The formation of the tilted phase is usually attributed to the head-tail mismatch, which is also a requirement for the formation of splay in our model.

In the phase diagram in figure 7.2 on page 74, the structures in the model occur close to the transition between the tilted gel phase $L_{\beta'}$ and the fluid phase L_{α} in a small temperature and pressure range. This corresponds well to the experimental phase diagram, where the ripple phases are found in a narrow temperature range close to the main transition.

The simulation snapshots of the structures and the in-plane structure factor in particular show, that the structures contain overall well-ordered lipid chains. In experiment, this can explain the clear peaks obtained in x-ray scattering experiments. Furthermore, underlying hexagonal lattices have been found in the model structures, which fits well to the experimental observation, that ripple domains typically join in discrete angles of 60^0 or 120^0.

However, a small fraction ($\approx 10\%$) of the lipid chains in the model are melted, which

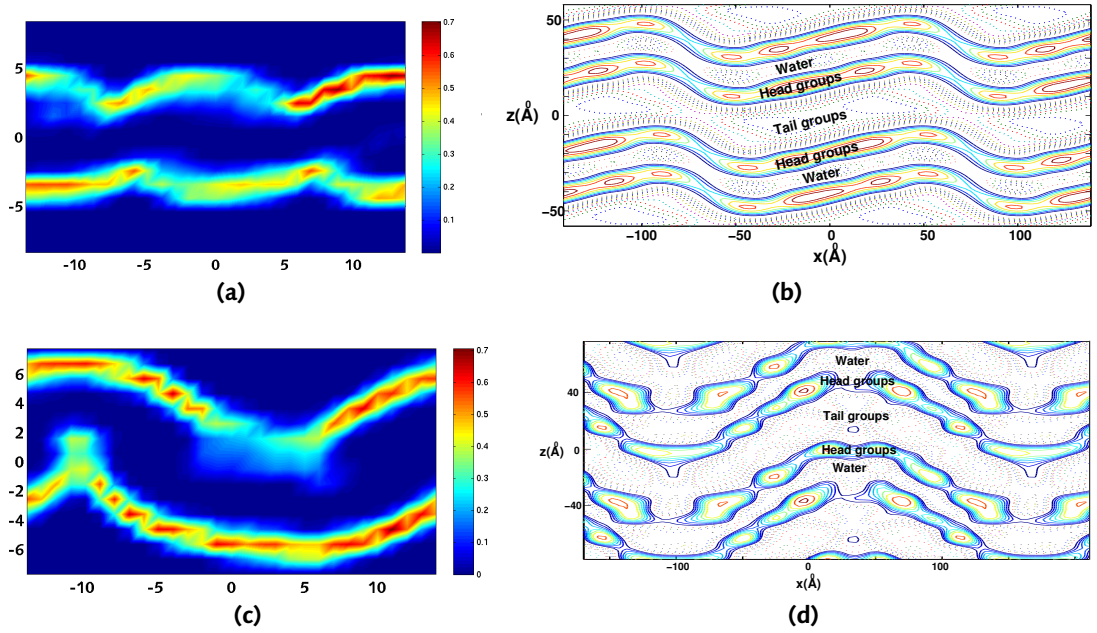


Figure 8.19: Comparison of the local densities of the head groups in x-z plane in the asymmetric (subfigure (a)) and symmetric ripple structures (subfigure (c)) of the model, and the EDMs of the asymmetric ripple phase $P_{\beta'}$ of DMPC (subfigure (b)) and the symmetric ripple phase $P_{\beta}^{(mst)}$ of DPPC (subfigure (d)) from [SRK03].

is consistent with the experimental evidence, that the peaks in x-ray scattering experiments are relatively weak, and also with the caliometric studies of the pretransition.

The highly anisotropic lipid self-diffusion found in experiments can be observed in the model. Even though is has not been explicitely measured, the anisotropy is immediately apparent when watching a simulation, and it can easily be explained, as only the lipids in the disordered domain are mobile, while the other lipids are in a confined, gel-like state.

In figure 8.19, the head group densities of the ripple structures in the model are compared to the EDMs of the ripple phases $P_{\beta'}$ and $P_{\beta}^{(mst)}$ of DMPC. The comparison points out a number of similarities. The asymmetric respectively symmetric height profiles can clearly be recognised. The asymmetric structures have a thicker and a thinner arm, while the symmetric structures have a mostly constant thickness. In the case of the asymmetric structures, the head group density of the thicker arm is slightly larger than that of the thinner arm.

Furthermore, it can be observed, that the symmetric ripple structure roughly has the double repeat distance of the asymmetric structure. The splay hypothesis provides a good explanation of this fact, as both repeat distances are gouverned by the characteristic size of the splayed bilayer domains.

Figure 8.20 on the following page shows sketches of the molecular structures that

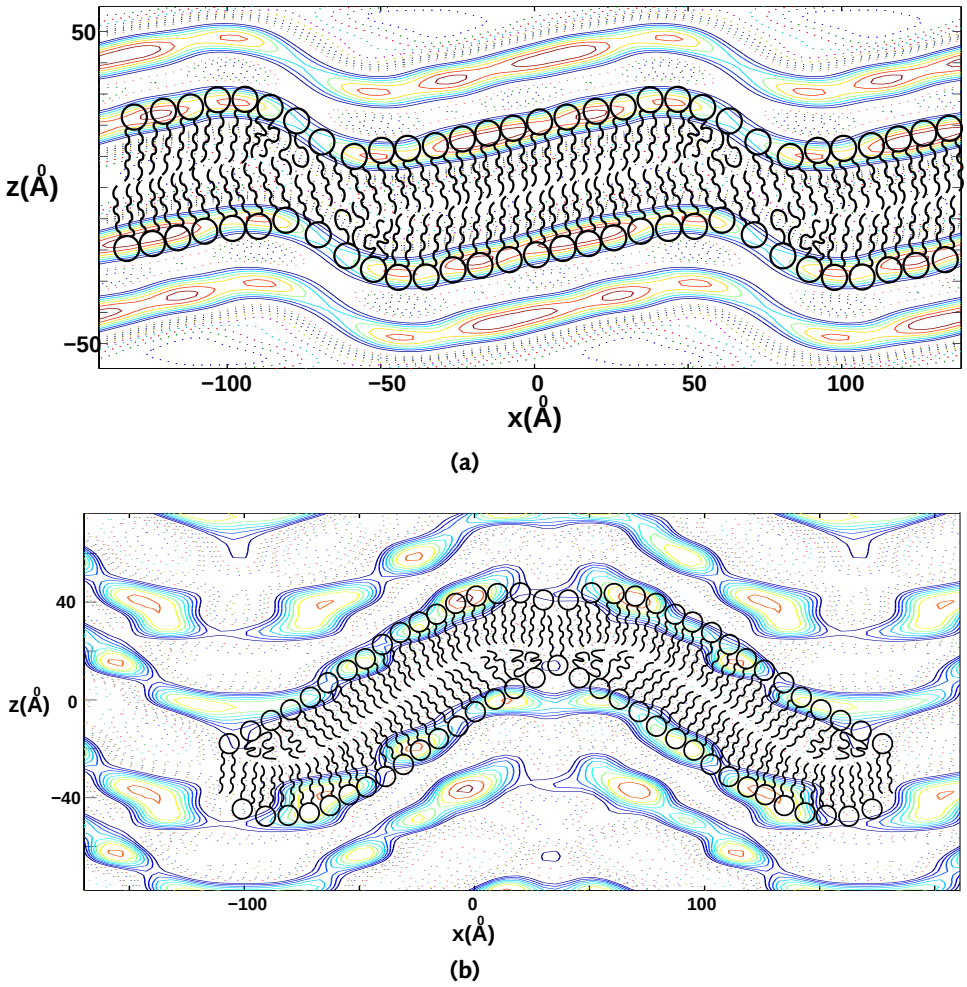


Figure 8.20: Sketch of the proposed microscopic structures of the asymmetric ripple phase $P_{\beta'}$ (subfigure (a)) and the symmetric ripple phase $P_{\beta}^{(mst)}$ (subfigure (b)), superimposed onto the EDMs of DMPC resp. DPPC from [SRK03].

are proposed for the asymmetric ripple phase $P_{\beta'}$ of DMPC and for the symmetric ripple phase $P_{\beta}^{(mst)}$ of DPPC, superimposed onto the EDMs of the phases. Note that for the asymmetric structure to fit, it must be assumed that the interdigitated domain is minimal and that there is no kink at the ripple. Also note, that the fit of the symmetric structure is not as good as in the asymmetric case, but that the EDM in the symmetric case is described as not very reliable by the authors [SRK03].

Another important aspect of the ripple phases is, that it has been discussed for a long time, whether both ripple phases $P_{\beta'}$ and $P_{\beta}^{(mst)}$ are stable phases, or whether one of the phases is metastable and only very slowly transforms into the other one. Meanwhile, it is generally accepted that the symmetric ripple phase $P_{\beta}^{(mst)}$ is a metastable phase (hence the label *(mst)*). Given the microscopic structures found in this work, it is clear, that the transformation from one of the structures into the other has a high energy barrier and requires a highly collective transition of a generally well-ordered system that is energetically gouverned.

Simulational studies

Mostly due to the relatively large size of the ripple structures, only two simulational studies of the ripple phases that include atomistic or molecular details have been performed so far. Even coarser-grained simulation studies are not considered in this work, as they do not contain the degrees of freedom required to model the structures observed in this work.

Kranenburg and Smit [KVS03a, KS04] studied a DPD-based coarse-grained model that contains some more details than the model used in this work. Notably, the lipids have two tails of 5 beads each, and a head consising out of two beads. The phase diagram of their model reproduces mainly has two phases: an untilted, gel-like phase, which is identified with the gel phase L_{β}, and a fluid phase, identified with the fluid phase L_{α}. Close to the main transition, a rippled phase is found. However, the structure of the ripple fundamentally differs from the structures found in this work. It has a symmetric height profile and no splay of the ordered domains has been observed. Instead, the lipids are tilted *perpendicular* to the ripple direction. Interestingly, although the level of detail of the model and the parameters are comparable to the model used in this work, the ripple structures found in this work have not been observed.

As has been shown before, an important ingredient to the ripple phase found in this work is the head-tail mismatch, which causes the tilt in the tilted gel phase $L_{\beta'}$ and the splay in the rippled structures. As the lipids in the model of Kranenburg and Smit exhibit two tail chains and heads of the same size, there does not seem to be a significant head-tail mismatch. This is supported by the fact, that the model only possesses an untilted gel phase. Therefore, it is not surprising that the ripple structure found in this work was not seen in their model.

Curiously, however, the model does exhibit the tilted gel phase and also a structure comparable to the ripple structure found in this work when model alcohols (*i.e.* very short lipid chains) are added to the system[KS04, fig. 2]. Kranenburg et al. interpret

this as an effect of alcohol onto the structure of the bilayer, and explain the rippled structure as a coexistence of the interdigitated phase $L_{\beta I}$ and the tilted gel phase $L_{\beta'}$. In the light of our studies, these results can possibly be reinterpreted: as described, the alcohols enrich mainly in the head region of the bilayers. This leads to an effectively increased head size and consequently to the head-tail size mismatch required for the formation of the tilted gel phase as well as the ripple phases.

In 2005, de Vries *et al* found a rippled structure in simulations of a *united atom model* of DPPC. The structure is identical to the structure found in this work, and it contains all of its features. However, as the model is more costly and uses many more parameters, it was not possible to systematically study the ripple phase. Furthermore, the symmetric ripple structure was not observed, presumably due to the smaller size of the simulations.

Theories

The only theory that catches certain aspects of the ripple phase structures from this work is that of Carlson and Sethna [CS87]. The theory predicted, that splay may play an important role for the ripple phase. Their one-dimensional static theory is purely based on the packing competition between heads and tails. It has two parameters, the head-tail size mismatch, and the relative strength of the head-tail interaction compared to the tail-tail interaction, where the so-called "head-tail interaction" acts on the tilt of a single lipid and prefers untilted lipids. In this theory, a large region is found, where the model bilayer exhibits a "modulated phase" with splayed domains, when the head-tail size mismatch is so large, that tilting all the lipids costs more energy than is lost in the gaps between two splayed domains where the tilt is reset. The gaps are assumed to be filled with disordered lipid chains, resulting in a structure similar to the structure sketched in figure 8.18(a).

However, the theory fails to give explanations for some of the features of the ripple phases. On the one hand, the theory can explain only one of the ripple phases, and it does not provide an explanation for the second occurrence of the phase, nor does it provide an explanation for the fact that the symmetric ripple phase $P_\beta^{(mst)}$ has roughly the double repeat distance of the asymmetric ripple phase $P_{\beta'}$. Furthermore, it can not provide a direct explanation for the asymmetry of the ripple phase $P_{\beta'}$ and instead assumes internal asymmetries of the lipid molecules as being responsible.

8.5. Conclusions

In this chapter it was shown, that the bilayer model possesses two occurences of a rippled phase close to the main transition. Furthermore, evidence was presented that the structures found in the simulations indeed correspond to the long-sought-after ripple phases $P_{\beta'}$ and $P_\beta^{(mst)}$ of phospholipid bilayers.

The simulations could shed light on the microscopical details of the ripple phases, and an explanation for the behaviour was given. The central element gouverning the

ripple phases is the splay of the lipids in the phases, which depends on the delivate interplay between the head-tail mismatch and the head-head interaction.

A. Volume moves

In a simulation at constant pressure, it is necessary to do *volume moves* every few Monte-Carlo steps.

In a volume move, the side length of the periodic system box and the position of each bead are scaled by a randomly chosen constant factor in the given dimension. To ensure detailed balance, the factor is given by $1 + \frac{r\,\Delta L_{\max}}{L_d}$, with $r \in [-1, 1]$ being a uniformly distributed random number, $\Delta L_{\max}$ being the maximal volume move range and L being the side length of the system box in the given dimension.

The grand-canonical Hamiltonian for N particles in one dimension is

$$H(\{\Phi_i\}, V) = \sum U(\Phi_i, \Phi_j) + pL \tag{A.1}$$

The grand-canonical partition function is

$$\begin{aligned} Z &= \int dL \int_0^L d^d\Phi_1 \ldots \int_0^L d^d\Phi_N e^{-\beta(\sum U(\Phi_i,\Phi_j)+pL)} && \text{(A.2)} \\ &= \int dL L^N \int_0^1 d^d s_1 \ldots \int_0^1 d^d s_N e^{-\beta(\sum U'(s_i,s_j)+pL)} && \text{(A.3)} \\ &= \int dL \int_0^1 d^d s_1 \ldots \int_0^1 d^d s_N e^{-\beta(\sum U'(s_i,s_j)+pL-\beta N \ln L)} && \text{(A.4)} \end{aligned}$$

where the s_i are normalised coordinates. This results in an effective Hamiltonian of

$$H_{eff}(\{s_i\}, L) = \sum U'(s_i, s_j) + pL - \beta N \ln L \tag{A.5}$$

In equation A.2, the "number of possibilites" for the particle coordinates depends on L (is proportional to L^N), whereas in equation A.4 the factor L^N needs to be introduced explicitly. This means, that the $\beta N \ln L$-term is needed in any MC-simulation where the volume is changed.

B. Acknowledgements

In the following, I would like to thank all those that helped with this thesis in various ways:

- **Prof. Dr. Friederike Schmid** for giving me the possibility to do this interesting project, for the long discussions, and for bringing me into the subject of soft-matter research. But most important, I would like to thank her for her incredible patience with me, who needed much longer to finish this thesis than anbody had anticipated (including myself).
- **Prof. Dr. Helge Ritter** for being willing to mentor this thesis, even though it has only very weak connections to the subjects of his work.
- **Eva Anna**, **Jule Svenja**, **Kaja Lena**, just for being there.
- **My parents** for bringing me up to where I am today.
- **Martin Streek**, **Beate West**, **Ulf Schiller**, **Britta Vogel**, **Martin Weinhold**, **Dominik Düchs** and **Harald Lange** for being the best office colleagues I can think of.
- My new boss **PD Dr. Christian Holm** who accepted me in a postdoctoral position more than one year before I actually had earned it.
- **Marieke Kranenburg** for inspring discussions.
- **V.A. Ragunathan** for providing the EDMs of the ripple phases.
- The **Deutsche Forschungsgemeinschaft (DFG)** for the funding in the frame of the Sonderforschungsbereich 613.
- All configuration snapshots were done using the great visualisation software VMD [HDS96].

C. Bibliography

[AT87] M. P. Allen and D. J. Tildesley. *Computer Simulation of Liquids*. Oxford Science Publications. Clarendon Press, Oxford, 1st edition, 1987.

[BB04] G. Brannigan and F. L. H. Brown. Solvent-free simulations of fluid membrane bilayers. *J. Chem. Phys.*, 120(2):1059–1071, 2004.

[BLCG92] T. Berners-Lee, R. Cailleau, and J.-F. Groff. The world-wide web. *Computer networks and ISDN systems*, 25(4-5):454–459, 1992.

[BPB05] G. Brannigan, P.F. Phillips, and F.L.H. Brown. Flexible lipid bilayers in implicit solvent, 25 February 2005.

[BSV92] R. Benzi, S. Succi, and M. Vergassola. The lattice boltzmann equation: theory and applications. *Physics Reports*, 222(3):145–197, 1992.

[BTB04] G. Brannigan, A.C. Tamboli, and F.L.H. Brown. The role of molecular shape in bilayer elasticity and phase behavior. *J. Chem. Phys.*, 121(7):3259–3271, 2004.

[But05] D. Butler. Science in the web age: Joint efforts. *Nature*, 438(7068):548–549, December 2005.

[CCM92] H. Chen, S. Chen, and W. H. Matthaeus. Recovery of the navier-stokes equations using a lattice-gas boltzmann method. *Phys. Rev. A*, 45(8):R5339–R5342, April 1992.

[CD05] I. R. Cooke and M. Deserno. Solvent-free model for self-assembling fluid bilayer membranes: Stabilization of the fluid phase based on broad attractive tail potentials. *J. Chem. Phys.*, 123(22):224710, 2005.

[CJR+06] S. Cikic, S. Jeschke, T. Richter, U. Sinha, and C. Thomsen. Networked experiments and scientific resource sharing in cooperative knowledge spaces. *ism*, 0:953–958, 2006.

[CKD05] I.R. Cooke, K. Kremer, and M. Deserno. Tunable generic model for fluid bilayer membranes. *Phys. Rev. E*, 72(1 Pt 1):011506, 2005.

[Con87] J. Conklin. A survey of hypertext. Technical Report STP356-86, Rev. 2, Microelectronics & Computer Technology Corporation, Austin, Texas, 1987.

[CS87] J.M. Carlson and J.P. Sethna. Theory of the ripple phase in hydrated phospholipid bilayers. *Phys. Rev. A*, 36(7):3359–3374, 1987.

[CŽP81] G. Cevc, B. Žekš, and R. Podgornik. The undulations of hydrated phospholipid multilayers may be due to water-mediated bilayer-bilayer interactions. *Chemical Physics Letters*, 84:209–212, December 1981.

[dGP93] P.G de Gennes and J. Prost. *The physics of liquid crystals*. Oxford Science Publications, 2 edition, 1993.

[DML91] J.M. Drouffe, A.C. Maggs, and S. Leibler. Computer simulations of self-assembled membranes. *Science*, 254(5036):1353–1356, 1991.

[DS01] D. Düchs and F. Schmid. Phase behaviour of amphiphilic monolayers: Theory and simulation. *J. Phys. Cond. Matter*, 13:4853, 2001.

[DYP93] T. Darden, D. York, and L. Pedersen. Particle mesh ewald: An $N\log(N)$ method for ewald sums in large systems. *J. Chem. Phys.*, 98:10089, 1993.

[ED05] D. J. Earl and M. W. Deem. Parallel tempering: Theory, applications, and new perspectives. *Phys. Chem. Chem. Phys.*, 7:3910 – 3916, 2005.

[EPB+95] U. Essmann, L. Perera, M. L. Berkowitz, T. Darden, H. Lee, and L. Pedersen. A smooth particle mesh ewald method. *J. Chem. Phys.*, 103:8577, 1995.

[ET05] E. Elrufaie and D. A. Turner. A wiki paradigm for use in it courses. *itcc*, 2:770–771, 2005.

[Far03] O. Farago. "water-free" computer model for fluid bilayer membranes. *J. Chem. Phys.*, 119:596, 2003.

[FCB06a] M. Ferrario, G. Ciccotti, and K. Binder, editors. *Computer Simulations in Condensed Matter: from Materials to Chemical Biology Volume 1*, volume 703 of *Lecture Notes in Physics*. Springer, Berlin, Germany, 2006.

[FCB06b] M. Ferrario, G. Ciccotti, and K. Binder, editors. *Computer Simulations in Condensed Matter: from Materials to Chemical Biology Volume 2*, volume 704 of *Lecture Notes in Physics*. Springer, Berlin, Germany, 2006.

[FS02] D. Frenkel and B. Smit. *Understanding Molecular Simulation*. Academic Press, San Diego, second edition, 2002.

[FSFM82] M. S. Falkovitz, M. Seul, H. L. Frisch, and H. M. McConnell. Theory of periodic structures in lipid bilayer membranes. *PNAS*, 79(12):3918–3921, 1982.

[FY96] Y. Fang and J. Yang. Role of the bilayer–bilayer interaction on the ripple structure on supported bilayers in solution. *J. Phys. Chem.*, 100:15614–15619, 1996.

[GC00] J. Gillies and R. Cailliau. *How the Web was Born: The Story of the World Wide Web*. Oxford University Press, USA, 2000.

[GDT+06] M. Galassi, J. Davies, J. Theiler, B. Gough, G. Jungman, M. Booth, and F. Rossi. *GNU Scientific Library Reference Manual*. Network Theory Ltd, 2nd edition, 2006.

[Gen89] R.B. Gennis. *Biomembranes: Molecular Structure and Function*. Springer-Verlag, New York, 1989.

[GHJV95] Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. *Design patterns: elements of reusable object-oriented software*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1995.

[Gil05] J. Giles. Internet encyclopaedias go head to head. *Nature*, 438(7070):900–901, December 2005.

[GL98] R. Götz and R. Lipowsky. Computer simulations of bilayer membranes: Self-assembly and interfacial tension. *J. Chem. Phys*, 108(17):7397, May 1998.

[GR87] L. Greengard and V. Rhoklin. A fast algorithm for particle simulations. *J. Comp. Phys.*, 73:325, 1987.

[HDS96] W. Humphrey, A. Dalke, and K. Schulten. VMD: Visual molecular dynamics. *Journal of Molecular Graphics*, 14:33–38, 1996.

[Hei00] T. Heimburg. A model for the lipid pretransition: coupling of ripple formation with the chain-melting transition. *Biophys. J.*, 78(3):1154–65, 2000.

[HGE73] R. W. Hockney, S. P. Goel, and J. W. Eastwood. A 10000 particle molecular dynamics model with long range forces. *Chemical Physics Letters*, 21(3):589–591, 1973.

[HK92] P. J. Hoogerbrugge and J. M. V. A. Koelman. Simulating microscopic hydrodynamic phenomena with dissipative particle dynamics. *Europhysics Letters*, 19(3):155–160, 1992.

[IMN76] J.N. Israelachvili, D.J. Mitchell, and B.W. Ninham. Theory of self-assembly of hydrocarbon amphiphiles into micelles and bilayers. *J. Chem. Soc. Faraday Transactions 2*, 72:1525 – 1568, 1976.

[Isr92] J.N. Israelachvili. *Intermolecular and Surface Forces*. Academic Press, London, U.K. – San Diego, CA, USA, second edition, 1992.

[JTS06] S. Jeschke, C. Thomsen, and U. Sinha. Collaborative working environment for virtual and remote experiments in nanoscience and nanotechnologies. In Thomas Reeves and Shirley Yamashita, editors, *Proceedings*

of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2006, pages 2055–2060, Honolulu, Hawaii, USA, October 2006. AACE.

[KC94a] R. Koynova and M. Caffrey. Phases and phase transitions of the glycoglycerolipids. *Chem. Phys. Lipids.*, 69(3):181–207, 1994.

[KC94b] R. Koynova and M. Caffrey. Phases and phase transitions of the hydrated phosphatidylethanolamines. *Chem. Phys. Lipids*, 69(1):1–34, 1994.

[KC95] R. Koynova and M. Caffrey. Phases and phase transitions of the sphingolipids. *Biochim. Biophys. Acta*, 1255(3):213–36, 1995.

[KC98] R. Koynova and M. Caffrey. Phases and phase transitions of the phosphatidylcholines. *Biochim. Biophys. Acta*, 1376(1):91–145, 1998.

[KKT96] R. Koynova, A. Koumanov, and B. Tenchov. Metastable rippled gel phase in saturated phosphatidylcholines: calorimetric and densitometric characterization. *Biochim. Biophys. Acta*, 1285(1):101–8, 1996.

[KLC+03] T. Kaasgaard, C. Leidy, J.H. Crowe, O.G. Mouritsen, and K. Jorgensen. Temperature-Controlled Structure and Kinetics of Ripple Phases in One- and Two-Component Supported Lipid Bilayers. *Biophys. J.*, 85(1):350–360, July 2003.

[KLS04] M. Kranenburg, C. Laforge, and B. Smit. Mesoscopic simulations of phase transitions in lipid bilayers. *Phys. Chem. Chem. Phys.*, 6(19):4531 – 4534, 2004.

[KNS04] M. Kranenburg, J.P. Nicolas, and B. Smit. Comparison of mesoscopic phospholipid-water models. *Phys. Chem. Chem. Phys.*, 6(16):4142 – 4151, 2004.

[KS04] M. Kranenburg and B. Smit. Simulating the effect of alcohol on the structure of a membrane. *FEBS Lett.*, 568(1-3):15–18, 2004.

[KTNL+00] J. Katsaras, S. Tristram-Nagle, Y. Liu, R.L. Headrick, E. Fontes, P.C. Mason, and J.F. Nagle. Clarification of the ripple phase of lecithin bilayers using fully hydrated, aligned samples. *Phys. Rev. E*, 61(5 Pt B):5668–77, 2000.

[KVS03a] M. Kranenburg, M. Venturoli, and B. Smit. Molecular simulations of mesoscopic bilayer phases. *Phys. Rev. E*, 67(6), 2003.

[KVS03b] M. Kranenburg, M. Venturoli, and B. Smit. Phase behavior and induced interdigitation in bilayers studied with dissipative particle dynamics. *J. Phys. Chem. B*, 107(41):11491 – 11501, 16 October 2003.

[LAMH06] H.-J. Limbach, A. Arnold, B. A. Mann, and C. Holm. ESPResSo – an extensible simulation package for research on soft matter systems. *Comp. Phys. Comm.*, 174(9):704–727, May 2006.

[Lar77] K. Larsson. Folded bilayers – an alternative to the rippled lamellar lecithin structure. *Chem. Phys. Lipids*, 20(3):225–228, 1977.

[LHvdS01] E. Lindahl, B. Hess, and D. van der Spoel. Gromacs 3.0: A package for molecular simulation and trajectory analysis. *J. Mol. Mod.*, 7:306–317, 2001.

[Lip02] R. Lipowsky. Domains and Rafts in Membranes: Hidden Dimensions of Selforganization. *J. Bio. Phys.*, 28:195–210, 2002.

[LKC+02] C. Leidy, T. Kaasgaard, J.H. Crowe, O.G. Mouritsen, and K. Jorgensen. Ripples and the formation of anisotropic lipid domains: imaging two-component supported double bilayers by atomic force microscopy. *Biophys. J.*, 83(5):2625–33, 2002.

[LM93] T.C. Lubensky and F.C. MacKintosh. Theory of "Ripple" Phases of Lipid Bilayers. *Phys. Rev. Lett.*, 71(10):1565–1568, 1993.

[LMDB84] D. Lichtenberg, M. Menashe, S. Donaldson, and R. L. Biltonen. Thermodynamic characterization of the pretransition of unilamellar dipalmitoylphosphatidylcholine vesicles. *Lipids*, 19(6):395–400, Jun 1984.

[LMKS03] C. Loison, M. Mareschal, K. Kremer, and F. Schmid. Thermal fluctuations in a lamellar phase of a binary amphiphile–solvent mixture: a molecular-dynamics study. *J. Chem. Phys.*, 119(24):13138–13148, 2003.

[LMS04] C. Loison, M. Mareschal, and F. Schmid. Pores in bilayer membranes of amphiphilic molecules: coarse-grained molecular dynamics simulations compared with simple mesoscopic models. *J. Chem. Phys.*, 121(4):1890–900, 2004.

[LMS05] C. Loison, M. Marechal, and F. Schmid. Fluctuations and Defects in Lamellar Stacks of Amphiphilic Bilayers. *Comp. Phys. Comm.*, 2005.

[LS05] O. Lenz and F. Schmid. A simple computer model for liquid lipid bilayers. *J. Mol. Liquids*, 117(1-3):147–152, 23 February 2005.

[LS07] O. Lenz and F. Schmid. Structure of symmetric and asymmetric "ripple" phases in lipid bilayers. *Phys. Rev. Lett.*, 98(5):058104, 2007.

[LZR98] P. Lague, M.M. Zuckermann, and B. Roux. Protein inclusion in lipid membranes: A theory based on the hypernetted chain integral equation. *Faraday Disc.*, 111:165, 1998.

[Mey96] H. W. Meyer. Pretransition-ripples in bilayers of dipalmitoylphosphatidylcholine: undulation or periodic segments? A freeze-fracture study. *Biochim. Biophys. Acta*, 1302(2):138–144, Jul 1996.

[MFLM84] M. Marder, H. L. Frisch, J. S. Langer, and H. M. McConnell. Theory of the Intermediate Rippled Phase of Phospholipid Bilayers. *PNAS*, 81(20):6559–6561, 1984.

[MGE+99] P.C. Mason, B.D. Gaulin, R. Epand, G.D. Wignall, and J.S. Lin. Small angle neutron scattering and calorimetric studies of large unilamellar vesicles of the phospholipid dipalmitoylphosphatidylcholine. *Phys. Rev. E*, 59:3361–3367, 1999.

[MJ97] O.G. Mouritsen and K. Jorgensen. Small-scale lipid-membrane structure: simulation versus experiment. *Curr. Opn. Struct. Biol.*, 7:518–527, 1997.

[MS95] David R. Musser and Atul Saini. *The STL Tutorial and Reference Guide: C++ Programming with the Standard Template Library*. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA, 1995.

[MVM04] S.J. Marrink, A.H. de Vries, and A.E. Mark. Coarse Grained Model for Semiquantitative Lipid Simulations. *J. Chem. Phys. B*, 108:750–760, 2004.

[MYKH93] S. Matuoka, H. Yao, S. Kato, and I. Hatta. Condition for the appearance of the metastable $P_{\beta'}$ phase in fully hydrated Phosphatidylcholines at studies by small-angle x-ray-diffraction. *Biophys. J.*, 64(5):1456 – 1460, 5 1993.

[Nog02] H. Noguchi. Fusion and toroidal formation of vesicles by mechanical forces: A brownian dynamics simulation. *J. Chem. Phys.*, 117(17):8130–8137, 11 November 2002.

[NT01a] H. Noguchi and M. Takasu. Fusion pathways of vesicles: A brownian dynamics simulation. *J. Chem. Phys.*, 115(20):9547–9551, 11 November 2001.

[NT01b] H. Noguchi and M. Takasu. Self-assembly of amphiphiles into vesicles: A brownian dynamics simulation. *Phys. Rev. E*, 64(4), October 2001.

[NT02a] H. Noguchi and M. Takasu. Adhesion of nanoparticles to vesicles: A brownian dynamics simulation. *Biophys. J.*, 83(1):299–308, July 2002.

[NT02b] H. Noguchi and M. Takasu. Structural changes of pulled vesicles: A brownian dynamics simulation. *Phys. Rev. E*, 65(5), May 2002.

[NWV06] Björn Naundorf, Fred Wolf, and Maxim Volgushev. Unique features of action potential initiation in cortical neurons. *Nature*, 440(7087):1060–1063, Apr 2006.

[NZTN+96] J.F. Nagle, R. Zhang, S. Tristram-Nagle, W. Sun, H.I. Petrache, and R.M. Suter. X-ray structure determination of fully hydrated L_α phase dipalmitoylphosphatidylcholine bilayers. *Biophys. J.*, 70(3):1419–31, 1996.

[PBW+05] James C. Phillips, Rosemary Braun, Wei Wang, James Gumbart, Emad Tajkhorshid, Elizabeth Villa, Christophe Chipot, Robert D. Skeel, Laxmikant Kalé, and Klaus Schulten. Scalable molecular dynamics with namd. *J. Comput. Chem.*, 26(16):1781–1802, 2005.

[PS82] P.A. Pearce and H. L. Scott. Statistical mechanics of the ripple phase in lipid bilayers. *J. Chem. Phys.*, 77(2):951–958, 1982.

[PSK05] M. Praprotnik, L. Delle Site, and K. Kremer. Adaptive resolution molecular-dynamics simulation: Changing the degrees of freedom on the fly. *J. Chem. Phys.*, 123(22):224106, 2005.

[RPR+00] M. Rappolt, G. Pabst, G. Rapp, M. Kriechbaum, H. Amenitsch, C. Krenn, S. Bernstorff, and P. Laggner. New evidence for gel-liquid crystalline phase coexistence in the ripple phase of phosphatidylcholines. *Eur. Biophys. J.*, 29(2):125–33, 2000.

[Sal00] T. Salditt. Structure and fluctuations of highly oriented phospholipid membranes. *Curr. Opn. Coll. Int. Sci.*, 5(1-2):19 – 26, MAR 2000.

[SB98] T. Sintes and A. Baumgärtner. Membrane-mediated protein attraction. a monte carlo study. *Physica A*, 249:571, 1998.

[Sch06] F. Schmid, 2006. oral communication.

[Sco02] H.L. Scott. Modeling the lipid component of membranes. *Curr. Opn. Struct. Biol.*, 12(4):495–502, August 2002.

[SCW83] M. B. Schneider, W. K. Chan, and W. W. Webb. Fast diffusion along defects and corrugations in phospholipid $P_{\beta'}$ liquid crystals. *Biophys. J.*, 43(2):157–165, Aug 1983.

[SDLL04] F. Schmid, D. Düchs, O. Lenz, and C. Loison. Amphiphiles at interfaces: Simulation of structure and phase behavior. In N. Attig, K. Binder, H. Grubmüller, and K. Kremer, editors, *Computational Soft Matter: From Synthetic Polymers to Proteins*, volume 23 of *NIC series*. Research Centre Jülich, 2004.

[SDLW07] F. Schmid, D. Düchs, O. Lenz, and B. West. A generic model for lipid monolayers, bilayers, and membranes. *Comp. Phys. Comm.*, 177(1-2):168, 2007.

[SL97] F. Schmid and H. Lange. Influence of head group size on the direction of tilt in langmuir monolayers. *J. Chem. Phys.*, 106(9):3757, 1997.

[SN72] S. J. Singer and G. L. Nicholson. The fluid mosaic model of the structure of cell membranes. *Science*, 175(23):720–731, 18 February 1972.

[SRK00] K. Sengupta, V.A. Raghunathan, and J. Katsaras. Novel structural features of the ripple phase of phospholipids. *Europhys. Lett.*, 49(6):722 – 728, MAR 2000.

[SRK03] K. Sengupta, V.A. Raghunathan, and J. Katsaras. Structure of the ripple phase of phospholipid multibilayers. *Phys. Rev. E*, 68(3), September 2003.

[SS99] C. Stadler and F. Schmid. Phase behaviour of grafted chain molecules: Effect of head size and chain length. *J. Chem. Phys.*, 110:9697, 1999.

[SSL99] F. Schmid, C. Stadler, and H. Lange. Theoretical modelling of langmuir monolayers. *Colloids and Surfaces A*, 149(1-3):301, 1999.

[SSR+01a] J. C. Shelley, M. Y. Shelley, R. C. Reeder, S. Bandyopadhyay, P. B. Moore, and M. L. Klein. Simulations of phospholipids using a coarse grain model. *J. Phys. Chem. B*, 105:9785–9792, 2001.

[SSR+01b] J.C. Shelley, M.Y. Shelley, R.C. Reeder, S. Bandyopadhyay, and M. L. Klein. A coarse grain model for phospholipid simulations. *J. Phys. Chem. B*, 105:4464–4470, 2001.

[Sta98] C. Stadler. *Monte Carlo-Simulationen von Langmuir-Monolagen*. PhD thesis, Universität Mainz, 1998.

[STNSN96] W.J. Sun, S. Tristram-Nagle, R.M. Suter, and J.F. Nagle. Structure of the ripple phase in lecithin bilayers. *Proc. Natl. Acad. Sci. USA*, 93(14):7008–12, 1996.

[TLR73] A. Tardieu, V. Luzzati, and F.C. Reman. Structure and polymorphism of the hydrocarbon chains of lipids: a study of lecithin-water phases. *J. Mol. Biol.*, 75(4):711–33, 1973.

[TMB97] D.P Tieleman, S.J. Marrink, and H.J.C. Berendsen. A computer perspective of membranes: molecular dynamics studies of lipid bilayer systems. *Biochim. Biophys. Acta: Rev. Biomembr.*, 1331(3):235–270, 1997.

[TNN04] S. Tristam-Nagle and J.F. Nagle. Lipid bilayers: thermodynamics, structure, fluctuations, and interactions. *Chem. Phys. Lip.*, 127:3–14, 2004.

[TTK97] D.J. Tobias, K. Tu, and M.L. Klein. Atomic-scale molecular dynamics simulations of lipid membranes. *Curr. Opin. Coll. Int. Sci.*, 1997.

[TUK+95] T. Takeda, S. Ueno, H. Kobayashi, S. Komura, H. Seto, and Y. Toyoshima. Small-angle-scattering study of the structural phase transition in the dipalmitoylphosphatidylcholine (dppc)-water-salt system. *Physica B*, 213&214:763–765, 1995.

[ULA+02] A. Uhlherr, S.J. Leak, N.E. Adam, P.E. Nyberg, M. Doxastakis, V.G. Mavrantzas, and D.N. Theodorou. Large scale atomistic polymer simulations using Monte Carlo methods for parallel vector processors. *Comp. Phys. Comm.*, 144(1):1–22, 15 March 2002.

[VSM+03] G. Vereb, J. Szollosi, J. Matko, P. Nagy, T. Farkas, L. Vigh, L. Matyus, T.A. Waldmann, and S. Damjanovich. Dynamic, yet structured: The cell membrane three decades after the Singer-Nicolson model. *Proc. Natl. Acad. Sci. USA*, 100(14):8053–8, 2003.

[VYMM05] A.H. de Vries, S. Yefimov, A.E. Mark, and S.J. Marrink. Molecular structure of the lecithin ripple phase. *Proc. Natl. Acad. Sci. USA*, 102(15):5392–6, 2005.

[WSG81] R. J. Wittebort, C. F. Schmidt, and R. G. Griffin. Solid-state carbon-13 nuclear magnetic resonance of the lecithin gel to liquid-crystalline phase transition. *Biochemistry*, 20(14):4223–4228, July 1981.

[WW89] D. C. Wack and W. W. Webb. Synchrotron x-ray study of the modulated lamellar phase $P_{\beta'}$ in the lecithin-water system. *Phys. Rev. A*, 40(5):2712–2730, 1989.

[WZ96] J.T. Woodward and J.A. Zasadzinski. Amplitude, wave form, and temperature dependence of bilayer ripples in the $P_{\beta'}$ phase. *Phys. Rev. E*, 53(4):R3044–R3047, 1996.

[YMTH91] H. Yao, S. Matuoka, B. Tenchov, and I. Hatta. Metastable ripple phase of fully hydrated dipalmitoylphosphatidylcholine as studied by small-angle x-ray scattering. *Biophys. J.*, 59(1):252 – 255, January 1991.

Index