UNIVERSIDADE NOVA DE LISBOA

MScCBBi

MASTER IN
COMPUTATIONAL BIOLOGY
& BIOINFORMATICS

SPECIALIZATION Biosystems Simulation for Life and Health Sciences

João Filipe Reis Catarino
BSc in Biomedical Sciences

Parameterization of bacterial peptidoglycan for state-of-the-art molecular dynamics coarse-grain force fields

Supervisor: Dr. Manuel N. Melo, PhD, ITQB-NOVA

Dec, 2024

ITQB NOVA | NOVA SCHOOL OF SCIENCE & TECHNOLOGY | NOVA MEDICAL SCHOOL | NOVA IMS Information Management School | DESDE 1902 INSTITUTO DE HIGIENE E MEDICINA TROPICAL UNIVERSIDADE NOVA DE LISBOA

UNIVERSIDADE NOVA DE LISBOA

MScCBBi

MASTER IN
COMPUTATIONAL BIOLOGY
& BIOINFORMATICS

SPECIALIZATION Biosystems Simulation for Life and Health Sciences

João Filipe Reis Catarino
BSc in Biomedical Sciences

Parameterization of bacterial peptidoglycan for state-of-the-art molecular dynamics coarse-grain force fields

Supervisor: Dr. Manuel N. Melo, PhD, ITQB-NOVA

Dec, 2024

ITQB NOVA | NOVA SCHOOL OF SCIENCE & TECHNOLOGY | NOVA MEDICAL SCHOOL | NOVA IMS Information Management School | DESDE 1902 INSTITUTO DE HIGIENE E MEDICINA TROPICAL UNIVERSIDADE NOVA DE LISBOA

Parameterization of bacterial peptidoglycan for state-of-the-art molecular dynamics coarse-grain force fields.

Copyright © João Filipe Reis Catarino, Instituto de Tecnologia Química e Biológica António Xavier, NOVA University Lisbon.
The Instituto de Tecnologia Química e Biológica António Xavier and the NOVA University Lisbon have the right, perpetual and without geographical boundaries, to file and publish this dissertation through printed copies reproduced on paper or on digital form, or by any other means known or that may be invented, and to disseminate through scientific repositories and admit its copying and distribution for non-commercial, educational or research purposes, as long as credit is given to the author and editor.

This document was created with the (pdf/Xe/Lua)LaTeX processor and the NOVAthesis template (v7.1.27) [1].

Acknowledgements

I am fully aware that no one man is an island and that any accomplishments can not be fulfilled without the influence of others. None of the work done throughout this year (and 3 months) would have been possible without the support of the people around me.

First, I would like to thank Dr. Manuel N. Melo for the opportunity he gave me to work with him as well the remarkable help and encouragement that he provided throughout this project.

To my friends at the Multiscale Modeling Lab, thank you for being so welcoming and for bringing joy into what could have been mundane work days.

A special thanks to Fernando, for helping me when I was too embarrassed to ask Manel and for always keeping me on my toes.

To Nuno, for quickly becoming one of my closest and best friends and being someone I know I can always rely on, be it in better or worse times.

Thank you to all the people who have helped me get to where I am, even if life has driven us apart.

Para a minha família, obrigado pela paciência e por estarem ao meu lado toda a minha vida.

Finalmente, ao Magnus, por ser tão fofo.

Abstract

The cell wall is one of the most defining characteristics of bacteria. It plays a crucial role in cell integrity while regulating the passive transport of many nutrients and proteins. The main culprit behind the cell wall properties is the peptidoglycan polymer. It is composed of a backbone of two alternating sugars (NAG and NAM) and a pentapeptide linked to every NAM sugar. This pentapeptide is capable of cross-linking with other peptides to create large meshes.

Being able to computationally simulate peptidoglycan meshes at the molecular level would possibilitate several research venues, such as drug targeting and screening towards development of novel medicine. However, simulating these meshes is unfeasible using standard all-atom (AA) simulations due to their considerable size. Coarse-grain (CG) molecular dynamics trade molecular detail for lighter calculation and larger time steps, reaching time and size scales compatible with simulation of the peptidoglycan mesh. In this work, we develop a peptidoglycan model for one of the most widely used coarse-grain frameworks for biomolecular simulations, the Martini 3 force field.

Martini 3 parameterization entails comparing behavior from tentative CG models with that of an AA counterpart. Using this information, CG parameters are then iteratively tweaked to best reproduce target behavior. Here, the glycan chain of peptidoglycan was first parameterized using separate simulations of single disaccharide units ran with existing AA models. These parameters were then extended to successfully represent a full glycan chain. Next, peptide stems were attached to each NAM residue and its parameters were determined based on the standard Martini 3 guidelines. Additionally, the parameterization process involved the tailoring of specific intramolecular non-bonded potentials to reproduce the rate and directionality preference with which disaccharide units flipped between two relevant torsional states. This is one of the first Martini 3 parameterization effort that takes these aspects into account as well.

Keywords: Bacterial cell wall, Peptidoglycan, Model parameterization, Coarse-grain, Martini 3

Resumo

A parede celular é uma das mais importantes características das bactérias. Esta desempenha um papel crucial na integridade celular ao regular o transporte passivo de nutrientes e proteínas. A molecula principal por detrás das propriedades da parede celular é o polímero peptidoglicano. Este é composto por uma cadeia de dois glucídos (NAG e NAM) com um pentapeptído ligado a cada NAM. Este pentapeptído é também capaz de formar ligações cruzadas com outras unidades para criar extensas malhas.

A capacidade de simular malhas de peptidoglicano a nível molecular possibilita diferentes vias de investigação. Por exemplo, a identificação de novos alvos e a triagem de fármacos para o desenvolvimento de novos medicamentos. No entanto, simular estas malhas é impratico se utilizar-mos métodos atomísticos (AA) devido à quantidade considerável de átomos. A dinâmica molecular *coarse-grain* (CG) troca detalhes moleculares por cálculos mais leves e intervalos de tempo maiores. Isto permite alcançar escalas de tempo e tamanhos compatíveis com simulações da malha do peptidoglicano. Neste trabalho, desenvolvemos um modelo do peptidoglicano para um dos *frameworks* de CG mais amplamente utilizados em simulações biomoleculares, o campo de força Martini 3.

A parametrização em Martini 3 envolve comparar o comportamento de modelos CG provisórios com um equivalente atomístico. Com base nesses resultados, os parâmetros CG são então ajustados iterativamente para reproduzir o melhor possível o comportamento alvo. Inicialmente, a cadeia de glicanos do peptidoglicano foi parametrizada usando simulações separadas de unidades de dissacarídeos únicas com base em modelos AA existentes. Esses parâmetros foram então extendidos para representar com sucesso uma cadeia completa de glicanos. De seguida, conectou-se o pentapeptído a cada resíduo NAM, e seus parâmetros foram determinados com base nas diretrizes padrão para Martini 3. Adicionalmente, o processo de parametrização utilizou forças para reproduzir a taxa média e a preferência de direcionalidade com que as unidades de dissacarídeos alternavam entre dois estados relevantes. Este é um dos primeiros esforços de parametrização em Martini 3 que leva em consideração esses aspectos.

Palavras-chave: Parede celular bacteriana, Peptidoglicano, Modelo *coarse-grain*, Martini 3

Contents

List of Figures

List of Tables

Acronyms

AA	All-Atom
CG	Coarse-Grain
CHARMM	Chemistry at HARvard Macromolecular Mechanics
COG	Center Of Geometry
COW	Center Of Weights
CV	Collective Variable
GaFF	General AMBER Force Field
MD	Molecular Dynamics
MM	Molecular Mechanics
NMR	Nuclear Magnetic Resonance
QM	Quantum Mechanical
R	Regular bead size
S	Small bead size
SASA	Solvent Accessible Surface Area
T	Tiny bead size
VS	Virtual Site

Symbols

Å	Angstrom, a unit of distance equal to 10^{-10} meters
ϵ	Epsilon, controls the well-depth of a Lennard-Jones potential
K	Kelvin, a unit of temperature
nm	Nanometer, a unit of distance equal to 10^{-9} meters
ns	Nanosecond, a unit of time equal to 10^{-9} seconds
ϕ	Phi, $O_5 - C_1 - O_4 - C_4$ dihedral angle of glycosidic link
ps	Picosecond, a unit of time equal to 10^{-12} seconds
ψ	Psi, $C_1 - O_4 - C_4 - C_5$ dihedral angle of glycosidic link
rad	Radian, a unit of angle
σ	Sigma, distance at which a Lennard-Jones potential is zero

Chemical Symbols

ACN Amide functional group

D-Ala D-Alanine amino acid
D-Glu D-Glutamate amino acid
D-iGln D-isoGlutamine amino acid

L-Ala L-Alanine amino acid
L-Lys L-Lysine amino acid
LAC Lactic acid functional group

mDAP meso-Diaminopimelic acid

NAG N-Acetylglucosamine
NAM N-Acetylmuramic acid

PG Peptidoglycan

1 Introduction

1.1 The Bacterial Cell Wall

The bacterial cell wall stands out as one of the most defining characteristics of all bacterial cells. It rests outside of the cell and spans out and surrounds the entirety of the cytoplasmic membrane of most bacteria [2]. The cell wall helps maintain cell integrity as it protects bacteria from lysis by resisting the intracellular osmotic pressure caused by the cytosolic components [3]. This allows the cell to maintain a well-defined cell shape and volume, while doubling as an anchor point for other cells envelope components such as proteins [3]. Furthermore, it confers the bacteria cell the rigidity needed to sustain adverse impacts from the environment, while still being flexible enough to allow bacteria to dynamically change their shape and volume [3].

The main component behind the cell wall's properties is the peptidoglycan (PG) polymer and its architecture defines the most common bacteria classification, the Gram classification [4]. This classification arises from different architectures of PG existing between Gram-negative and Gram-positive bacteria, as seen in Fig. 1.1. The PG in Gram-positive bacteria is organized in multiple layers, which form a thick sheet that rests outside of their only cytoplasmic membrane [3, 4]. On the other hand, in Gram-negative bacteria, the PG is assembled in a monolayer which is relatively thin and located in-between the bacteria's outer and inner membrane (the periplasm) [3, 4]. As an example, the thickness of the bacterial cell wall for Gram-positive bacteria ranges from 20 to 80 nm, while Gram-negative bacteria have a much thinner bacterial cell wall that ranges from 1.5 to 10 nm [5].

1.2 Peptidoglycan

Peptidoglycan is a unique and essential structural element in the cell wall of almost all bacteria [3]. As the name suggests, it is made up of two main components, a glycan backbone comprised of repeating disaccharide units and a small peptide chain of two to five amino acid residues (depending on species) [3]. The number of repeated disaccharide

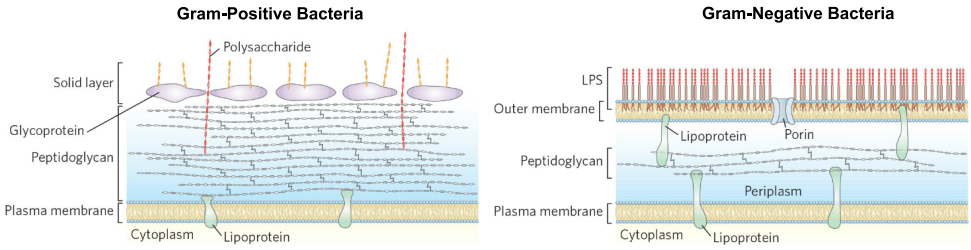

Figure 1.1: **Schematic of Gram-Positive and Gram-Negative bacterial cell wall.** Gram-positive bacteria have a multilayered thick cell wall which rests outside their only plasma membrane. In contrast, the peptidoglycan of Gram-negative bacteria rests inside of the periplasm and forms only a relatively thin sheet. Adapted from Lehninger. [6].

units varies both within the same cell wall and between bacteria themselves. As an example, the Gram-positive *S. aureus* has an average length of 18 disaccharide units, while the Gram-negative *E. coli* peptidoglycan ranges from 20 to 40 disaccharide units when in exponential growth [7]. Bacilii are an extreme example where the average chain length is between 50 to 250 disaccharide units [7].

The glycan backbone is consistent in all bacteria and is made up of two alternating sugars, N-acetylglucosamine (NAG or GlcNAc) and N-acetylmuramic acid (NAM or MurNAc) [3, 4]. NAG is an amide derivative of glucosamine by a N-link at C2 [4]. Successively, NAM is derived from NAG by the addition of a Lactic acid moiety at C3 [4]. These two alternating sugars are linked by a β-1,4 glycosidic bond to form long polymer of varying length [4]. In all Gram-negative bacteria and some Gram-positive, the terminal sugar in the glycan strand is not one of the two standard sugars (NAG or NAM), but instead a non-reducing alternative (1,6-anhydroMurNAc), which has an intramolecular ring between C1 and C6 [3, 4]. The structure of all aforementioned sugars can be consulted at Fig. 1.2.

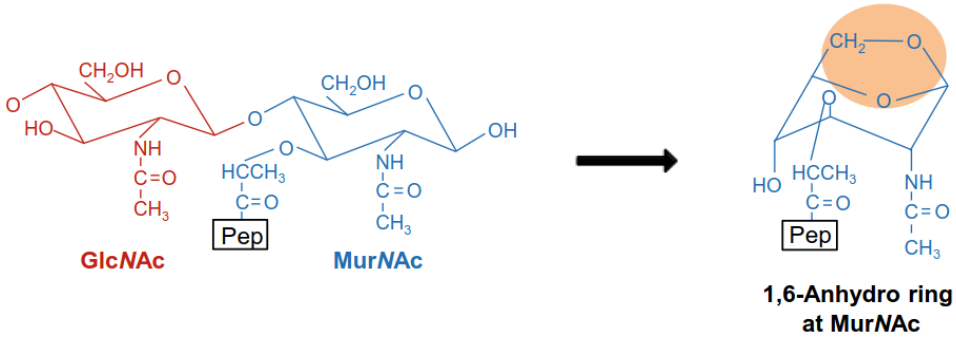

Figure 1.2: **Structures for the main sugars in the peptidoglycan glycan strand.** The peptidoglycan strand is made up of two alternating sugars N-acetylglucosamine (red) and N-acetylmuramic acid (blue). The terminal sugar of the glycan strand is a non-reducing MurNAc/NAM derivative with a intramolecular ring between C1 and C6. Adapted from Pazos et al. [4].

1.2.1 Peptide Strand

At every NAM monosaccharide, the lactyl moiety serves as the anchoring point to the short peptide of peptidoglycan [4]. This short peptide is capable of cross-linking with other peptide strands in other peptidoglycan molecules to form 3D mesh structures that are characteristic of cell wall bacteria and gives them the strength and flexibility needed to perform their function [3, 4].

The peptide anchored to NAM is most commonly made up of five amino acids, whose type varies from bacteria to bacteria. In Gram-negative bacteria, such as *E. coli*, the five amino acids usually are: L-Alanine (L-Ala), D-Glutamate (D-Glu), meso-Diaminopimelic acid (mDAP), and, finally, two D-Alanines (D-Ala) [3, 4]. All bonds between amino acids are normal peptide bonds with the exception of D-Glu and mDAP which are linked though D-Glu's side chain (isopeptide linkage) [3]. In Gram-positive bacteria, the D-Glu frequently undergoes amination of its α-carboxyl group transforming into D-isoGlutamine (D-iGln) and mDAP is swapped for another dibasic amino acid, most commonly L-Lysine (L-Lys) [4].

The presence of a dibasic amino acid, such as mDAP or L-Lys, is very important for the peptidoglycan molecule as it allows for the establishment of cross-links with neighbouring peptide strands. These cross-links are extremely important for the bacterial cell as they form the basis of the peptidoglycan mesh structure that gives the cell wall its properties [3, 4].

1.2.2 Peptide Cross-Links

As with all other aspects of the cell wall, the cross-links between peptide stems also differ vastly between bacteria. The most common type is by far the 4-3 cross-links, which get their name from the involved amino acids. Here, the amino group of the dibasic amino acid at position 3 (acceptor) is covalently bonded to the carboxyl group of the D-Ala at the forth position (donor) [4]. As this bond occurs via the fourth D-Ala's carboxyl group, the fifth and last D-Ala has to be cleaved out before cross-linking occurs, which causes the donor strand to be reduced down to 4 amino acids [4]. While in Gram-negative bacteria the mDAP and D-Ala amino acids are most commonly directly bonded to each other, in Gram-positive bacteria this link usually occurs indirectly through a pentaglycine bridge between the L-Lys and D-Ala amino acids [3]. The peptide and cross-link variations between Gram-classification bacteria can be seen in Fig. 1.3.

Although by far the most common, the 4-3 cross-links account for 93% of cross-links in *E. coli*. Cross-links between dibasic amino acids, the so called 3-3 cross-links, account for a small percentage of the linkage inside many species [4]. However, in some, such as *M. tuberculosis*, they account for the overwhelming majority of all cross-links [4].

Nonetheless, all this variations do not mean that all peptide stems are cross-linked to each other. The cell wall is a dynamic structure which is constantly remodelled through out the bacteria's life cycle [8]. The degree of cross-linking is usually higher in stationary

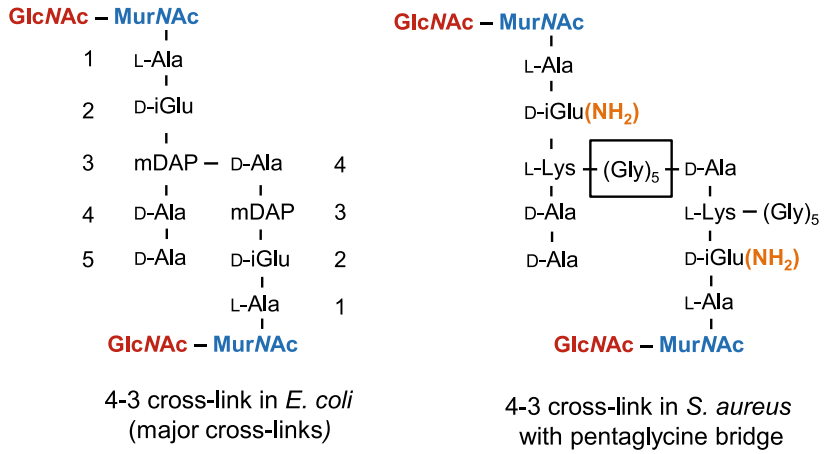

Figure 1.3: **Diagram of the peptides and their cross-links in different Gram-classification bacteria.** In the Gram-negative E. coli (left) cross-links occur directly between the 4th and 3rd amino acids. In the Gram-positive S. aureus (right) cross-links occur indirectly through a pentaglycine bridge. Adapted from Pazos et al. [4].

cells than in exponentially growth cells, which strengthens the cell wall and presumably leads to a stiffer and more resistant cell structure [4]. Furthermore, to adjust for rapid expansion or changes in cell turgor, the peptide cross-links, as well as the glycan strands themselves, are continuously cleaved and reformed to adjust for the cell needs [3, 4]. Even so, as an example, the degree of cross-linking is around 40% in growing *E. coli* cells [3].

Due to both this inconsistent linkage and the aforementioned heterogeneity in the length of the glycan backbones, the cell wall mesh forms pores that allow for the passive transport of nutrients and proteins up to 50kDa. Some studies even suggest larger pores exist that allow for the transport of larger protein complexes across the periplasm [4].

1.2.3 Peptidoglycan Architecture

Despite how fundamental peptidoglycan is for bacterial cells, for a long time it was very difficult to study this molecule due to the complexity and time-consuming nature of the conventional methods used [9] and the difficulties in purifying soluble and homogenous PG fragments for analysis [10]. These issues led to a great number of unknowns in the peptidoglycan architecture. However, overtime, with the improvements in the analytical toolset used to study peptidoglycan structure [9] and, since Meroueh et al. [11] solved the purification problem by determining the total synthesis of a solubilized PG-mimic, a great deal more information about the peptidoglycan architecture has become available.

One of the main debated topics about the peptidoglycan architecture is the disaccharide backbone conformation. The first conformation proposed was a helical 2-fold conformation, where successive PG-stems are rotated 180° relative to previous stem orientation, resulting in a distance between peptide chains with the same orientation (periodicity) of 20 Angstroms (Å) [12]. Once X-ray diffraction analysis of isolated PG was done, this

conformation was quickly replaced by the most commonly accepted one today, a 4-fold helical screw. Here, the PG-stems are rotated 90° relative to the previous stem resulting in a periodicity of 40 Å [10]. When Meroueh et al. [11] synthesised and studied using Nuclear magnetic resonance (NMR) spectroscopy a PG-fragment, they argued in favor of a 3-fold symmetry conformation with a periodicity of 30 Å and a 120° relative rotation. Even so, most *in silico* studies [13] and the most up to date *in situ* studies both argue for a 4-fold symmetry for the PG strand.

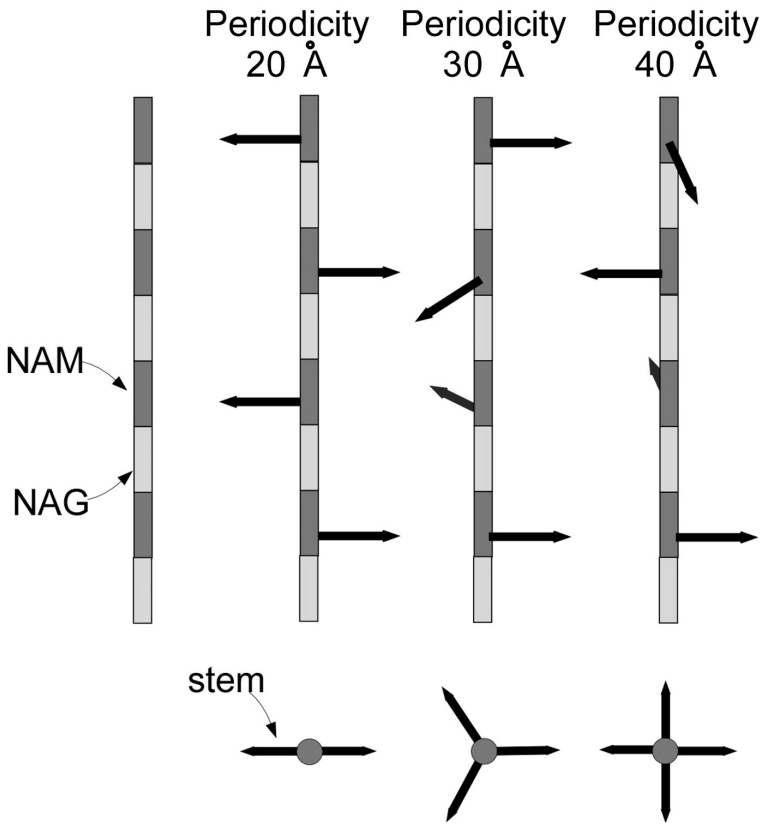

Figure 1.4: **Diagram showcasing all the proposed glycan backbone conformations.** First, a 180° PG-stem rotation was proposed which resulted in a periodicity of 20 Å. Afterwards, the most widely accepted 90° rotation was proposed once more data came forth. Eventually, a periodicity of 30 Å with a 120° relative rotation was also proposed. Adapted from Kim et al. [10].

Furthermore, the orientation of the peptidoglycan chains is also not yet known for certain. Studies into the elasticity of bacterial cells point towards an anisotropic organization of peptidoglycan [4]. Considering volume changes occur mainly along the long axis of a bacteria cell, it has been proposed that the more flexible peptide cross-links run along the long axis of the bacteria cell, whilst the more rigid glycan strands run along the short axis [4].

1.3 Antibacterial Drugs and the Peptidoglycan

Bacterial cell wall's vital role in many of bacteria's processes makes it an ideal target for many antibacterial medications [8]. As an example, the most widely used group of antibiotics, the β-lactam antibiotics, work by specifically inhibiting the formation of cross-links between peptide stems of peptidoglycan molecules. By mimicking the D-Ala-D-Ala end of the peptide stem, which is cleaved during the formation of cross-links, β-lactam can bind to and inhibit one of the proteins responsible for the biosynthesis of the bacterial cell wall, the DD-Transpeptidase [4, 8].

Given its role as a primary drug target, it is easy to see why gathering as much possible information about this molecule is in the interest of the areas of drug discovery and design [14]. Furthermore, as one of the barriers needed to cross for drugs to enter the bacterial cell, data on the mesh is also crucial to understand if medication will reach their target location. As such, computer simulations via molecular dynamic simulations come as a great initial screening process to easily gather data on many potential medications and further determine their adequacy [14, 15].

1.4 Molecular Dynamics

Experimental methods have proved fundamental in pioneering almost all fields of biology and laying the groundwork for many others. However, even though they offer tangible data and ensure physiological relevance, the development and continual refinement of computational methodologies have made *in silico* experiments progressively more accurate, cheaper, and accessible over the years, building up to today's ever-increasing popularity [16].

The study of protein folding, conformational changes, ligand affinity, as well as many other bio-molecular processes, can be substantially improved by the level of atomistic detail offered by molecular dynamics (MD) simulations, which can't be matched by modern experimental methods [16]. Nevertheless, we must always be aware that most MD simulations are but a simplification of reality and only as accurate as the approximations they make. This is a fact that we must always be conscious of when dealing with data generated by these computational methods.

In MD simulations, classical mechanics are used to ascertain how systems of atoms evolve over time. By using an empirically-backed calculation of the force that each atom experiences, regarding its relative position to every other atom in the system, we can solve Newton's laws of motion to predict each atom's path over a small time step [16, 17]. During the time step, movement is assumed to be linear and, once it ends, we can redo the aforementioned calculations at the new particle positions and arrive at new forces that are then used to update the atoms' respective velocities. This process can be done iteratively for a long period of time, and the atoms' coordinates can be saved every N number of time steps, leading to a movie of particle positions evolving over time at the atomic level, called

a trajectory [16, 17]. Notably, as these calculations have to be done for every particle in a system, there is an exponential relationship between the number of particles simulated and the time the simulations take to run.

1.4.1 Molecular Mechanics Methods

Quantum mechanical (QM) models may prove too cumbersome to tackle the size and time scale of the problems we might want to solve with molecular modeling [17]. To overcome this, different methodologies with varying degrees of simplification have been developed over the years.

Compared to QM models, all-atom (AA) models disregard electrons in their calculations, opting for representing atoms as a singular nuclear position with an Van de Waals radius assigned to it [17]. Consequently, all of the energy calculations of a system are performed based only on the nuclear positions of the atoms therein [17]. Undeniably, this does not accurately represent the underlying mechanics behind atom movement, but it gets close enough to reality, such that we are willing to take a hit to accuracy if it means we can simulate longer and bigger systems.

Coarse-grain (CG) models arise from the need to simulate, in a reasonable amount of time, processes that operate at an even larger time and length scales [18]. The peptidoglycan mesh would be an example of such large systems, which are not reasonable to simulate using traditional atomistic methods. The most common CG models for biomolecular simulations try to reduce the overall number of degrees of freedom by grouping 3-to-6 non-hydrogen/heavy atoms into a single representative particle, called CG bead. This particle should attempt to capture the most important and defining features of its grouped atoms by behaving like their average, both chemically and kinetically.

1.4.2 Martini Force Field

The Martini force field is a general-purpose coarse grain force field widely used in biomolecular simulations that canonically uses a 4-to-1 mapping scheme, meaning a single CG bead represents, on average, four heavy atoms and their associated hydrogens [19]. Its versatility allows for the modeling of a wide array of biomolecules, such as sugars [20], proteins [21], and lipids [22, 23], among others [24, 25].

Martini 3 is the third iteration of the Martini force field and has tried to solve some of the main pitfalls of Martini 2, namely its propensity for artificial clustering [26]. Some of these issues resulted from the lack of specific non-bonded parameters for cross interactions between different bead representing moieties of different sizes. This problem was tackled by the introduction of parameters for cross-size interactions and the reparametrization of the same-size ones [26]. Additionally, in Martini 3, water is also defined as its own separate bead type (W) for purposes of non-bonded interactions, as opposed to previous versions [19].

Some of the problems of Martini 2 also arose from its more limited number of bead choices. To ensure we can represent the behavior of all mapped chemical groups with each bead, Martini 3 features four main classes of CG beads, which themselves can come in three different sizes. These bead classes represent the different possible types of chemical groups: nonpolar (C), intermediately polar (N), polar (P) and charged (Q) groups [19]. When using a 4-to-1 mapping scheme, the default Regular (R) bead is used, but when the situation calls for a finer approach, a smaller mapping scheme, such as 3-to-1 or 2-to-1, can be used, leading to the smaller beads sizes of Small (S) and Tiny (T), respectively [19]. However, these mapping schemes are not set in stone, and it is common to undermap groups, such as assigning a S bead size to a 4-to-1 mapping, especially in the case of ring-like and branched chemical groups, something that happens commonly in sugar rings [19], for example.

Furthermore, in Martini 3, the center of mass of the mapped chemical groups is no longer used to define bead placement and is instead replaced by a size-shape approach that tries to conserve, as much as possible, the original solvent accessible surface area (SASA) of the mapped atoms [19]. To do this, the Martini 3 initial suggestion was the use of a Center of Geometry (COG) approach to mapping (with hydrogens included). While this rule-of-thumb approach leads to an overall satisfactory mapping, other mapping strategies that allow for more control of bead placement also exist.

1.5 State of the Art and Goals

In 2014, Gumbart et al. developed a CHARMM36 force-field for the peptidoglycan polymer. They started by parameterizing GlcNAc (NAG) by linking glucose and acetamide, and then MurNAc (NAM) parameters were developed by linking this GlcNAc to lactic acid. When any linking occured, the atoms in the interface between both residues got their charges modified to reflect the changes in the electron environment around the interface atoms [13]. To ensure all involved atoms got assigned the correct charges, *ab initio* quantum chemical calculations were used to determine what the new charges were, as well as all the bonds, angles, and dihedrals involved. For the peptide strand, D-isoglutamate and diaminopimelic acid's parameters were both solved by analogy based on their standard amino-acid counterparts, glutamate and lysine, respectively [13]. To form the complete glycan chain, parameters for NAG-NAM and NAM-NAG links were both set according to the default 1-4 glycosidic bond parameters, equatorial at C1 and equatorial at C4 [13].

While Gumbart's AA model was capable of creating meshes [13], the time and size scale needed to simulate the peptidoglycan mesh is mainly compatible with CG methods, not AA ones. As such, in 2021, Vaiwala et al. developed a CG model based on Gumbart's AA PG force field. This CG model was the first CG parameterization of Gram-negative peptidoglycan for what was, at the time, the latest Martini force field available, Martini 2 [15]. The main goal of their parameterization effort was to create a realistic model capable

of recreating the behavior showcased by the AA trajectories. Apart from the end-to-end distance, both structural properties, such as the equilibrium angle between adjacent peptides, and mechanical properties, like area compressibility, were also targets for the model [15]. Despite accurately reproducing the intended targets, this Martini 2 model initially suffered from aggregation problems common to the force field, which resulted in unrealistic dynamics when compared to its AA counterparts [15]. While this problem was mitigated by uniformly increasing repulsions between all beads, this approach leads to transferability problems across the force field. With the release of Martini 3 in 2021, we hope the improvements to the force field can resolve the aforementioned problems with the previous peptidoglycan model. Even so, given the incompatibility of the parameters between these two versions, Martini peptidoglycan parameters have to be redone [19].

As such, with the help of the previous developers of the Martini 2 PG model [15], our goal for this work was to improve on the previous CG model for the peptidoglycan polymer by taking advantage of the previously mentioned innovations brought forth by the new and improved Martini 3 force field [19].

1.6 Objectives

Despite the simplifications inherent to the CG methodology, a CG model should always strive to preserve as much information from the original molecule as possible. As such, we aimed to accurately replicate the conformational space of peptidoglycan using the coarse-grained model, based on insights from all-atom simulations.

Additionally, we wished to be able to capture both the small-range and long-range behavior of the molecule by matching both the behavior of each individual parameters and the end-to-end distance of our polymers, respectively. Furthermore, this model should also be stable to allow for long lasting simulations. Finally, we also took great effort in trying to replicate the kinetic behavior of the links between monosaccharide units by replicating the rate at which flips between different identified torsional states occurred.

2

Methodological Details

2.1 Empirical Force Fields: Molecular Mechanics

Although a simplification of reality, molecular mechanics (MM) methods try to represent and account for many of the physical properties that govern atom movement with the goal of making simulations as accurate as possible [27]. Therefore, it is important to understand the inner workings of simulations so as to mindfully analyse their results.

2.1.1 Approximations

Out of all approximations made to make all-atom (AA) MM methods work, the Born-Oppenheimer approximation is one of the main ones. Given that, in an atom, the electrons are both much faster and lighter than the nucleus, you may treat their motion as two separate entities. Due to their relatively faster speed, the approximation states that you may neglect the motion of electrons and assume that, as the nucleus moves, the electron cloud instantly adjusts to its movement. What this means to MM is that atoms are now represented as a single particle where the electron cloud and the nucleus are collapsed into one entity assigned with an immutable charge [17, 27].

Coarse-grain (CG) MM methods include further approximations by further reducing the degrees of freedom of their systems. These force fields group atoms into single overall generic beads whose interaction potentials may be the same despite representing distinct molecules/moieties [17]. As a trade-off, energy landscapes get smoothed and larger times steps become available, allowing for much faster simulations.

2.1.2 Force Fields and Potential Energy

Simply put, the force field refers to the equation that determines the potential energy of a system as a function of its internal coordinates [17, 28]. Through the force field, the position of a particle, relative to all others, can tell us the amount of energy contributed by each particle's interactions to the potential energy, which then allows us to calculate the **force** controlling the movement of all atoms in the system [28].

A force field lays out how particles will interact with each other, and, given that the interaction of any two particles can occur by intra- and inter-molecular means, most force fields are split into bonded and non-bonded interactions (Eq: 2.1) [17, 28].

$$U^{system} = U^{bonded} + U^{nonbonded} \tag{2.1}$$

In the case of bonded interactions, these can be further divided into bond stretching, angle bending and dihedral torsions (Eq: 2.2) [29]. Deformations of bond and angle geometry confer energy penalties due to deviations from the set equilibrium point and a function can outline how the energy changes as certain dihedral angles rotate [17, 29]. The parameters that govern these functions are usually obtained via ab initio, semi-empirical quantum mechanical calculations or fittings to experimental data [30]. In CG's case, these parameters usually come from data obtained in simulations using MM methods at finer resolutions.

$$U^{bonded} = U_{bond\ stretching} + U_{angle\ bending} + U_{dihedral\ torsions} \tag{2.2}$$

Additionally, inside non-bonded interactions both van de Waals and electrostatic forces are represented using different functions which dictate energy fluctuations based on the distance and polarity/electric charges (Eq: 2.3).

$$U^{non-bonded} = U_{Van\ der\ Waals} + U_{electrostatic} \tag{2.3}$$

A visual representation of all the mentioned parameters can be seen in Fig. 2.1.

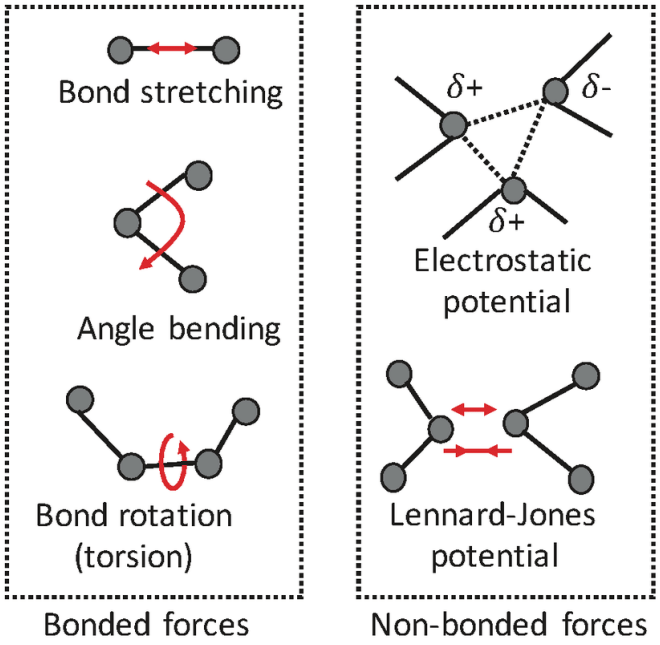

Figure 2.1: **Bonded and Non-bonded forces accounted for in empirical force fields.** Adapted from Waidyasooriya et al. [31].

Overall, force fields are developed for different resolutions and with different goals in mind, and thus should be used accordingly [32]. Some of the most widely used AA general purpose force-fields are the **C**hemistry at **Har**vard **M**acromolecular **M**echanics (CHARMM) [33] and **G**eneral **A**MBER **F**orce **F**ield (GaFF) [34]. The most widely used CG force field for biomolecular simulations is the general purpose Martini force field [19], although others such as SIRAH also exist [18].

2.2 Bonded interactions

2.2.1 Bond stretching

In molecular mechanics, bonds stretching (U_{bonds}) is usually described by an harmonic potential (Eq: 2.4). For bond n, K_{d_n} refers to the force constant of bond n, meaning the energy needed to stretch or compress said bond value (d_n) significantly away from d_{0_n}, which is the equilibrium point for bond n [17, 29].

$$U_{bonds} = \sum_{bonds} \frac{1}{2} K_{d_n} (d_n - d_{0_n})^2 \tag{2.4}$$

Although easier to compute, the harmonic potential does not accurately represent bond stretching. As shown in Fig. 2.2, when compared to the more realistic Morse potential, an harmonic potential can only accurately capture the energy of a system for small deviations from the equilibrium point and does not allow for bond breaking [29]. Nonetheless, it's unmatched computational efficiency makes it the premier choice for many MM force fields.

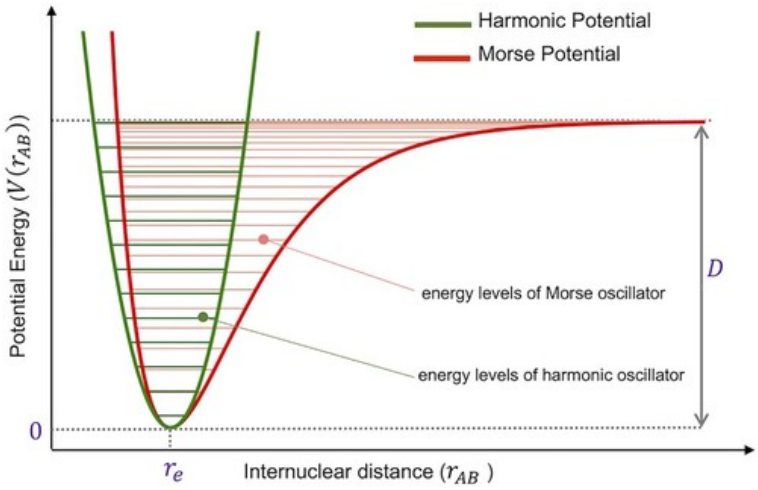

Figure 2.2: **Comparison of the Harmonic and Morse potential.** The harmonic potential does not allow for bond breaking and only accurately captures the energy of a bond for small deviations from the equilibrium point. Adapted from Amir Mirzanejad and Sergey A. Varganov [35].

Although they can be used in angles and dihedrals as well, constraints are most commonly used for bonds. Constraints fix a parameter to a specific value, preventing any fluctuations and effectively eliminating one degree of freedom from the system [36].

2.2.2 Angle bending

Similarly to bond stretching, angle bending (U_{angles}) is also commonly modelled using a harmonic potential (Eq: 2.5) [37]. Once again, K_{θ_n} represents the energy needed to significantly shift away the real angle value, θ_n, from the equilibrium point, θ_{0_n}.

$$U_{angles} = \sum_{angles} \frac{1}{2} K_{\theta_n} (\theta_n - \theta_{0_n})^2 \tag{2.5}$$

Another common way of modelling angle bending is using a cosine harmonic potential (Eq: 2.6) [37]. This potential is the one used by the general-purpose Martini force field [38].

$$U_{angles} = \sum_{angles} \frac{1}{2} K_{\theta_n} (\cos\theta_n - \cos\theta_{0_n})^2 \tag{2.6}$$

In 2013, Bulacu et al. laid out improved angle potentials (Eq: 2.7) which restricted them from reaching a 180° value [37].

$$U_{angles} = \sum_{angles} \frac{1}{2} K_{\theta_n} \frac{(\cos\theta_n - \cos\theta_{0_n})^2}{\sin^2\theta_n} \tag{2.7}$$

These restricted bending (ReB) potentials are useful for angles participating in dihedrals prone to instabilities [37]. Even if the differences between these equations might seem minimal, they have a profound impact on the underlying potentials, as shown in Fig. 2.3. Notably, when reaching either the 0° or 180° value, the cosine harmonic potential weakens, while the restricted angle potential restricts it from ever reaching it [37]. This fact needs to be accounted for when restricting angles of particles over which a dihedral potential is applied, given that these can not reach close to collinearity or the dihedral potential calculation will diverge and, colloquially, crash the simulation [29, 37].

2.2.3 Dihedral Angle Bending

Although all dihedrals refer to the angles formed between two planes, there still exist two types of dihedral potentials: *proper* dihedrals and *improper* ones (Eq: 2.8).

$$U_{dihedral\ torsions} = U_{proper} + U_{improper} \tag{2.8}$$

Given the atoms *i-j-k-l*, connected by bonds in order, the dihedral angle is the angle formed between the planes *ijk* and *jkl*, with the torsion occurring around the bond *j-k*.

The potential energy function of a *proper* dihedral (U_{proper}) creates an energy landscape where the height of the energy barriers is controlled by K_{ϕ_n}, the number of minima by n,

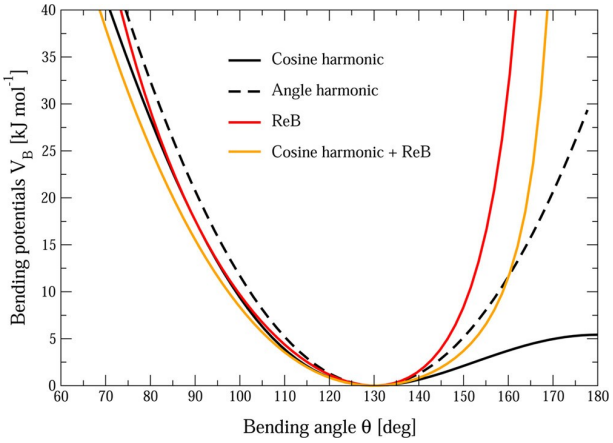

Figure 2.3: **Visual representation of the different angle potentials.** The most common angle potential used in Martini 3, the Cosine harmonic (line), looses strength when reaching the 180° values, which can lead to numerical instabilities in dihedrals. The Restricted Bending potential (Red) never allows an angle to reach close to the 180° mark, thus making it the preferred choice when concerns due to dihedral instabilities arise. Adapted from Bulacu et al. [37].

and the phase factor, δ, determines where the torsion angle passes through its minimum and maximum energy value (Eq: 2.9) [29, 37].

$$U_{proper} = \sum_{proper} K_{\phi_n}[1 + \cos(n\phi_n - \delta)]^2 \tag{2.9}$$

To model more complex behaviors, several dihedral potentials, each with their own torsion constant (K_{ϕ_n}), phase factor (δ) and multiplicity (n), may be summed up to create specific energy landscapes [37].

Similarly to bond stretching and angle bending, *improper* dihedrals angles ($U_{improper}$) are usually modelled using an harmonic potential. This dihedral type is more commonly used to preserve the geometry and chirality of certain groups, such as aromatic rings [29]. The main difference to the *proper* one, is that *improper* dihedrals are not differentiable at 180° from their targets and are thus only used when it is not expected for particles to explore the entire torsional space around the bond *j-k*.

$$U_{improper} = \sum_{improper} \frac{1}{2} K_{\xi_n}(\xi_n - \xi_{0_n})^2 \tag{2.10}$$

In the previous subsection, the instabilities of the dihedral potential calculation were mentioned. These instabilities arise from the calculation of the force $F(\phi_{ijkl})$ that is applied to the atoms of the dihedral. Here, the derivative of the dihedral potential as a function of the atom's coordinates is used to find said force. However, this calculation relies on solving both $\frac{-1}{\sin\theta_{ijk}}$ and $\frac{-1}{\sin\theta_{jkl}}$, and, as the function sine approaches its limit (0° or 180°), the gradient becomes very steep, resulting in unreasonably large forces and consequently unstable integration [37, 39].

2.3 Non-Bonded interactions

When independent molecules or atoms separated by three or more bonds interact in atomistic simulations—two or more for Martini—they do so through non-bonded terms [17, 29]. As will be elaborated on later on in section 2.8, these interactions only occur within a certain radius, the cut-off radius.

In the case of non-bonded interactions, software such as GROMACS [36] implement the strategy of *atom types*. This helps first define the interaction parameters between all *atom types* and then reuse said parameters for all particles of the same *atom types* in the system.

2.3.1 Van der Waals forces

The Van der Waals forces (U_{VdW}) between 2 particles are modelled using a Lennard-Jones potential, also called the 12-6 potential. In this equation (Eq: 2.11), the first term $\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12}$ accounts for the repulsive forces at short distances, while the second term $\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6}$ accounts for the dispersive forces at long distances [40]. Two defining constants exist in this potential, ϵ and σ. By dictating the well depth, ϵ measures how strongly two particles attract each other [40]. σ reflects how close two particles can get before the repulsive term starts dominating [40]. In the equation itself, σ is the distance at which the intermolecular potential is equal to 0. r is the distance between the two interacting molecules [40]. All of these terms can be seen explicitly in Fig. 2.4.

$$U_{VdW} = \sum_{i,j} 4\epsilon_{ij} \left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6} \right] \tag{2.11}$$

2.3.2 Electrostatic forces

The final force field term handles the interaction potential between particles with electric charges. The most common way to represent electrostatic potentials is by using Coulomb potentials to model how charged particles interact [29]. Here, the distance between any two charged particles, i and j, is represented by r_{ij} and the charge of said particles is denoted by q.

$$U_{electrostatic} = \sum_{i,j} \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}} \tag{2.12}$$

2.4 Virtual Sites

Virtual sites (VS) in molecular mechanics correspond to massless particles in a system, which are not moved or accelerated on their own, and, instead, are derived from the positions of other particles in the system. They are commonly employed to remove degrees

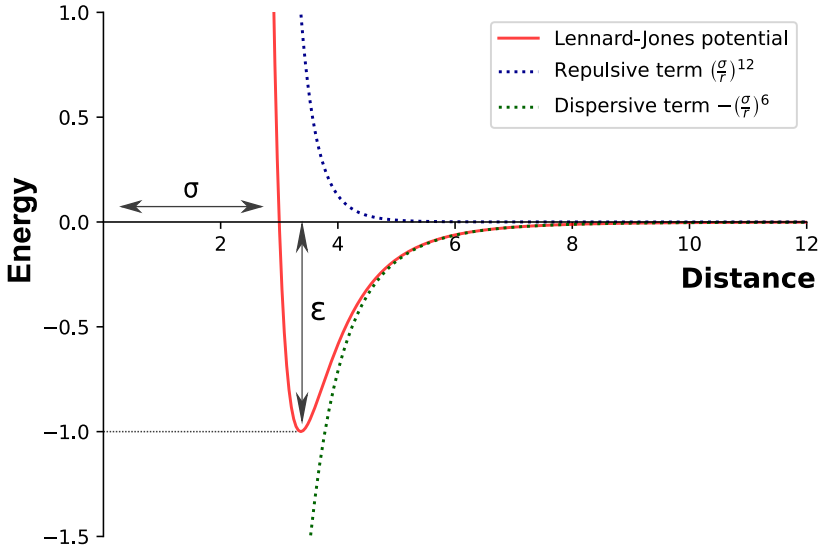

Figure 2.4: **A Lennard-Jones potential with its corresponding terms ($\epsilon = 1; \sigma = 3$).** The dispersive term decreases more rapidly at shorter distances, but eventually, the steep increase of the repulsive term dominates the interaction. The resulting sum creates a potential which more realistically captures how particles interact.

of freedom from the system at higher dimensions than the aforementioned constraints. As an example, these are commonly used as a virtual interaction site at specific locations, such as the center of a benzene or pyranose ring [36].

2.5 The algorithm

The driving factor dictating molecular dynamics (MD) simulations is the force experienced by each particle; however, all of the equations mentioned before only dealt in energy (U), not force (F). The force that particle i experiences as a result of interaction n is given by (minus) the derivative of the potential energy of interaction n as a function of particle i's position ($\mathbf{r}_i$) (Eq: 2.13) [37].

$$F_n^i(\phi) = -\frac{\partial U_n(\phi)}{\partial \mathbf{r}_i} \tag{2.13}$$

For each potential in the system, the forces it imposes are calculated, and each participating particle stores its respective value [36]. Given that a particle can participate in more than one interaction, we sum all of the calculated forces imposed on each particle after all interactions are accounted for. This final sum is the force each particle experiences at time point t.

Knowing the mass of each atom (m_i) and the force (F_i) imposed on it at time point t, Molecular Dynamics simulations integrate Newton's Laws of Motion to determine the particle's acceleration (Eq: 2.14) [17, 41].

$$a_i = \frac{F_i(t)}{m_i} = -\frac{1}{m_i}\frac{\partial U(t)}{\partial \mathbf{r}_i} \tag{2.14}$$

From the acquired acceleration and the previous velocity and particle coordinates, new position and velocities can be obtained for the successive time step $t + \Delta t$ [36, 41].

For as long as the simulation is desired, this procedure can be repeated, starting at some initial state, calculating the forces, determining the new velocities and positions, updating the initial condition, and starting the process again.

2.6 Time steps and integrators

As biological systems are usually composed of more than two particles, there are no analytical solutions to the equations of motion discussed earlier [42]. Instead, a different approach is used, where the integration is divided into small finite time steps (Δt). While arbitrary, choosing the length of a time step should take into account the nature of the issues being dealt with. In the case of molecular dynamics, the fastest motion in the system is the vibration frequency of bonds between atoms, and, given that these occur in the 10^{-14} second (s) range (10 femtoseconds), the Δt typically used is in the order of 10^{-15} s [42]. Thus, the most common Δt used in AA simulations is 0.002 picoseconds (ps), assuming all bonds are swapped for constraints, otherwise, 0.001 ps has to be used. CG simulations use fewer degrees of freedom, heavier atoms and potentials with a smoother energy landscape, consequently allowing for an increased integration time (0.02 ps) [41].

The most commonly used integrator in MD is the *leap-frog* algorithm [36]. The name originates from the positions and velocities "leaping over" each other in time due to updating at alternating instants, separated by half a time step, as highlighted in Fig. 2.5. Essentially, the algorithm stores the positions $\mathbf{r}$ at time t, but the velocities $\mathbf{v}$ at $t - \frac{1}{2}\Delta t$ [36].

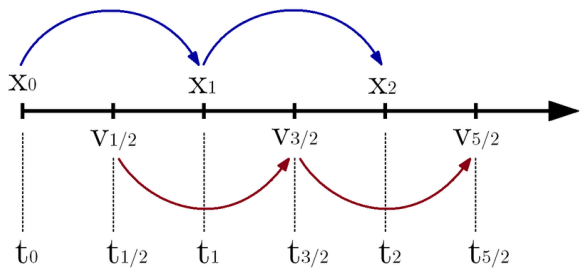

Figure 2.5: **Schematic of the leap-frog integration algorithm.** The velocities and positions are calculated separately at half-time time intervals and using the previous half-time step values. Adapted from T.Gisbert et al. [43].

The velocities at time step $t + \frac{1}{2}\Delta t$ are obtained using the previous half-time step velocities $\mathbf{v}\left(t - \frac{1}{2}\Delta t\right)$ and the acceleration at the current time step $\frac{\mathbf{F}(t)}{m}$ (Eq: 2.15) [36].

$$\mathbf{v}\left(t + \frac{1}{2}\Delta t\right) = \mathbf{v}\left(t - \frac{1}{2}\Delta t\right) + \frac{\mathbf{F}(t)}{m}\Delta t \tag{2.15}$$

Then, the positions at the next time step $\mathbf{r}(t + \Delta t)$ are based on the current position $\mathbf{r}(t)$ and the previously calculated velocities $t + \frac{1}{2}\Delta t$ (Eq: 2.16) [36]. During each time step interval, it is assumed that the applied forces are constant and each particle's motion is linear.

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \mathbf{v}\left(t + \frac{1}{2}\Delta t\right) \tag{2.16}$$

Stochastic dynamics is another integration algorithm that adds a random friction and noise term to the calculated velocities, which implicitly models the energy exchange with the rest of the system. As a result, the system explores the conformational environment in great detail while still maintaining proportionality [44]. This algorithm enables a more thorough search of the conformational space while reproducing the proportion of time spent in each state, which facilitates the parameterization process of CG molecules. Even so, we must be aware that this upside comes at the cost of unrealistic kinetics [36, 44].

2.7 Simulation Workflow

Before we are able to run long simulations that produce the data we wish to analyse, we need to slowly and gradually prepare the system for it. The first step in any system preparation involves obtaining the structure of the biomolecule(s) of interest. For AA methods, this can be done through experimental techniques, such as X-ray Crystallography [45] and Nuclear Magnetic Resonance (NMR) [46], or computational ones, such as structure prediction software [47]. For CG methods, the simplification of an AA model is usually used to obtain a initial structure, although prediction software also exists [48].

In the case of biological systems, the particles should be solvated with water and their charges should be nullified with counter-ions. The behavior of a molecule in vacuum is completely different from the behavior in a biological solution, and, therefore, we should always try to resemble as closely as possible the physiological environment of the particles of interest.

2.7.1 Energy Minimization

The conformations in which the molecule of interest exists before simulation are not necessarily the energetically favourable ones. The aim, then, is to iteratively adjust the atom's coordinates to avoid any steric clashes or inappropriate geometries. These conformations artificially raise the energy of the system and lead to extremely high forces, which in turn causes the system to be unstable [17].

Given that we can write the system's potential energy as a function of the positions of all its atoms, we may imagine a hypersurface where each point corresponds to one possible configuration of the atom positions and the value at that point is the corresponding potential energy of that conformation [41]. This surface can be imagined in a simplified way as the one in Fig. 2.6.

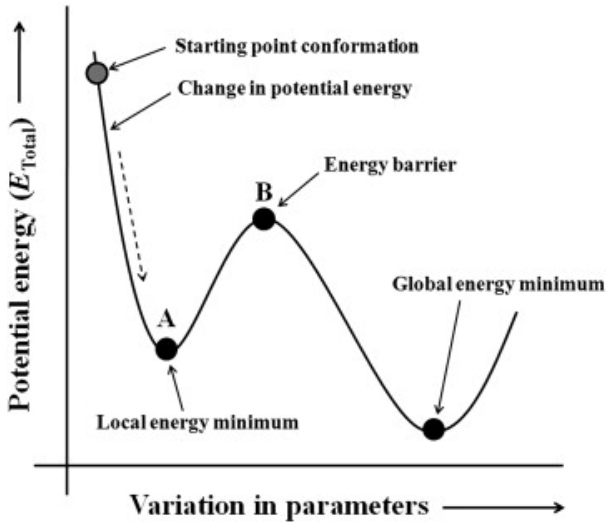

Figure 2.6: **Energy minimization process of a molecule using the Steepest Descent algorithm.** Depending on the starting point conformation, it is possible that the algorithm gets stuck in a local minima, instead of finding the global one. Adapted from Roy et al. [41].

The *Steepest Descent* algorithm "slides" down this energy surface towards the configuration that corresponds to the surface's local minimum. For each atom, it computes the forces acting on it and then updates its positions by moving them in the direction of said force. This ensures the atoms move in the direction of the steepest descent on the energy surface towards more energy favourable configurations [17, 36]. This process is iteratively repeated until a maximum number of steps have been performed or the energy changes between each step are deemed no longer significant. While efficient, this algorithm cannot guarantee that at the end of the iteration, we reached the conformation corresponding to the global energy minimum of the system, also shown in Fig. 2.6 [41].

2.7.2 Equilibration

The minimization process only alters the initial coordinates of our system. There is no concept of time in minimization, and therefore particles do not have velocities or momentum nor are they accelerated. As such, the minimization step does not tell us the initial velocities for our simulations [36]. Instead, these are randomly assigned at the start of the equilibration stage from a Maxwell-Boltzmann distribution at the desired temperature.

Before we can start producing data, we need to ensure the system is in such a state where its thermodynamic properties (temperature, volume and pressure) are brought to the defined target values and fluctuate persistently around those. We do this by using thermostats and barostats to control the temperature and pressure, respectively [36]. The use of thermostats and barostats stabilize simulation conditions and ensure us we are simulating at the temperature and pressure we say we are.

In AA simulations, the first equilibration step is usually a short NVT simulations where the Volume (V) and Temperature (T) are kept constant and the pressure (P) is allowed to fluctuate [36, 42]. Followed by a NPT simulation where the Pressure (P) and Temperature (T) are kept constant, while the Volume (V) adjusts [36, 42]. Due to the robustness of CG systems, the first equilibration simulation (NVT) is usually skipped.

At the end of the equilibration step, we will have a NPT ensemble resembling real world conditions (constant pressure (P) and temperature (T)), which is ready to generate data for analysis [17, 36].

2.7.3 Temperature

Thermostats regulate the average temperature in a simulation by adjusting the velocities of the particles [36, 42]. The two thermostats used in our work were the Velocity-rescaling and Nose-Hoover thermostats.

At every step, the Velocity-rescaling method simply multiplies the velocity of every particle by a scaling factor that is based on the current temperature and the desired temperature. If the current temperature is warmer than the desired one, then the velocities get scaled down; if the opposite is true, then they get scaled up. The larger the difference between the temperatures, the larger the scaling is [36]. Additionally, a stochastic term is also added to ensure a true canonical ensemble is being sampled by introducing realistic energy and temperature variations.

In the Nose-Hoover thermostat, the entire system gets coupled to an external thermal reservoir with its own mass, potential and kinetic energy. This is incorporated into the equations of motion themselves, which now have an additional term slowing or speeding up the particles so as to control the system's velocities [36].

2.7.4 Pressure

Barostats control the pressure of a system by adjusting the volume of the simulation box throughout the run [36, 42]. Here, the Berendsen and Parrinello-Rahman barostats were used to control the pressure in our simulations.

Similarly to how the Velocity-rescaling method rescaled velocities to control temperature, the Berendsen barostats rescales the volume of the simulation box to control the pressure. At every step, the Berendsen algorithm scales the atom's positions and box vectors by a matrix, which gradually moves the pressure towards the desired pressure value. Although this barostat yields the correct average pressure over simulation time, it

does not recreate a true NPT ensemble. Therefore, it is most commonly used during the equilibration step of CG systems [36].

The Parrinello-Rahman barostat works similarly to the Nose-Hoover thermostat, in that it also supplements the equations of motion for the atoms to increase or decrease pressure as needed. Crucially, the box vectors themselves also start to respect the equations of motion [36].

2.8 Periodic Boundary Conditions

Molecular Dynamic simulations apply Periodic Boundary Conditions (PBC) to deal with surface effects inherent to finite systems. This entails placing the simulated particles inside a finite box and then surrounding said box with translated copies of itself [36]. Without PBC, the atoms near the boundaries of the simulation box would experience a different environment to those at the center; with PBC, the atoms at the boundaries interact with the images of the adjacent translated copy, thus mimicking an "infinite" system [36].

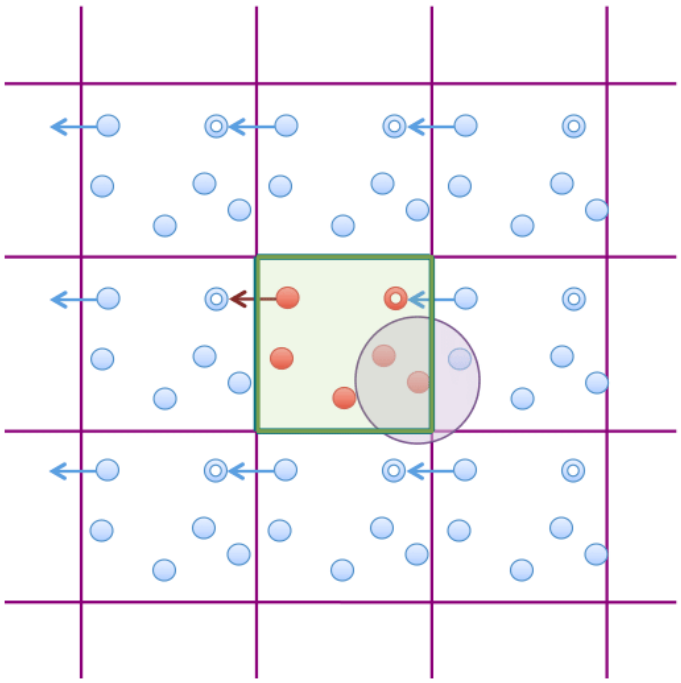

Figure 2.7: **Two-dimensional illustration of Periodic Boundary Conditions and cut-off radius.** The system is surrounded with identical copies of itself and particles only interact with other particles that are within a defined radius. Adapted from Garcia et al. [49].

PBC allows us to approximate the behavior of very large infinite systems, while simulating only a finite and comparatively small number of particles. However, it introduces other problems, as we need to make sure atoms don't interact with several copies of another atom or even with themselves. To solve this, packages such as GROMACS implement methodologies like minimum image convention.

Minimum image convention makes sure that each particle only interacts once with any other particle by ensuring we always use the closest image of each particle to calculate interactions [36].

Another methodology applied by GROMACS is the cut-off radius. The cut-off radius refers to the distance used to determine which particles can "feel" which [36]. As shown in Fig. 2.7, only particles inside the circle defined by the radius are able to establish non-bonded interactions with each other. As the strength of these interactions fade away with distance, the non-bonded terms between two particles at long enough distance are not significant enough to be accounted for. Thus, this method greatly speeds up simulation time by limiting the number of calculations that need to be performed. As an example, Martini 3 uses a 1.1 nm cut-off radius, which means that two particles that are further away then 1.1 nm do not establish non-bonded interactions [19].

Additionally, when defining the length of the box vectors, we should always take into account the size of the macromolecule being simulated. Neither should two ends of a macromolecule be able to interact with each other, nor should a solvent molecule interact with both ends of the macromolecule at the same time [36]. To this end, the length of each box vector must be larger than the length of the macromolecule plus two times the force fields cut-off radius [36].

2.9 Metadynamics

Despite all its optimizations and simplifications, all molecular dynamics simulations are still limited in their ability to sample the entire conformational space of their molecules of interest [50]. Looking back at Fig. 2.6, we can imagine a scenario where the energy barrier **B** is large enough that only in an unreasonably long simulation would our molecule surpass the energy barrier and start sampling the global minimum. Effectively, due to the size of the surrounding energy barriers, our molecule is trapped in the starting configuration's local minima, resulting in a poor sampling of its actual underlying behavior. One way to solve this problem is using Metadynamics.

Metadynamics is an enhanced sampling technique which accelerates the kinetics of transitions by continually imposing potential on a few selected degrees of freedom, the so-called collective variables (CVs) [51]. At predetermined intervals, this method regularly adds Gaussian functions to the system's potentials controlling the selected CVs.

These bias potentials are not deposited randomly; instead, they are deposited to the potentials of the system where the CV's variables stagnate, thus forcing the system to leave the sampled configurations and explore new ones [51]. As highlighted in Fig. 2.8, this procedure continually fills up the basins of the CVs' potential energy surface and allows the systems to evolve from the initial conformational state, in an order that follows the height of the energy barriers [52]. For a long enough simulation, the shape of the added Gaussians potentials tend towards that of the underlying potential energy surface, also shown in Fig. 2.8.

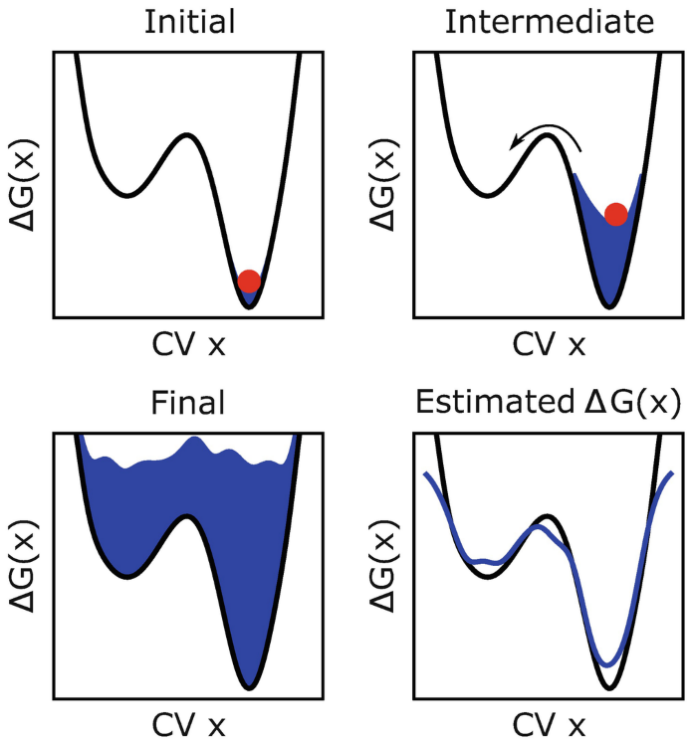

Figure 2.8: **Build up of the sum of Gaussian potentials in a metadynamics simulations across time.** When the simulation ends, the sum of deposited Gaussians allows for the reconstruction of the underlying energy landscape. Adapted from Clayton et al. [53].

2.10 All-Atom to Coarse-Grain Mapping

As previously mentioned, to accurately recreate the non-bonded interactions between molecules, Martini 3 suggests the use of a Center of Geometry (COG) approach to mapping (with hydrogens included) [19]. Additionally, for sugars specifically, Grünewald et al. [20] proposes a 15% increase in the ring's bead distance for a better match in SASA values. The main goal surrounding these methods is to propose general rules-of-thumb that lead to a good match between AA and CG SASA values. As we wished to precisely fine tune the positions of the beads, to lead to the best possible overlap between AA and CG SASAs, we used a Center of Weights (COW) approach. Here, we are able to finely place the position of the beads by assigning different weights to different atoms. For simplicity, only integer values were used for the weights.

GROMACS was used to compute the SASA values for our models. The Van de Waals radii used for SASA calculations followed the standard Martini 3 protocols: 0.264 nm for R beads, 0.230 nm for S beads and 0.191 nm for T beads. The probe used had a radius of 0.191 nm. 4800 dots per sphere were used when creating data for analysis and 240 per sphere were used for visualization purposes.

2.11 Coarse-Grain Parameterization

Parameterization refers to the iterative process of figuring out what combination of parameters for our CG representation best reproduces the target behavior of the system. This is done by imposing and continually tweaking bonds, angles, dihedrals, and other parameters, such that the model eventually matches the target behavior [54]. There are two approaches to CG parameterization. Bottom-up approaches focus on fitting the model to higher-resolution atomistic simulations, to the extent the model will be able to reproduce the target's structural distributions on its own [55]. In top-down approaches, the goal is mainly to reproduce chemical properties by fitting the model to experimental data of interest [55]. In the case of Martini 3, both strategies can be employed.

The steps taken in the parameterization process of a new molecule in the Martini 3 force field are shown in Fig. 2.9 [54]. First, we need to obtain an atomistic structure and topology of our molecule of interest. We use these to run a long AA simulation which explores the conformational space of our molecule of interest.

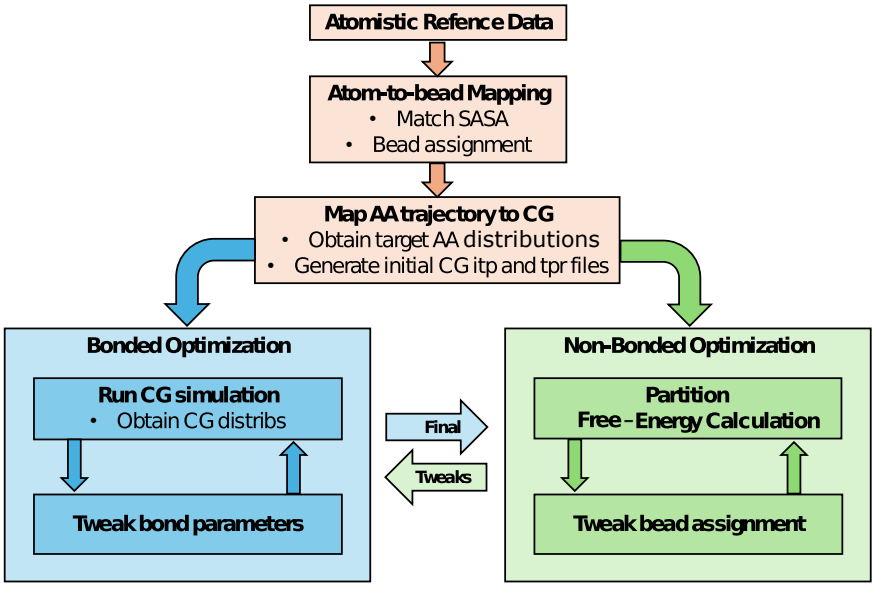

Figure 2.9: **Typical parameterization workflow for the Martini 3 force field.** The bonded and non-bonded optimizations are performed hand-in-hand and continually influence each other during the parameterization process.

Then, based on the structure of the molecule (functional groups, side chains, and more), we start grouping several atoms into single beads. Mapping a molecule should strike a balance between simplification and preserving as much physical and chemical properties as possible [54]. One of the target features to preserve is the Solvent Accessible Surface Area (SASA), due to being one of the main physical properties controlling how the molecules interact with their solvent and/or other molecules. At this point, beads are also assigned their chemical type based on the atoms they represent, usually by following guidelines made for that type of biomolecule [54, 56].

With the mapping established, we can now convert the obtained AA data to the CG resolution. This allows us to create all the bonds, angles and dihedrals distributions that are used to tweak our CG model against [54]. Additionally, these distributions give us an initial guess into the parameters needed to recreate AA behavior, consequently allowing us to start running our first CG simulations [54].

The optimization process can be split into two parts. The bonded optimization involves iteratively checking the distributions obtained from CG simulations, tweaking the parameters to better account for any discrepancies to the AA distributions, and running new simulations to obtain new CG distributions [54]. This process is repeated until CG simulations match AA ones to a satisfactory degree. The non-bonded optimization entails comparing the experimental partition free energies to the one our model obtains using computational free energies techniques and then adjusting the bead assignment to better match the two values [54, 56].

Apart from the non-bonded optimization, all of these parameterization steps and targets are applicable to our parameterization effort of peptidoglycan. Even so, other targets which are not laid out here can also be relevant depending on the parameterization goals.

2.12 Simulation Conditions

The GROMACS simulation package was used for both AA and CG standard simulations [36, 57]. For Metadynamics simulations, we used GROMACS's plugin PLUMED [58] and followed the instructions available in the virtual "PLUMED Masterclass" [59]. Many of the topologies used in this work were created using PolyPly [48] and CGsmiles [60]. The versions of these software can be checked in Table 2.1.

Table 2.1: **List of all software used for preparing and running Molecular Dynamics simulations.**

Software	Version
GROMACS	2021.7
PLUMED	2.8.3
Polyply	1.6.1
CGsmiles	—

2.12.1 All-Atom Simulations

The glycan and peptidoglycan AA force field was developed by Gumbart et al. [13] using the CHARMM36 force field, as previously detailed in section 1.5. All structures and topologies were originally developed by Vaiwala et al. [15]. The TIP3P water molecule model was used for all solvations. The structures and topologies for the disaccharide units NAG-NAM and NAM-NAG were obtained by adapting the AA glycan structure and topology. For the NAG-NAM structure, we capped it with the last disaccharide unit methyl

group by aligning neighboring carbons in the pyranose ring and then transferring the functional group. The NAM-NAG structure was already adapted with the methyl-capping.

As the first step in any CG parameterization process relies on all-atom reference data, several AA simulations were done. These can be consulted in Table 2.2.

Table 2.2: **The properties of all the all-atom simulations ran to gather reference data.** The molecule in solution, the length of said molecule in number of disaccharide units, the duration of the replicates and the number of said replicates are all shown.

Simulation Type	Number of Disaccharide Units	Time (ns)	Number of Replicas
Glycan	8	400	1
Peptidoglycan	8	3500	2
NAG-NAM	1	1500	4

For the 8-mer glycan simulation, the molecule was placed in a ~10x10x10 nm^3 square box and solvated with ~32000 water molecules. The peptidoglycan strand was placed in a bigger ~12x12x12 nm^3 square box with ~56000 water molecules and neutralized with 16 potassium ions (+1 charge). For the disaccharide simulation, a 4 nm vector length dodecahedral box was used to place NAG-NAM and further solvated with ~1400 water molecules.

For all atomistic simulations, the standard minimization and equilibration steps were taken. First, minimization was carried out using the steepest algorithm. Then, the Nose-Hoover thermostat with a coupling time of 1 ps was used to fix the temperature at 310K during the first equilibration step (NVT) [15]. Afterwards, to bring about a isothermic-isobaric ensemble (NPT) at 1 bar, the system underwent a second equilibration step additionally using the Parrinello-Rahman barostat with a coupling time of 5 ps. The production runs utilized the same thermostat and barostat algorithms and settings. For all simulations a time step of 0.002 ps was used.

In all simulations, the Verlet cutoff scheme was used to compute the non-bonded interactions with a max cut-off distance of 1.2 nm [15]. Both, long range electrostatic and VdW forces were handled using the Smooth Particle Mesh Ewald (SPME) summation [36] to ensure a smooth continuous transition in between the 1.0 nm and 1.2 nm range [15].

2.12.2 Metadynamics All-Atom Simulations

Both disaccharide links where explored using metadynamics. Similarly to the standard all-atom simulations, these were placed in a 4 nm vector length dodecahedral box and solvated with ~1400 water molecules. Additional properties can be consulted in Table 2.3.

Minimization and equilibration steps similar to the ones previously discussed in section 2.12.1 were conducted to ensure stability and accuracy in these simulations. The glycosidic link torsion angles ϕ and ψ were the chosen collective variables explored in the metadynamics simulation. We used the C+1 crystallographic convention to define the torsion angles, making the definition of ϕ as $O_5 - C_1 - O_4 - C_4$ and ψ as $C_1 - O_4 - C_4 - C_5$.

Table 2.3: **The properties of all the metadynamics simulations ran to gather reference data.** The molecule in solution, the length of said molecule in number of disaccharide units, the duration of the replicates and the number of said replicates are all shown. For the collective variables, the ϕ and ψ angles follow the C+1 crystallographic convention(.. – $C_x - C_{x+1}$) [61].

Simulation Type	Number of Disaccharide Units	Time (ns)	Number of Replicas	Collective Variables
NAG-NAM	1	300	4	ϕ; ψ
NAM-NAG	1	300	4	ϕ; ψ

Gaussian potentials with a width of 0.20 rad and height of 1.20 kJ/mol were deposited every 50000 steps. The system temperature was set to 310K.

2.12.3 Coarse-Grain Simulations

CG simulations were performed using structures obtained by mapping the corresponding AA models. The topologies were developed during the parameterization process of the peptidoglycan while following the official guidelines for carbohydrates laid out by Grünewald et al. [20]. Martini 3 standard water model was used for all solvation. The final simulations can be consulted in Table 2.4.

Table 2.4: **The properties of all the coarse-atom simulations ran to gather reference data.** The model in solution, the length of said model in number of disaccharide units, the duration of the replicates and the number of said replicates are all shown.

Simulation Type	Number of Disaccharide Units	Time (ns)	Number of Replicas
Glycan	8	3500	2
Peptidoglycan	8	4000	2
NAG-NAM	1	10000	4

The size of the simulations boxes remains the same as their mapped counterparts. While the minimization step for the CG simulations were similar to the AA ones, the equilibration step differs. Only one equilibration step was done to ready the system up for production. Here, we used the Velocity-rescale thermostat with a coupling time of 1 ps to keep the temperature at 310K and the Berendsen barostat with a relaxation time of 4 ps to stabilize the pressure at 1 bar. During the production run, the barostat was swapped for the Parrinello-Rahman barostat with a coupling time of 12 ps. For all simulations a time step of 0.02 ps was used.

While the standard leap-frog integrator was used for all final simulations, the Stochastic Dynamics integrator was used during the glycan and peptidoglycan initial parameterization runs to facilitate sampling of bond, angle and dihedral behavior.

2.12.4 Input File Generation

In the GROMACS software, the ITP file is one of the input files needed to run molecular dynamics simulations. It is part of the system topology definition and contains information related to the atoms, bonds, angles, dihedrals and other possible interactions existing within a molecule. Across the parameterization effort, when a CG model is tweaked, the ITP file is the main input file that is being altered.

The parameterization process of a molecule entails continually tweaking the input files for production runs of a model so as to test the new changes and ascertain if they got us any closer to meeting the goals set for our model.

Polyply [48] is a Python package capable of easily generating input files for simulating biomolecules, especially suited for polymers and polysaccharides, such as the case of peptidoglycan. Polyply allows us to easily change the parameters of a single template monomer or link and, from there, build an entire ITP file where this change is applied to every repeat of said template.

The second piece of the puzzle comes in the form of another Python package, CGsmiles. CGsmiles allows for the creation of a graph from just a sequence of our connected monomer names. This graph lays out the structure of our polysaccharide by detailing which residues are connected to which and in which order.

Leveraging these two programs in combination allows us to easily create input files for biomolecules of any size and structure while continually changing parameters between parameterization runs.

2.13 Analysis of MD Simulations

All analysis were conducted using in-house scripts developed with Python 3 and its MDAnalysis [62] library, along side the numpy [63], SciPy [64] and matplotlib [65] libraries. The visualization program Visual Molecular Dynamics (VMD) was used for all simulation visualization and rendering of simulation frames. The versions of all these software can be consulted in Table 2.5.

Table 2.5: **List of all software and libraries used for analysing all molecular dynamics simulations.**

Software	Version
Python 3	3.10.12
MDAnalysis	2.7.0
Numpy	1.26.4
SciPy	1.14.0
Matplotlib	3.9.0
VMD	1.9.4a57

2.13.1 Bead Assignment

After mapping, beads were assigned their type according to the chemical profile of the represented atoms. Three main sources were used to guide the bead type assignment: Souza's "Martini 3: a general purpose force field for coarse-grained molecular dynamics" [19], Grünewald's "Martini 3 Coarse-Grained Force Field for Carbohydrates" [20] nad Alessandri's "Martini 3 Coarse-Grained Force Field: Small Molecules" [24].

2.13.2 Dihedral Sculpting

As previously discussed, the energy of a dihedrals torsion is described by a function which relates the dihedral angle to an energy value. Additionally, as we may add different potential together, we can construct complex energy surfaces that precisely match our distributions to better recreate the targets' behavior. Given this, we may create an algorithm capable of precisely matching the target distribution by "sculpting" its underlying function with a series of additive potentials.

We start out by having three initial inputs: a target distribution, a starting potential (which can be the sum of potentials with different multiplicities), and a scoring function capable of comparing distributions (in this case, a simple sum of squares function). Then, we use input all of these to SciPy's $minimize$ function. Here, the goal is reach the forces and angles for the dihedral potentials that when summed, create a final potential that minimizes the difference between its underlying distribution and the target distributions. Crucially, this process works because we are capable of converting potentials to probability distributions by utilizing the Boltzmann factor.

2.14 Model Evaluation

2.14.1 Obtaining Distributions

The target AA distributions were obtained after mapping the entire AA trajectory to CG resolution. Due to the number of repeats in the peptidoglycan molecule, the script we used accounted for all repeats values when creating the CG distributions. If we were to only look at the average value of all repeats in a frame, the less probable states would get swept under the mean, resulting in all plots looking unimodal. Instead, our script first transforms every repeat to its own individual histogram and only then performs averages on every single bin. This allows us to consider all repeats, whilst still visualizing the less populated states in our distributions. When plotting the histograms, each individual histogram is shown along side the average one.

2.14.2 Evaluation of Link Kinetics

Two relevant torsional states were defined during the parameterization effort of the glycan strand. These two states were defined by their respective ψ value, the "linear" state corresponds to a negative ψ value and the "kinked" state correlates with a positive ψ value. The transitions between these two states was named "flips". We made it a goal to be able to replicate the kinetics of the links between monosaccharides by matching the CG flip rate betweens these two states to the AA one.

To be able to evaluate the flip kinetics, we developed a script which counts how many long-lasting transitions occur between these two states by analysing the trajectory. First, to ensure we are only accounting for long-lasting transitions, we perform a rolling window average to smooth out all the values. Then, the script checks the smoothed ψ value to determine if at any point the predefined state limits were crossed. If any occurred, a flip is counted.

Finally, the Flip rate (Flips/DU/µs) was calculated by simply dividing the number of Flips by the length of the glycan chain in disaccharide units (DU) and by the length of the simulation (µs). The time in each state was calculated by looking at how many frames the dihedral value for the mapped equivalent of ψ was between a certain range. The -20° to 160° range defined the kinked state, while the -180° to -20°, defined the linear state. The 160° to 180° defined a rare saddle point which was too insignificant to consider.

As dihedrals are periodic, depending on the direction of rotation, two different paths may be taken to transition from one state to the other, the 0° and the 180° path. Here, a flip was said to take the 0° path, if the value 0° was crossed during the transitions period between both states. The same reasoning is applied to the 180° path. As such, a flip path was assigned to a flip based on the torsion value of the previous frame once said flip was detected. If the dihedral angle value, clipped to the -180° to 180°, was above 160°, then the 180° path was taken, otherwise, the 0° was used.

2.14.3 End-to-End Distance

For both the glycan and peptidoglycan chains, the end-to-end distance of the polymers was described using only the glycan chain. Specifically, the end-to-end distance is calculated as the distance between the terminal beads of the first and last monosaccharides in the glycan chain.

3

Results and Discussion

3.1 Collaboration and Division of labor

This project was developed in collaboration with the authors of the first coarse-grain (CG) parameterization of the peptidoglycan molecule [15], Rakesh Vaiwala and K. Ganapathy Ayappa at the Department of Chemical Engineering of the Indian Institute of Science of Bangalore. The parameterization process of the glycan strand of peptidoglycan was carried out by myself and Dr. Manuel N. Melo and took place at ITQB NOVA's "Multiscale Modeling Lab", while the peptide portion will be done by Rakesh Vaiwala and K. Ganapathy Ayappa.

Some of the AA simulations used in this parameterization process were the same ones used in the previous parameterization process for Martini 2 and were thus provided by the Ayappa group.

It should be noted that all results are being presented in chronological order following the parameterization workflow. This allowed showing and discussing the problems encountered, as they came up during the parameterization process.

3.2 All-Atom Glycan Chain Simulations

In the steps of Martini 2's parameterization effort, the glycan chain was simulated and parameterized independently of peptidoglycan's peptide. After the behavior of the glycan chain was correctly captured, the peptide chain would be transferred onto the glycan model and adjustments would be made to accommodate for any discrepancies.

As such, we started the parameterization process by first obtaining some reference AA simulations containing only the glycan chain. It was in these AA glycan simulations where we first noticed that the glycan chain of peptidoglycan was not as linear as it is usually depicted. Instead, the chain had the ability to bend, and remain bent, for long periods of time due to the link between monosaccharides introducing a ~90° bend in the glycan backbone structure, as can be seen in Fig. 3.1.

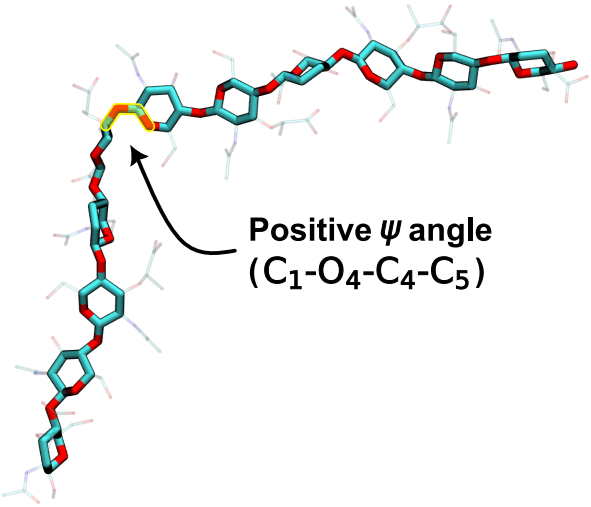

Figure 3.1: **VMD rendering of a simulation frame showcasing the influence of the kinked state (yellow) in the glycan backbone structure.** The kinked state has a very clear influence on the molecule structure by bending the glycan backbone. Both amide and lactate functional groups are greyed out for visual clarity.

We investigated this phenomenon further by plotting the ϕ and ψ angles (C+1) of all glycosidic bonds. As shown in Fig. 3.2, we can clearly see that for both NAG-NAM and NAM-NAG links, there exists two ψ minima, a positive and negative one. On the contrary, the ϕ angles do not seem to have the same bimodal behavior and are instead restricted to the -120° to -30° range. For both disaccharides, we were able to identify the ψ positive conformation as the one responsible for causing this ~90° bend in the glycan backbone. The ψ negative one simply keep the structure linear as it is usually envisioned and can be seen either half of the glycan in Fig. 3.1. Given the influence each state has on the glycan backbone, we named them the "kinked" and the "linear" conformations, respectively. The transitions between the two states was named a "flip".

Although similar, the two disaccharides have different behaviors. Where as both the positive and negative ψ minima in NAG-NAM had around the same frequency, in NAM-NAG, the ψ dihedral is much more skewed towards the negative values. Furthermore, while in the NAG-NAM disaccharide, the positive ψ region has two clear minima, NAM-NAG only has one well defined minimum.

3.3 Mapping the Glycan Chain of Peptidoglycan

Mapping the AA glycan molecule to CG resolution followed the guidelines outlined by Grünewald et al. [20]. First, we followed the easiest rule: keeping functional groups together. This rule was applied to both NAG and NAM's amide group (ACN) and NAM's lactic acid group (LAC). Then, starting from C1, diol groups were grouped together following the increasing carbon number. Following these rules, the mapping showcased in Fig. 3.3 was created.

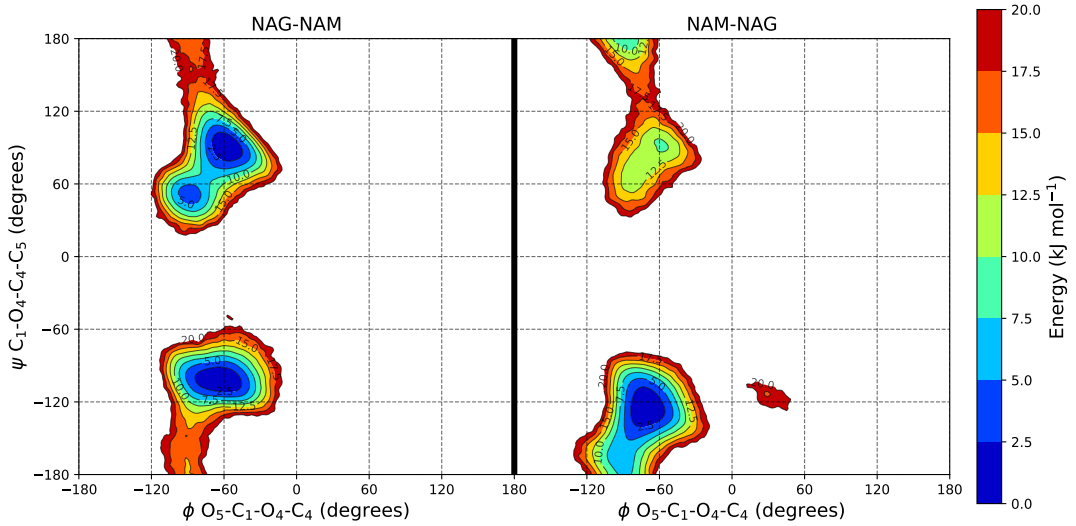

Figure 3.2: **Contour plots for NAG-NAM (left) and NAM-NAG (right) showcasing the changes in the ϕ and ψ torsions (C+1) across the all-atom glycan simulation.** In both disaccharide units two ψ minima were detected across the simulations, one positive and one negative. These minima had a profound influence on the structure of the glycan backbone with the positive ψ values introducing a clear bend, in the otherwise linear glycan structure.

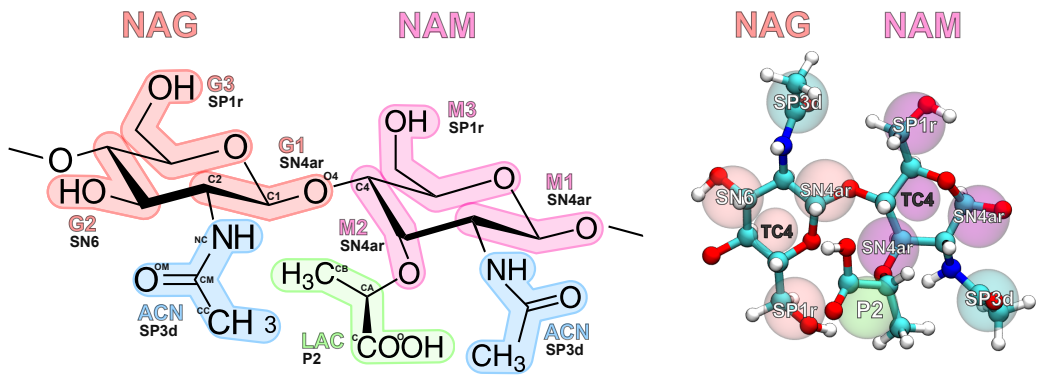

Figure 3.3: **Mapping of all the beads composing the coarse-grain glycan model.** Functional groups were kept together and diol groups were grouped along the increasing carbon index. Besides each bead type, the left scheme also shows which atoms are mapped to each bead and their relative position to the AA model is shown on the right. The virtual site TC4 is omitted on the left for clarity.

When dealing with disaccharides, Grünewald et al. [20] mentions that the bridging oxygen atom should be assigned to the residue with the highest connecting carbon. However, due to the amide functional group replacing a diol group in our C2 carbon, if we were to follow that rule, the G1 and M1 would only be assigned 2 heavy atoms. This alternate mapping is shown in Fig. 3.4 and causes a single T bead to emerge in the pyranose rings. While this isn't necessarily problematic, we still wished to create an uniform pyranose ring composed of only S beads. Therefore, the bridging O was assigned to the carbon of lesser index (C1). In the original guidelines we can see the same principle being applied to the Martini model for cellulose [20, 48].

Although omitted in Fig. 3.3 leftmost picture, our mapping contains a Virtual Site (VS) in the center of all pyranose rings. This VS is used as a massless interaction site which helps replicate the non-polar interactions of the carbon atoms composing the pyranose group [20]. Furthermore, it also helps introduce some of the hydrophobic nature of the pyranose ring and helps counteracts the imbalance between the number of represented atoms and beads [20].

As previously mentioned, in Martini 3, the different bead sizes R, S and T represent 4, 3 and 2 heavy atoms, respectively. As such, including the VS, our model represents the 14-heavy atom NAG with 4 S beads (4*3 = 12 total atoms) and 1 T beads (12 + 2 = 14 total atoms), which results in the exact same number of real to mapped atoms. On the other hand, a 19-heavy atom NAM falls 1 atom short by our representation of 1 R beads (4 total atoms), 4 S beads (4 + 4*3 = 16 total atoms) and 1 T beads (16 + 2 = 18 total atoms). We did not find this problematic, given that a undermapping of less than 10% is considered acceptable and, nonetheless, in the peptidoglycan molecule, lactate loses a single heavy atom due to its connection to L-Alanine, thus resulting in a second exact match.

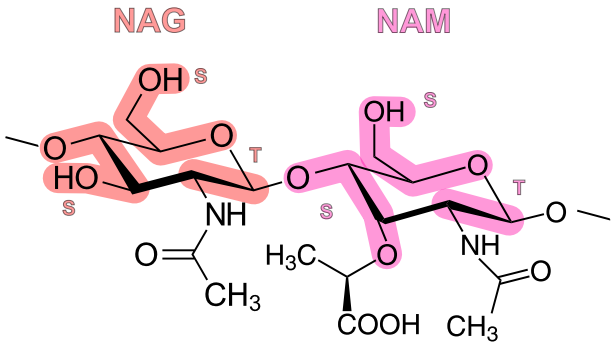

Figure 3.4: **Possible alternate mapping that assigned the bridging O was assigned to the carbon of lesser index [20]**

During the parameterization process, the mapping and the bead type assignment go hand-in-hand. The sources utilized to inform us of mapping standards also provide information on the right bead types to use in each instance. Each bead type selected was based on a reference molecule present in the guidelines for either small molecules or sugars. All of this information can be consulted in Table 3.1.

Table 3.1: **List of all the beads and their represented atoms and assigned type.** The references used to decide each bead type are also shown.

Name	Atoms	Bead Type	Reference
G1/M1/M2	CH-CH-O	SN4ar	Deoxy-Hemiacetal [20]
G2	CH-CH-OH	SN6	Alcohol [20]
G3/M3	O-CH-CH_2-OH	SP1r	Hemiacetal [20]
G4/M4(VS)	Pyranose Ring	TC4	—- [20]
ACN	NH-C(=O)-CH_3	SP3d	Amide [20]
LAC	CH(-CH_3)-COOH	P2	Propanoic Acid [24]

As previously mentioned in section 2.10, we did not follow the Center of Geometry (COG) methodology for bead placement. Instead, we carefully placed each bead by assigning different weights to different atoms to ensure the best SASA matches between each chemical group and its representing bead. Through trial and error, we continually fine tuned the position of the beads and checked the corresponding SASA against the target surface until the two matched. The final scheme can be seen in Fig. 3.2.

Table 3.2: **List of all beads and the respective atoms used to position them.** A Center of Weights approach was used to position each bead. The number attached to each atom correspond to their relative weight when calculating the new bead's position.

Name	Atoms
G1/M1	C1 C2 $O4^2$
G2/M2	C3 C4 $O3^2$
G3/M3	C5 $C6^3$ $O6^4$
ACN	NC CM OM^2 CC^3
LAC	CA C O^2 CB^2

One issue of note is that the SASA of a CG bead will always be a perfect sphere. Consequently, if a chemical group branches out in opposite ways, such as the amide's O and CH_3 atoms, it is not possible to perfectly match the SASA with a single CG bead. In these situations, we simply try to compromise between the two and place the bead such that its surface partially covers both. This results in a frequent underestimation of CG SASA values and is one of the down sides of most CG models, caused by the limitations of utilising less particles.

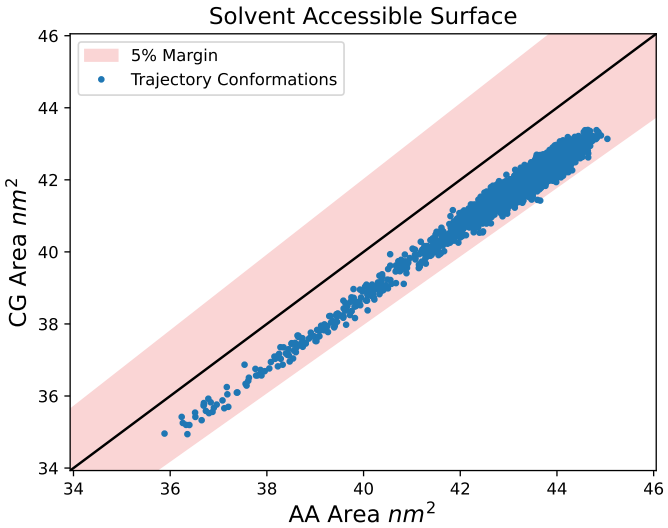

Figure 3.5: **Solvent Accessible Surface Area (SASA) comparison between the original AA glycan model and its mapped counterpart across a 400ns molecular dynamics simulation.** The overwhelming majority of mapped glycan conformations fall inside the 5% discrepancy margin deemed acceptable for CG models. A frame skip of 40 and a 0.191nm probe were used.

Following Martini 3 guidelines, there should be a 5% discrepancy between a mapped trajectory and its reference simulation [20, 54]. Therefore, to measure the quality of our mapping, we compared the SASA value of every conformation adapted by the glycan chain over the course of a ~400 ns AA simulation, to its mapped counterpart. As shown in Fig. 3.5, the SASA value of the mapped conformations never drops bellow the 5% margin. Nonetheless, the SASA values for the mapped model are consistently bellow their AA counterparts for the reasons previously discussed.

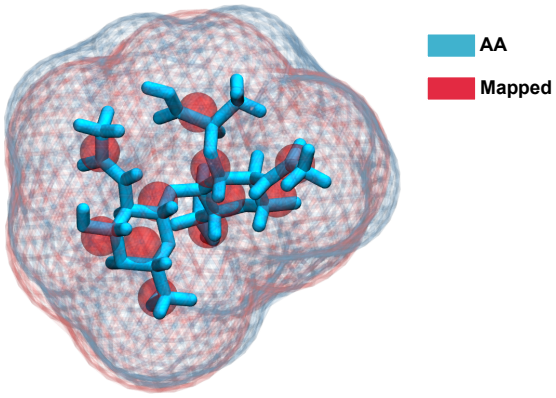

Figure 3.6: **VMD rendering of a simulation frame showcasing the visual differences in Solvent Accessible Surface Area between a single disaccharide unit in the AA model (blue) and its Mapped counterpart (red).** Overall, both SASAs are in good agreement. Most discrepancies occur in the functional groups ACN and LAC due to their forked nature. Unfortunately, these could not be fixed without increasing bead count or compromising the SASA somewhere else.

Furthermore, we must be aware that matching SASA values does not necessarily mean there is an overlap of the AA and mapped surface areas. It could be the case that some areas are underrepresented, while others over-compensate for them. Given this, we also visually checked the SASA results to ensure there was a good overlap between the two (Fig. 3.6). Although discrepancies are present, not only could these not be fixed without increasing the number of beads, but also part of the mismatches would get attenuated once the peptide connections was established, due to changes in the lactate motif. As such, we found that overall the mapped and AA models SASA matched adequately.

3.4 Bonded Interactions Scheme

Satisfied with our mapping scheme, we moved on to defining the design philosophy. The division of the parameterization process between glycan chain and peptidoglycan prompted us to parameterize the glycan chain in such a way that the model could reproduce the reference atomistic data with as few parameters as possible. The idea behind this decision is that, by not heavily restraining the freedom of the glycan chain, it will be able to easily adapt to the restraints imposed by the peptide chain later on. This would facilitate

the transfer of parameters between the glycan chain and the final peptidoglycan model and decrease the overall number of parameters that would need to be tweaked.

Given this, the initial model was built with the following bonded interactions in mind: the triangle representing the pyranose ring would be held together only with bonds, creating a reference central plane and functional groups (ACN and LAC) would be controlled using one bond, one angle (that controls the side-to-side movement) and one dihedral (that controls out of plane movements). This angle could also be swapped for an additional bond if it was deemed too instable to recreate the bonded behavior. Constraints would be used where too high bond force constants would be needed.

The links between residues are the most complex connections of the model, since they were initially composed of 1 bond, 2 angles and 3 dihedrals. While one dihedral is responsible for the torsion around the link bond, the other two dihedrals are in charge of correctly positioning the linking bead to the opposite triangle plane. The angles are complementary to the first dihedral, which is also named the main torsion dihedral. In the end, we noticed one more dihedral and its complementary angles were also needed to control the behavior of the torsion around the link bond, pushing the final count to 1 bond, 4 angles and 4 dihedrals.

Finally, we also checked both terminal residues to gauge if there were any differences in functional group and link distributions. As an example, in Fig. 3.7 we can see there are no meaningful enough differences in the terminal distributions for the amide group when compared to all other residues in the glycan chain. Both lactate and link terminal distributions are also similar to the rest and, thus, terminal residues were not treated differently from any other in our model. All the analysed bonds, angles and dihedrals are available in the appendix A.

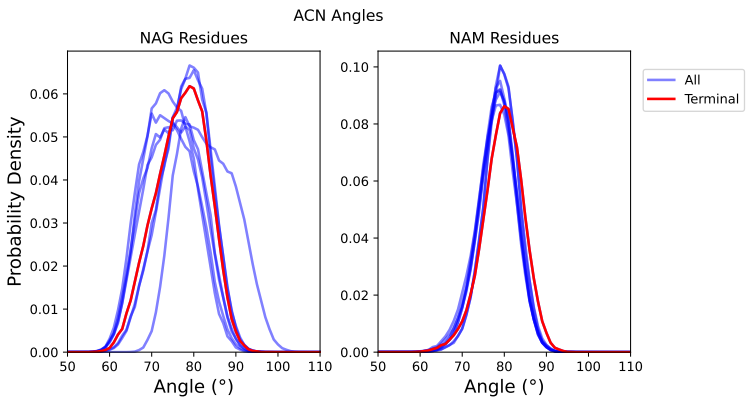

Figure 3.7: **Comparison of the angle controlling ACN functional group in both monosaccharides.** There is not a significant enough difference to justify the parameterization of specific terminal residues. The same conclusion was reached using all the other bonds, angles and dihedrals analysed.

3.5 Initial Coarse-Grain Glycan Chain Models

In this work, the parameterization process of the glycan chain was done through a trial-and-error approach where we iteratively ran simulations, compared each parameter's distribution to its mapped counterpart and adjusted the parameters accordingly to better mimic the reference data.

Initially, we guessed each parameter's equilibrium point by using its median value and started the trial-and-error process henceforth. To ensure stability, we first introduced the bonds, followed by angles and finally dihedrals. In the end, the results of these initial glycan chain simulations led to the creation of a stable model which, on average, was able to capture the behavior of the AA model.

We evaluated the fitness of the model by plotting the CG distribution of all parameters used to create it against their AA counterparts. In Fig. 3.8, we can see that the CG and AA bond distributions are in good agreement. Notably, some parameters, such as the bond connecting the second and third beads of the rings, *G2_G3* and *M2_M3*, have a bimodal distribution. These types of distributions can not be perfectly recreated with a harmonic potential due to it only having one possible minima, and, thus, the medium point between the two modes is used as the CG bond equilibrium point.

The angle distributions of the CG model were also in good agreement with their AA counterparts, as shown in Fig. 3.9. Particularly, only NAG's amide functional group (ACN) was not parameterized using the standard one bond, one angle and one dihedral scheme that was used for other functional groups. Without any exclusions this scheme could not reach the correct *ACN_G1_G2* angle, due to clashes between the ACN and G2 beads. We first tried to introduce an exclusions between these two clashing beads and, although rare, some instabilities arose from singularities in the dihedral controlling out of plane ACN movements. Playing it safe, we swapped this angle for an additional bond between G2 and the amide group. This explains the non-highlighted *ACN_G1_G2* angle in Fig. 3.9 and the non-standard *G2_ACN* bond, back in Section 3.8.

Finally, the dihedral angles were imposed and refined until the match we see in Fig. 3.10 was achieved. At this point, we were still not aware of the importance of a second dihedral controlling the torsion of the links between sugars and, thus, both *G2_G1_+M2_+M1* and *M2_M1_+G2_+G1* were not yet enforced. Back in Fig. 3.9, we can also see that their complementary angles were not enforced neither. When some of the target dihedrals could not be easily captured with the use of a single function, we used an in-house script to sculpt the underlying function of a distribution with a series of additive potentials. As an example, the two maxima in the *NAG_triangle_+M2* dihedral are too far apart to be captured with a single proper dihedral potential. In such cases we used our script to develop a more complex function composed of additive potentials to recreate the target distribution.

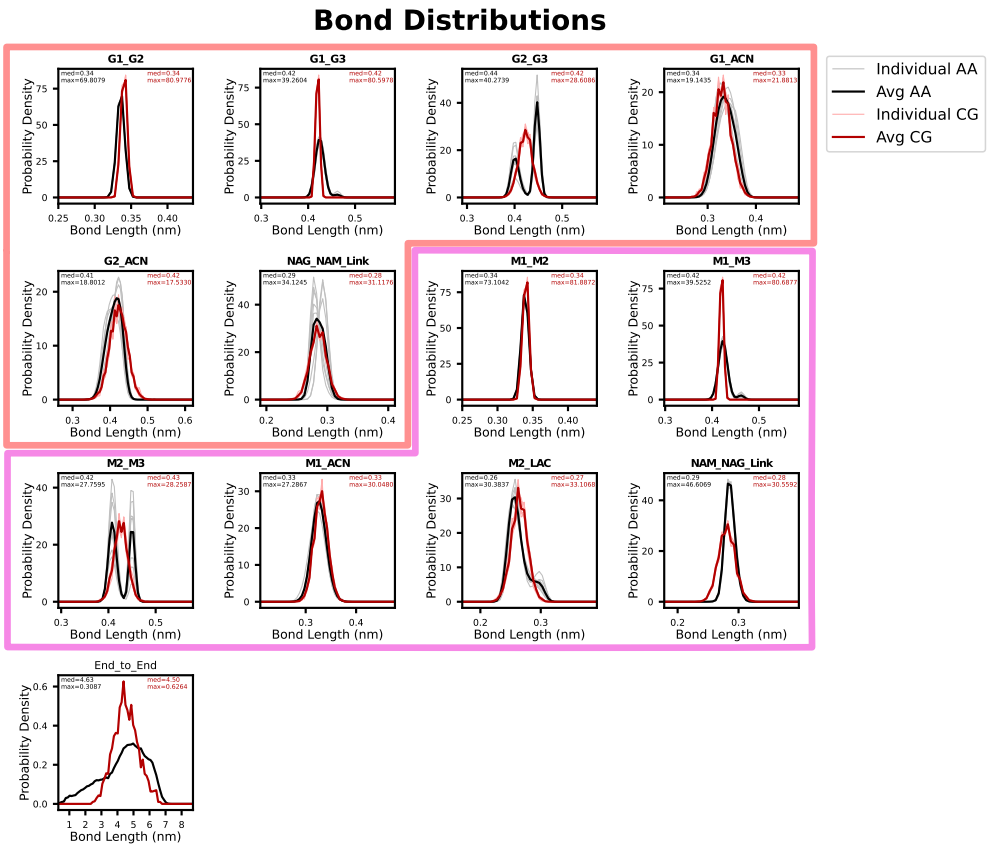

Figure 3.8: **Distributions of all bond parameters that compose the initial glycan chain model.** The CG model is capable of accurately reproducing the average behavior displayed in the AA simulations. However, the end-to-end distance hints at a lack of flexibility in the glycan chain. Henceforth, the orange and pink borders encompasses all parameters that start from the NAG and NAM residue, respectively. Additionally, parameters in bold are enforced by potentials or constraints and all others are purely monitored.

In this initial parameterization effort we used very small simulation times of around 50 nanoseconds (ns) to obtain our distributions. This was done as we were simply trying to recreate the bonded parameters' behavior, without much consideration for any kinetic behavior such as flips occurring. As an example, we can see that when looking at the main torsion dihedral of the NAG-NAM disaccharide (*G3_G1_+M2_+M3*) in Fig. 3.11, no flips occur in the CG model. The absence of flips in our CG model has two explanations. First, as flips seemed to occur at the 100 ns time-scale, not enough time was being simulated for them to occur. Furthermore, as recreating these flips was not a goal for our initial model, there was a lack of parameterization for this specific behavior.

Our initial parameterization effort was able to design a stable model that, on average, accurately recreated the AA distributions. Nonetheless, we were aware that the kinetics of the links between residues, meaning the rate at which transitions between the "linear" and "kinked" states occurred, were off. Given that these states have both a significant and long-lasting impact on the overall structure of the glycan chain backbone, especially the end-to-end distance, the lack of parameterization of these kinetics could impact the

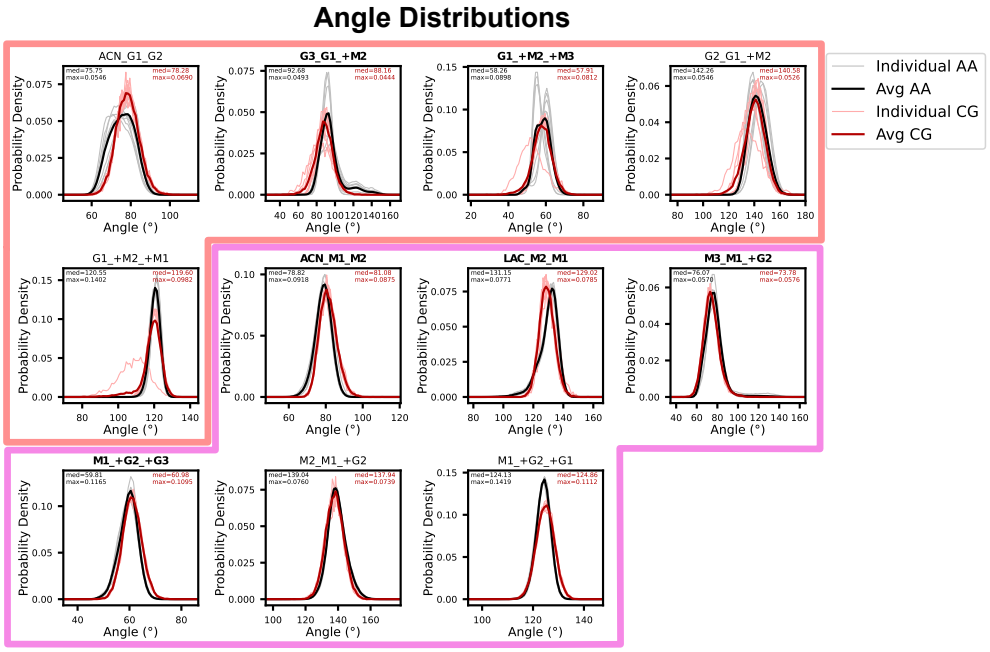

Figure 3.9: **Distributions of all angle parameters that compose the initial glycan chain model.** Once again, we find good agreement between the distributions from the CG and AA simulations. At this point in time, we were not aware of the importance of some of the dihedrals controlling the link between residues and, therefore, their complementary angles were still not enforced.

accuracy of our model. This phenomenon can already be seen back in Fig. 3.8, where the lack of flips in the initial model lessens the flexibility of the glycan backbone and leads to a mismatch in the end-to-end distance distribution.

It was both the importance of this kinetics and our curiosity with the readiness of the Martini 3 force field to be able to capture this behavior that made us pursue the parameterization of the link kinetics.

3.6 State Transition Kinetics

Although there are two different disaccharide units, the flips in the NAG-NAM disaccharide were heavily prioritized during this parameterization effort. While the NAG-NAM link has two well-defined ψ minima and the flips that occur between these two states cause long-lasting impacts in the molecule's configuration, the flips in the NAM-NAG link very rarely lead to long-lasting changes in the molecule, as it very quickly switches back to the clearly preferred negative ψ minima showed in Fig. 3.2.

With this in mind, the parameterization effort of the NAG-NAM flips entailed three goals. First, we wanted to match the average time spent in each state. Second, we wanted to match the rate at which transitions occurred between states, which we named the flip rate and was measured in Flips per Disaccharide Unit per microsecond (Flips/DU/μs). Finally, as transitions between states can occur through the 0° or 180°, we also wanted to

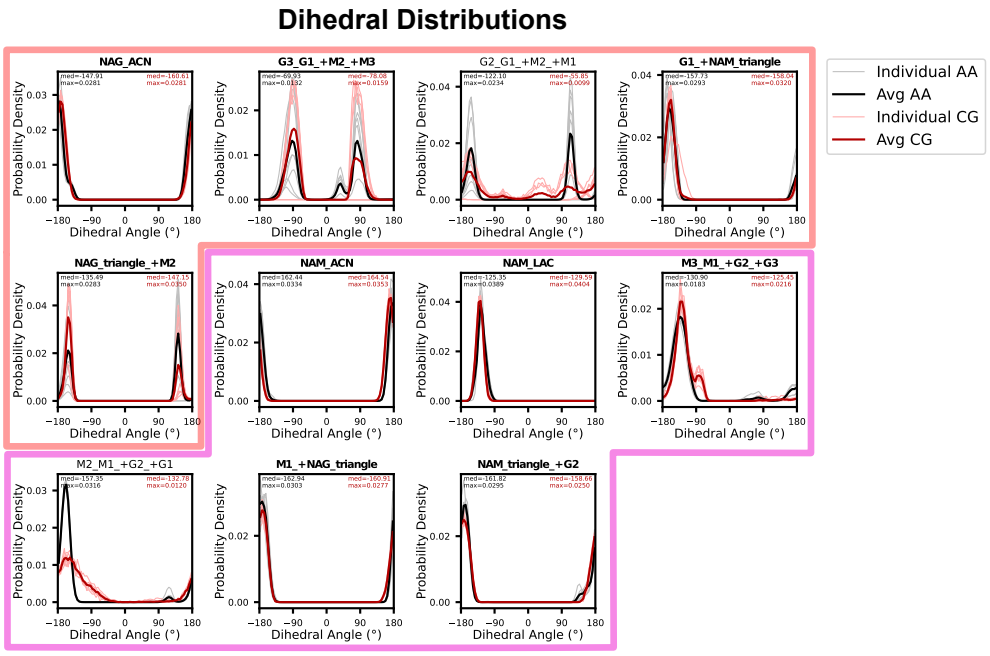

Figure 3.10: **Distributions of all dihedral parameters that compose the initial glycan chain model.** Unlike in the AA simulations, the individual and average distributions of the *G3_G1_+M2_+M3* dihedrals are not the same. This occurs due to individual NAG-NAM units not having the freedom to transition between the two torsional states.

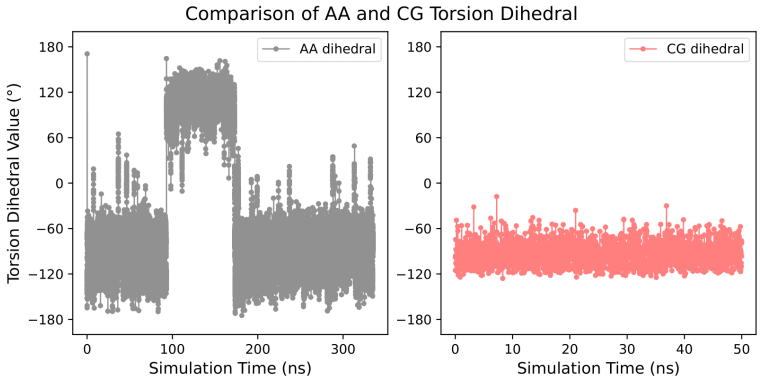

Figure 3.11: **Comparison of the AA and CG main torsion dihedral in a NAG-NAM subunit of the glycan chain.** Even if the CG simulation time is much shorter than the AA one, some transitions between the two torsional states were still expect to occur. As none happened across the entire glycan chain, this result hints at a lack of freedom to freely reproduce this behavior.

match how often each path was taken to transition between states. For an explanation on the scripts used to calculate these look back to Section 2.14.2.

The goal behind this second parameterization effort is to refine the parameters to match the behavior of a single disaccharide unit, transfer them to the original glycan chain and allow the non-bonded interactions to correct the behavior. If both the disaccharide units' parameters and the non-bonded interactions are correctly captured, then we hoped that the flip kinetics and end-to-end distance of the glycan chain would also match.

3.6.1 Obtaining Kinetic Reference Values

To parameterize NAG-NAM flips correctly, we first needed a robust reference simulation where a sufficient amount of flips occurred so as to guarantee all possible configuration of flips were properly sampled. Given this, our all-atom glycan chain simulation was deemed ill-fit, as it was not long enough to properly sample the flip rate, with only a total of 12 flips being recorded.

That being said, we decided to run new simulations to gather more data. We believed that the length of the molecule, in addition to the time scale, was impeding a good sampling of the flip rate dynamics. Even if a longer glycan chain has more links that could perform flips, the kinetics of the flips get slowed down due to the hindrance caused by the disaccharide units behind and in front.

Therefore, instead of simply extending the ~400 ns glycan chain simulation, we developed a NAG-NAM disaccharide structure and topology by analogy of the full glycan chain and capped it with a methyl group. Using this structure, we ran 4 replicates of 1 500 ns each to get a better sampling of the kinetics of the NAG-NAM link. The simulations were analysed independently and the kinetics results where averaged out afterwards to create the flip rate reference points.

Across all 4 replicates, a total of 81 flips were recorded. The data regarding the average flip path and direction can be seen in Fig. 3.12. Notably, the path through 180° is 4 times more populated then the one through 0°, which hints that the energy hills separating the two minima are smaller in the 180° direction.

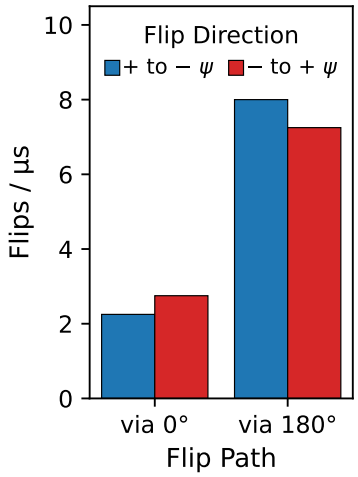

Figure 3.12: **Bar plot of the average flip rate via each path and through both possible direction in a NAG-NAM AA simulation.** Flips via 180° are clearly preferred by the NAG-NAM disaccharide. There is no directionality preference inside of each via which is expected in a properly calibrate and accurate system.

Additionally, the number of flips occurring via the same path is equal in both directions. If, conversely, one of the directions was preferred, the molecule could seemingly rotate in the same sense forever due to the periodic nature of torsions (violating the second law of

thermodynamics in the process). The flip path being independent of flip direction also indicates a correct equilibration process was previously taken by allowing the system to converge properly.

Finally, all target values were gathered from this replicates to form our references which can be consulted later on, in Fig. 3.16.

3.6.2 Parameterization of CG Kinetics

By themselves, the atomistic reference distributions only give us a target for the parameterization effort, but they do not give us any information on how to reach said goal. For this, we first had to study what occurred right before each flip to better grasp how to introduce flips into the initial CG model.

It was by analysing the mapped trajectories that we noticed that some beads would get close enough during the flipping process to cause repulsions at the CG level. After studying the CG trajectories, we confirmed that the main problem blocking flips were indeed steric clashes between beads. This explains why, without any changes to the initial parameters, flips very rarely occurred during the initial CG parameterization simulations. As exemplified in Fig. 3.13, flips through different vias lead to clashes between different beads at both sides of the link, which stop the flipping from completing and confine the links in their initial configuration.

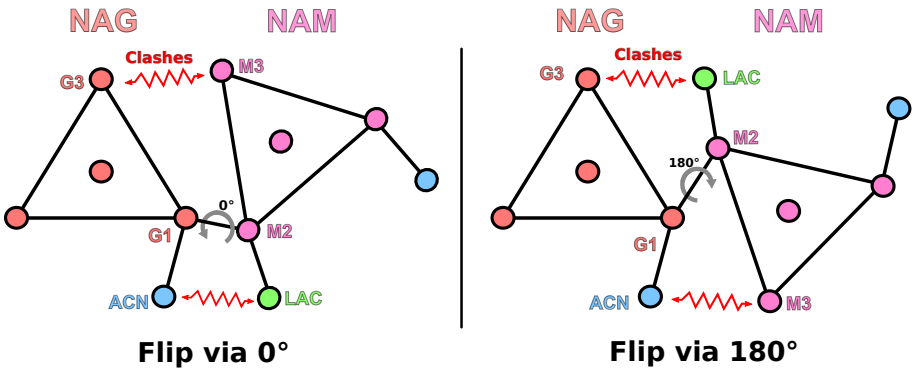

Figure 3.13: **Schematic showcasing the clashes that prevented flips from completing in the initial glycan model.** Depending on the via of the flip, different beads would clash and prevent the flips from completing. Flips occurring via the 0° cause clashes between the *G3* and *M3* beads and the *ACN* and *LAC* beads of NAG and NAM, respectively. On the contrary, flips occurring via the 180° cause clashes between the *G3* and *LAC* beads and the *ACN* and *M3* beads of NAG and NAM, respectively. This resulted in disaccharide units having their main torsional dihedral be stuck in its initial conformation.

Furthermore, our distributions only gave us a one dimensional view into each individual distribution, which means that they can not give us any insight into correlations occurring between themselves. To overcome this, in Fig. 3.14 we plotted the distance between one of the clashing beads pair against the main NAG-NAM torsion dihedral to better visualize wether clashes were the reason for the lack of flips. As shown in the previous schematic (Fig. 3.13 left), the beads *G3* and *M3* clash when the flip happens via

the 0°, and, as shown by the **mapped** trajectory (Fig. 3.14 A left), these two beads get the closest when the *G3_G1_+M2_+M3* dihedral is around the 0° range. The same can be seen for the clashes via 180° (Fig. 3.13 right), as the beads *ACN* and *M3* clash when the flip happens via the 180°, and, as shown by the **mapped** trajectory (Fig. 3.14 B left), these two beads get the closest when the *G3_G1_+M2_+M3* dihedral is around the 180° range.

As expected by the lack of flips, **before** the introduction of the pair scheme (Fig. 3.14 center), the disaccharide units where restricted to their initial configuration and the clashing beads could get much closer in the **mapped** trajectory (left) than in our initial glycan model (center). This pointed us towards the fact that the clashing beads during flips were repulsing each other excessively and preventing the flips from completing. Consequently, this lack of flips resulted in a poor sampling of the conformational space as NAG-NAM disaccharide units were not allowed to leave their starting configuration.

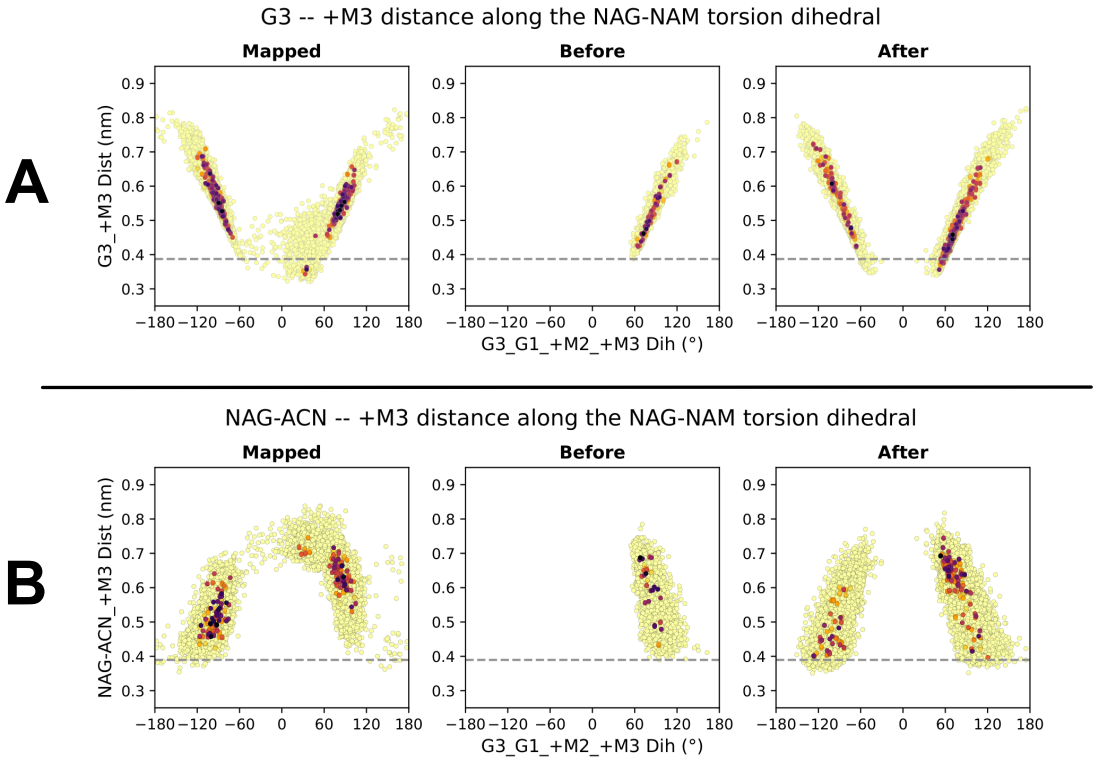

Figure 3.14: **Correlation analysis of the main torsional dihedral of a NAG-NAM unit against the distance between the G3 and M3 beads (A) and the ACN and M3 beads (B).** The grey dotted line shows the lowest recorded distance between beads before the implementation of the exclusion and pair scheme. In the mapped simulations these beads get much closer together, which allows for transitions between both torsional states. After the introduction of the exclusion and pair scheme, as beads were allowed to get closer than before, flips between states began to occur as well and the two torsional states were abled to be sampled by our CG model.

With this knowledge in mind, we decided to use GROMACS's exclusion and pair interactions to completely override the default interactions between beads involved in clashes and precisely control the repulsion/dispersive forces occurring around the link.

This scheme will be referred to as "exclusion and pair scheme" from now on and its core can be visualized in Fig. 3.15. All exclusions and pairs are provided in the appendix B.

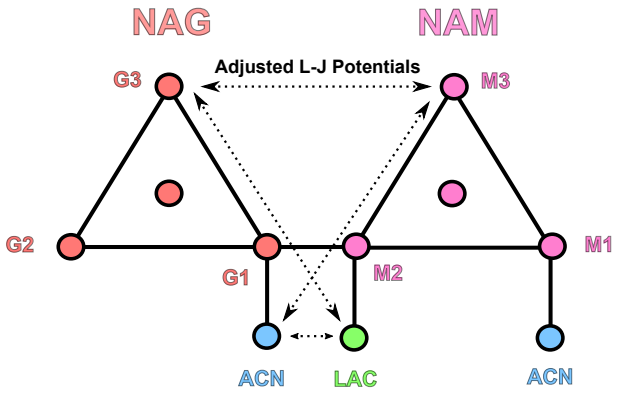

Figure 3.15: **Schematic showcasing the exclusion and pair scheme developed to reintroduce flips to the CG glycan model.** This scheme allows for a deeper control of all non-bonded interactions happening around the link between disaccharide units. This is done by completely overwriting the base L-J potentials with new ones to allow for beads to get closer and, consequently, flips to occur. Apart from the highlighted exclusions and pairs, there was also an exclusion-pair between beads G1 and LAC. This one was mainly precautionary to avoid interferences and is omitted in the scheme.

Just like before, a trial-and-error approach was taken to the parameterization process of NAG-NAM kinetics. First, we ran 4 all-atom replicates for 10 000 ns, analysed each trajectory independently and averaged out all results. Then, we compared these AA results to CG ones and modified the σ and ϵ values controlling the pairs in the CG model, to adjust how many flips occurred and in what direction.

As a rule-of-thumb, the bigger the σ values the further away beads "feel" reach other, resulting in stronger clashes and fewer flips. The larger the ϵ value the closer beads got to each other, resulting in more flips occurring from the increased number of pre-transition states. As an example, if we wished to specifically increase the number of flips occurring via 0°, we would decrease the sigma values of the interaction between *G3* and *M3* and NAG's *ACN* and *LAC*, while increasing their epsilon.

The results of the disaccharide simulations employing the exclusion and pair scheme are showcased in Fig. 3.16. Overall, our CG disaccharide model was able to successfully recreate the AA flip behavior. As shown, the flip rate is slightly increased in the CG model, which we found to be acceptable due to the fast kinetic nature of the CG resolution causes by its smoothed energy landscapes. Furthermore, both the fraction of time spent in each state and the paths used during flips are in good agreement. In the end, this results validated our exclusion and pair scheme and were able to achieved the stated goal.

Although the most influential, the exclusion and pair scheme were not alone in modulating the flip rate. When studying the trajectories, we noticed that some additional tweaking was needed in the bonded parameters to accommodate for flips. Namely, for flips to occur via the 0°, the *G3_G1_+M2* angles needed the ability to reach the ~120°

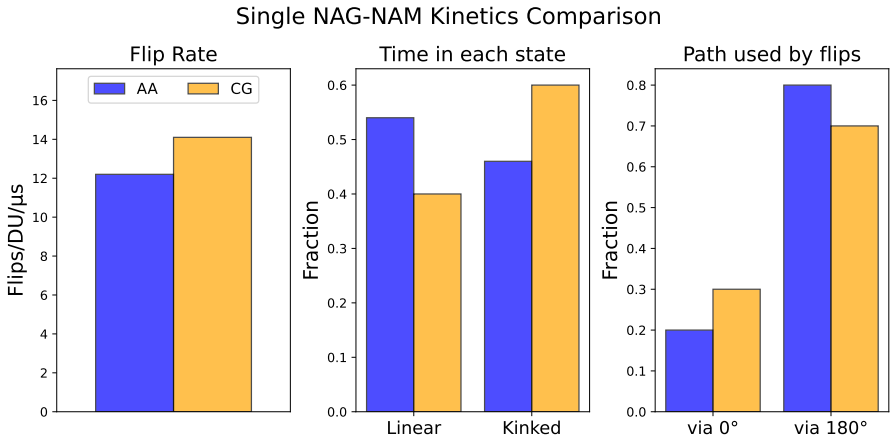

Figure 3.16: **Kinetic results from the parameterization process of a single NAG-NAM disaccharide unit.** Only NAG-NAM flip kinetics were accounted for during this parameterization effort due to it having two clearly defined and long-lasting minima (unlike NAM-NAG).

values. Additionally, the force controlling the *G3_G1_+M2_+M3* dihedral was crucial in controlling the flip rate, as the larger the value the fewer flips occurred, if at all. Thus, this dihedral force had to be tweaked until a perfect middle point was reached between constant flipping or no flips at all.

Looking back at Fig. 3.14, we can see that **after** the introduction of all the new parameters (right), the *G3* and *M3* beads were allowed to get closer together than **before** (center). Now, the new model (right) mimics the **mapped** trajectory (left) by allowing the beads to get closer, which results in flips to occur and both states to be sampled accurately. Additionally, it is worth mentioning that the correlation between parameters is something not actively looked for during most CG parameterization processes and it is left to assumption that, if all bonded and non-bonded parameters are captured, then so will the correlations. These plots are also able to confirm that the correlation between these two parameters was maintained in our model, despite not actively looking for it.

Finally, it is important to note that the script analyzing the flips uses the frames preceding a flip to determine its direction. If the interval between saved frames is too large, then it could give enough time for the dihedral value to escape the ranges defining a flip's direction and cause misassignments. To prevent this, we always made sure that both the AA and CG frame's were saved frequently to minimize the risk of such errors occurring.

3.6.3 Transferring Parameters to Glycan Chain

Despite the good agreement in kinetic results, some final changes were done when transferring over the parameters. Namely, for the sake of simplicity and standardization, all pairs' ϵ values were changed back to their original values. To account for this change, some σ values had to be brought down as well to account for the decreased dispersion.

After all adjustments were done, two replicates of ~3 500 ns were run to collect data on the new glycan CG model. The CG distributions were obtained by joining the two replicates together and analysing the ~7 000 ns long trajectory. For the target kinetic values, each replicate was analysed independently and their values were averaged.

Before the introduction of the pair scheme, the lower distance regions of the distributions were not sampled by our CG model due to the aforementioned clashes. Now, as shown in Fig. 3.17, the pair scheme allows the CG model's bead to sample the conformations at which they are the closest. The fine-tuning of the sigma values also allowed us to control exactly how close the two beads could get, thus preventing any over-corrections from occurring.

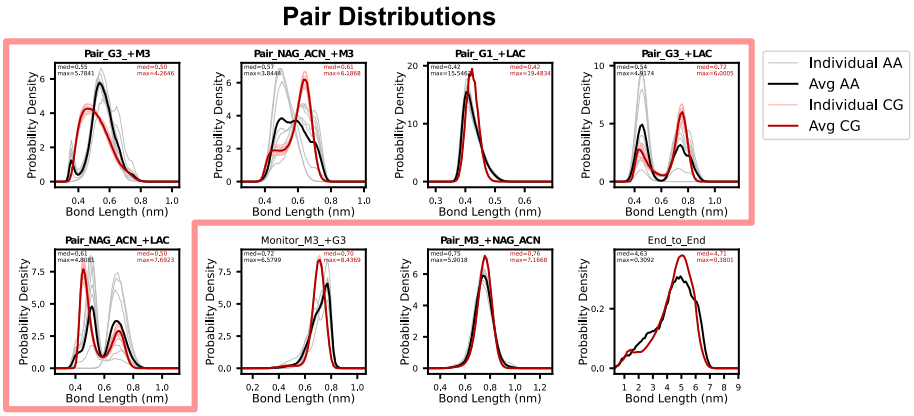

Figure 3.17: **Distributions of all exclusion and pairs introduced for the final glycan chain model.** The refinement of previous parameters and the exclusions and pairs scheme allowed for the reintroduction of flips to the model. This confers to the glycan chain the flexibility it needed to more accurately match in the end-to-end distance.

Fig. 3.18 shows the differences in the torsion dihedrals due to our second parameterization effort. As shown, our exclusion and pair scheme translated accurately to the full glycan chain model and allowed each disaccharide unit to flip between their two preferred states. This freedom means that each disaccharide unit is no longer stuck in their initial configuration, causing the massive individual peaks we had seen before in Fig. 3.10. This is especially evident in the *G3_G1_+M2_+M3* dihedral.

Although not given the same amount of attention, the NAM-NAG dihedral distribution now closely follows its AA counterpart with both states being sampled as well. To achieve this, we "drew" the underlying potential with our script and used a single exclusion and pair combo between the *M3* and NAG's *ACN* bead. Getting the +90° state to be sampled proved difficult due to our script heavily prioritizing the massive size of the -130° state. To overcome this, we artificially boosted the distribution side on the +90° side before sculpting its underlying potential. With this, the new sculpted potential had a much deeper metastable state on the +90° side, allowing for both sides to be sampled proportionally .

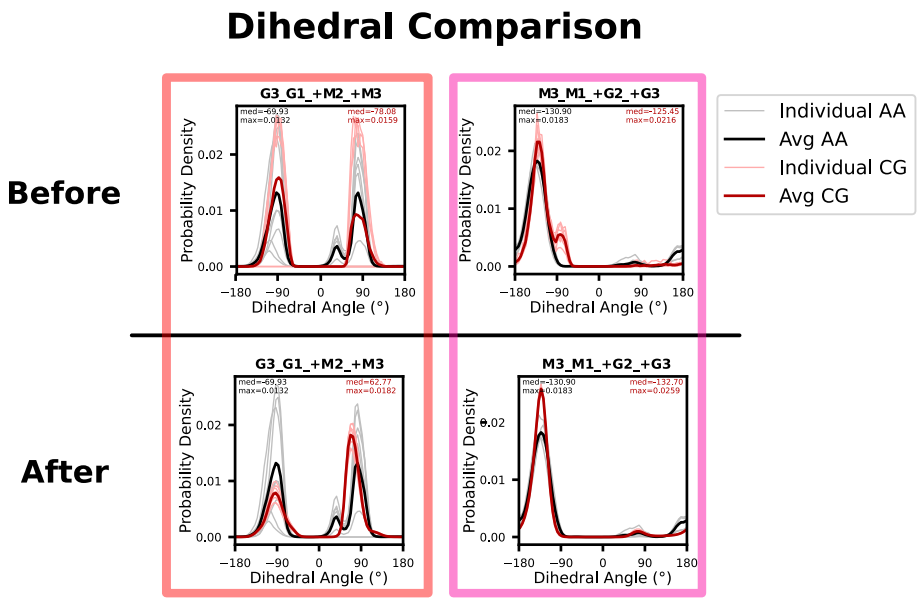

Figure 3.18: **Comparison of the distribution for the main torsion dihedral of NAG-NAM (left) and NAM-NAG (right) before and after the reintroduction of flips.** By reintroducing flips, the exclusion and pair scheme allowed for the individual CG distributions to sample both torsional states instead of being stuck in their initial conformation like before.

Finally, we also compared the NAG-NAM kinetics results of the AA and CG full glycan chain to gauge how accurately the kinetics translated to the full model. Fig. 3.19 showcases the results. First, the Flip Rate is accelerated in our CG model, and, just as before, we deemed this result adequate due to the faster kinetics inherent to the CG resolution. Although not too significantly, the fraction of time in each state is the value which most deviates from its AA target. Our main hypothesis for this divergence is that the AA trajectory was not long enough to allow for all dihedrals to converge correctly. This idea is supported not only by the low number of total flips in the original AA (only 12 were recorded) and by the huge difference between the individual and average AA distributions pointing towards a lack of convergence on this dihedral, as seen back in Fig. 3.18. Finally, the fraction of path used by the flips in our CG model matches almost perfectly its AA counterpart.

Even though we must be aware of the limitations in the atomistic sampling, these results support the accuracy of our model. The combination of sculpting underlying potentials and the ability of the exclusion and pair scheme to perfectly fine-tune each interaction, gave us the tools needed to correctly capture the interactions occurring around the link between disaccharides. The ability of our model to recreate higher-level behavior, such as flip rate and end-to-end distance using parameters fine-tuned in single disaccharide unit simulations also contribute to the notion that the bead size and type choice contribute to faithfully reproduce the non-bonded interactions of the entire molecule. Given this results, we decided to move onto the peptidoglycan chain.

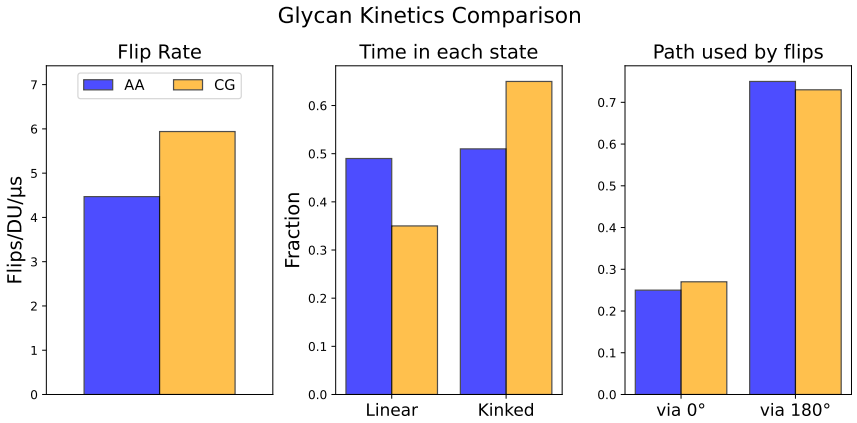

Figure 3.19: **Comparison of the kinetic results from the AA and CG full glycan chain simulations** We found there was good agreement in the kinetic results of all evaluated parameters. Only NAG-NAM flip kinetics were accounted for.

3.7 Metadynamics Simulations

Just as before, we began the parameterization process of the peptidoglycan's glycan strand by obtaining some reference simulations from our collaboration with the Ayappa group at the Laboratory for Molecules at Interfaces.

One of our main questions regarding the peptide chain, was its effect on the two states of the disaccharide units. It could be that the peptide chain completely removes one of the states or simply slows down the flip rate between the two. However, when we reran the analysis by plotting the ϕ and ψ angles (C+1) of the glycosidic bonds, we saw something completely unexpected. As shown in Fig. 3.20, new states suddenly appeared in the positive ϕ regions of both disaccharide units, which seemed to be mirrored and inverted copies of their negative ϕ counterparts.

We found the sudden appearance of these new states to be very confusing. Assuming the positive ϕ states were as probable and prevalent in the glycan chain as they are here, then it stands to reason they should have been recorded before, especially given that we expected the peptide chain to slow down the transitions between all states.

Our main hypothesis was that when building the initial peptidoglycan structure, not enough care was given to ensure the torsions of the links between disaccharides were at their most representative states. However, it could also be that the transitions between the negative and positive ϕ states happen only at a larger time scale, which we had not yet reached with the previous glycan simulations, causing these new states to not have been sampled yet. Even so, this last explanation seems unlikely, as the initial peptidoglycan simulation was 200 ns long, which is even shorter than the glycan's 400 ns simulation.

Wary of these new findings, we decided to investigate further to ensure we are utilising a representative starting point for our simulation before moving forward with our peptidoglycan parameterization effort. Otherwise, the disaccharide units could simply be stuck in high energy minima due to high energy barriers separating the two ϕ states.

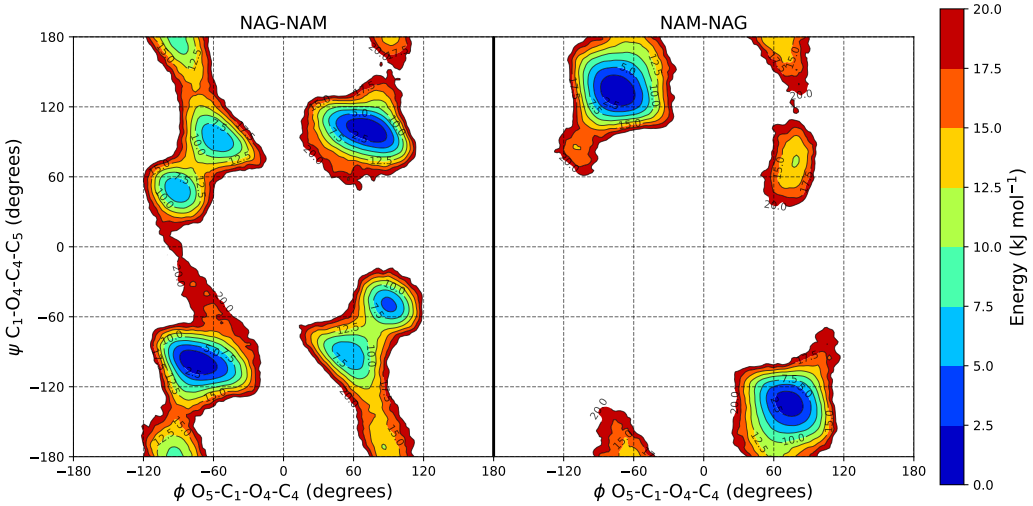

Figure 3.20: **Contour plots for NAG-NAM (left) and NAM-NAG (right) showcasing the changes in the ϕ and ψ torsions (C+1) across the initial all-atom peptidoglycan simulation.** Comparing these to the results from the glycan simulation shows that new states have appeared in the positive ϕ regions for both disaccharide units. The absence of these ϕ states in the previous glycan simulations made us suspect there were problems with the starting configurations of some simulations.

To study all possible configurations of our two variables of interest, ϕ and ψ, we resorted to metadynamics simulations to accelerate the kinetics of the links between sugars and sample a much wider range of structures. Simulations for both NAG-NAM and NAM-NAG were run using 4 parallel sampling simulations to speed up the process. Furthermore, no peptide chains were included in the structures used for these simulations, as they would greatly increase the dimensions of the system as well as slow down the kinetics. The problem here is that metadynamics assumes that the collective variables have proportionally sampled the respective phase space before placing gaussian potentials, and, as the peptide chain would slow down convergence, said assumption would be invalidated.

The results for the metadynamics simulations are shown in Fig. 3.21. The NAG-NAM results show that a total of 4 local minima exist, one in each quadrant (ϕ/ψ: -/- , -/+, +/-, +/+), and while this number coincides with the previous results, the local minima at positive ψ are at too high energies to have any preponderance. Given these results, we can probably conclude, as was hypothesized before, that the reason we had a roughly 50/50 split before was due to the energy barriers between the two ϕ minima being high enough to keep the torsions kinetically trapped in the positive ϕ configurations. Furthermore, the positive ϕ minima are not exactly located at the same ϕ, ψ coordinates that we found before, which could have occurred due to presence of the peptide in the previous simulation. Looking back at Fig. 3.20, it is also compelling that flips still occur between the two ψ states in the NAG-NAM units, despite being stuck in the unfavorable positive ϕ range due to a connecting path via 180°.

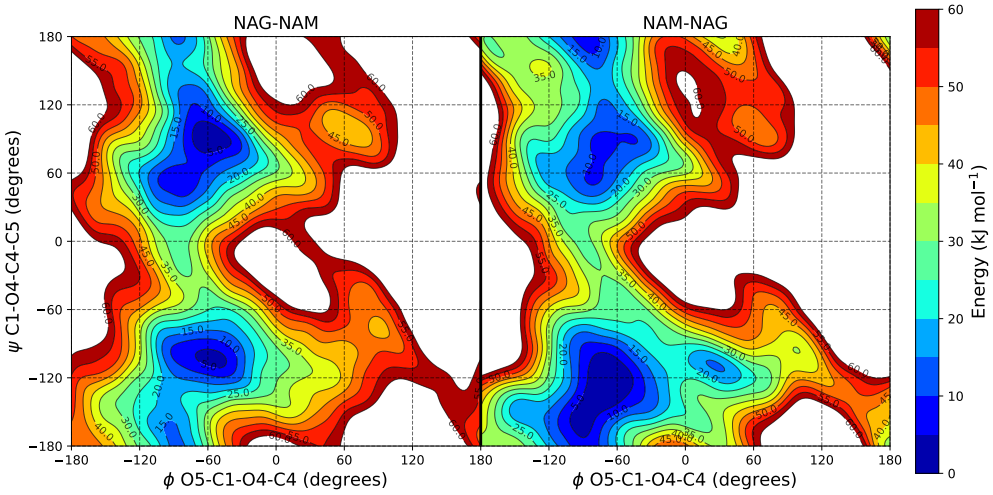

Figure 3.21: **Sum of Gaussian potentials in a metadynamics simulation of NAG-NAM (left) and NAM-NAG (right).** While these simulations did not have the peptide strand included, the energies exhibited by the minima at the ϕ minima are at too high for these states to have any preponderance.

The results for the NAM-NAG only come to further substantiate the idea that the minima at positive ϕ values were not representative of the disaccharide probable conformations. Interestingly, NAM-NAG does have a much more probable state at the mid-way point between the two negative ϕ states, which, looking back at Fig. 3.1, we can see that this metastable state was sampled back in the initial glycan simulation.

It should be noted that these results do not account for the influence of other disaccharides in the chain as we only have two. Additionally, they also do not account for the presence of the peptide strand and its influence on the torsion. Even so, we find that the most probable explanation for the results of the initial peptidoglycan simulation (Fig. 3.20) was the aforementioned idea that a lack of consideration for the initial structure of the glycan strand resulted in half the disaccharide units being kinetically stuck in high energy minima.

Considering all of this, the reference peptidoglycan simulations were rerun with the disaccharide torsions in the initial structure set to the more representative negative ϕ values.

3.8 All-atom Peptidoglycan Simulations

The new simulation received from our collaboration with the Ayappa group was ~400 ns long and the positive ϕ conformations were no longer sampled. Now, the ϕ and ψ angles (C+1) of the glycosidic bonds resemble much more closely the results from previous glycan chain (Fig. 3.2) and metadynamics simulations (Fig. 3.21), as shown in Fig. 3.22. Interestingly, the flips previously identified were detected in the peptidoglycan as well, which means the restrictions imposed by the peptide strand were not enough to stop the transitions between the two ψ minima.

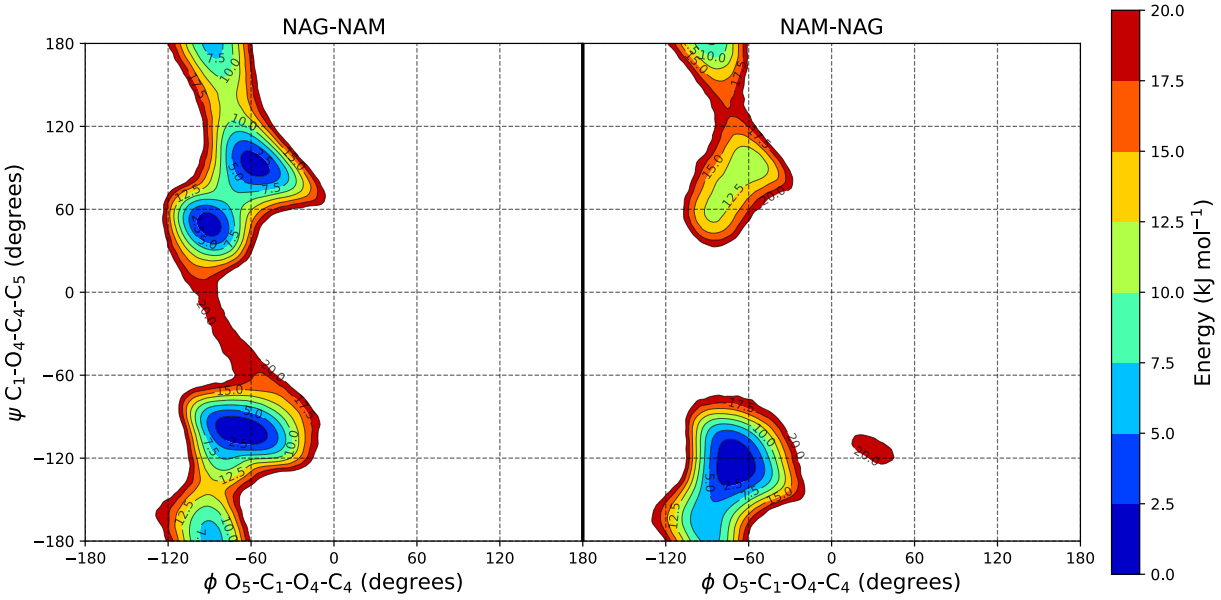

Figure 3.22: **Contour plots for NAG-NAM (left) and NAM-NAG (right) showcasing the changes in the ϕ and ψ torsions (C+1) across the final all-atom peptidoglycan simulation.** Using ϕ and ψ values that are closer to NAG-NAM and NAM-NAG's global minima led to the disappearance of the positive ϕ minima. This substantiates the hypothesis that the previous positive ϕ results arose from disaccharide units stuck in an unrepresentative initial conformations.

However, as we wished to ensure the system had appropriately converged, especially in regards to the flip rate and end-to-end distance, we decided to extend this simulation and create a new replicate. In the end, two replicates of ~4 000 ns were ran using the corrected starting point and, once again, these two were combined when calculating the distributions and analysed independently when looking at their kinetic values. As an example of the importance of simulation length, we can see in Fig. 3.23 that 400 ns was not enough for the simulation to converge and equilibrate properly. By extending the simulation we can be more confident that our reference results are more representative of the average behavior of peptidoglycan in a equilibrated state.

Even if flips were present in the peptidoglycan simulations, it was evident by a decrease in Flips/DU/µs that, as predicted, the bulkiness of the peptide chain slowed down the kinetics of the transitions. Interestingly, it seems that the first disaccharide unit was not as affected by the peptide chain as the rest. While the first disaccharide units had on average a total of 19.5 flips occurring throughout the simulations, the rest only had a total of 10.0 flips on average. Not being restricted by another unit behind it confers a lot of freedom to the first residue in a chain and it seems that the peptide chain further accentuates this effect. With this in mind, we decided to remove the first disaccharide unit when calculating the Flip Rate during kinetic analysis for both the reference and the model's peptidoglycan data to ensure a more stable measurement.

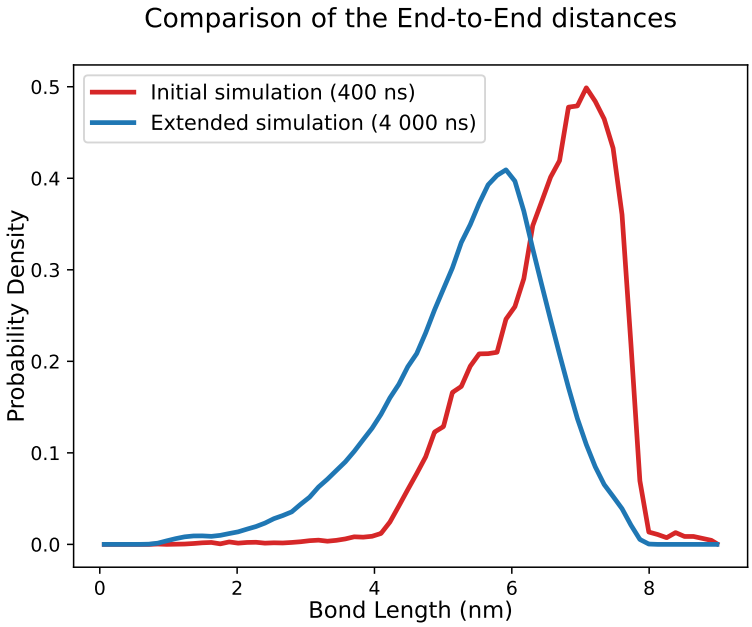

Figure 3.23: **Comparison of the end-to-end distances between the initial shorter glycan simulation and the final extended one.** Extending the simulation to 4 000 ns allowed the simulation to converge properly which brings confidence to the validity of the data we are training the model with.

3.9 Peptidoglycan Mapping

The mapping of the peptide strand of peptidoglycan was done by Ayappa's group and is shown in Fig. 3.24. As the only standard amino acid in the peptide strand, the first alanine was mapped as according to the amino acid guide lines for Martini 3. The second amino acid, glutamate, continues the peptide bond through its side chain instead of the backbone, which renders the standard mapping for glutamate invalid. To accommodate for the different linkage, glutamate was split into 3 total beads. While the old side chain now forms a second backbone bead (SN4a), the old backbone was split in two S beads, with SN6d representing the amine group and SQ5n representing the carboxyl group (now the side chain bead). As the only non-proteinogenic amino acid in the peptide strand, diaminopimelic acid does not have a standard mapping developed for it. As such, a new mapping was developed based on existing guidelines. Just like a normal amino acid, the backbone was assigned a single P2 bead. Then, its side chain was split into 3 beads: the first 3 CH_2 atoms were grouped into a single SC3 bead, the amino group into a SQ4p bead and the final carboxyl group into a SQ5n bead. Despite the last two D-alanines having a different chirality than the first, they were mapped exactly like it as its chirality is not maintained at the CG level. As the terminal amino acid, the backbone bead type for the second D-Alanine was changed to Q5. Finally, a negative charge was assigned to all carboxyl group beads (-3 charges) and a positive charge was assigned to the amino group (+1 charge), meaning the total final charge for each peptide strand was -2.

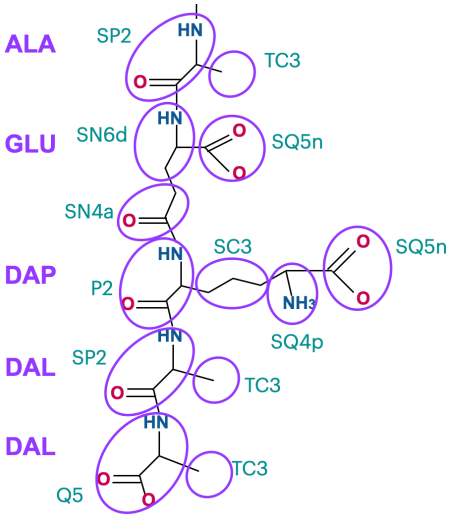

Figure 3.24: **Mapping for all the beads composing the coarse-grain peptide stem model.** The mapping was done by Ayappa's group by following the standard Martini 3 guidelines where possible and solved by analogy when needed.

Additionally, the link between the peptide strand and glycan strand incurs the loss of a oxygen atom in the lactate residue, which leads to a change of its bead size, from regular (R) to small (S), and bead type, from a P2 to a SN4a to reflect the increased hydrophobicity caused by the disappearance of a hydroxyl group. In the end, the mapping for the entire peptidoglycan chain was obtained by combining an altered glycan chain mapping (lactate mapping was changed to account for the peptide link) and Ayappa's group's peptide strand mapping.

3.10 Coarse-Grain Peptidoglycan Models

3.10.1 Transferring Glycan Parameters and Parameter Refinement

The parameterization effort for the bonded interaction of peptidoglycan was split between the two groups. The Ayappa group was responsible for the parameterization of the bonded interaction of the peptide chain, while we oversaw the transition of the glycan parameters to peptidoglycan and the link between the glycan and peptide strands through lactate.

The peptide strand was parameterized by the Ayappa group using the standard Martini 3 parameters where possible and developed when needed. Using what they developed, we started transferring all parameters from the glycan strand to gauge how accurately the glycan parameters would translate without any revisions.

While most individual glycan bonded parameters remained accurate, there were some inaccuracies in NAM's lactate and in the links between residues presumably caused by the peptide. Furthermore, there was also an almost total disappearance of flips and there were mismatches in the end-to-end distance.

We began by easily fixing lactate's bonded parameters by decreasing the *M2_LAC* bond by 0.01 nm, the angle controlling lactate's side-to-side by 10° and by increasing the dihedral controlling out-of-plane movements by 5°. Then, the next mismatches were all related to the links between residues and, as we later found out, were also responsible for the disappearance of the flips. First, NAG-NAM's dihedral *G3_G1_+M2_+M3* had its phase factor changed from -25° to -5°. To accommodate for the increase in the space needed for links to occur, the *G3_G1_+M2* angle was increased by 4° to be able to reach the most stretched populations. Finally, the dihedral *M2_LAC_ALA_B1_S1* was also sculpted precisely. This dihedral was critical for the flipping process as it had two relevant population states. Being able to sample both states gave Alanine's S1 bead the flexibility it needed to create room for NAG's G3 or ACN beads during the flipping process, thus reducing clashes and allowing for more flips to complete. These two changes, shown in Fig. 3.25, allowed the flip rate to jump up from 0.37 Flips/DU/µs to 3.05 in these intermediate simulations.

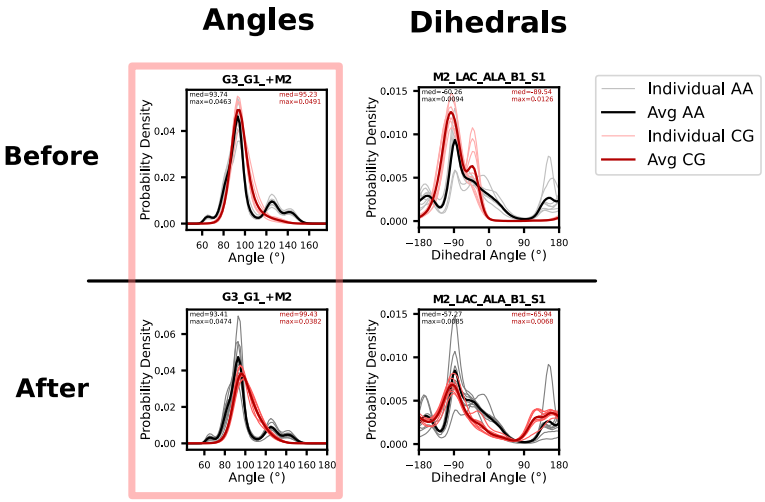

Figure 3.25: **Comparison of angle and dihedral distributions tweaked in the peptidoglycan model to accommodate for flips to occur.** Both right-most populations of both parameters had to be freely sampled to allow for room for flips to occur. This changes boosted the flip rate from 0.37 to 3.05 Flips/DU/µs.

3.10.2 Fixing End-to-End Problems

Transferring the glycan parameters to peptidoglycan was not the end of our parameterization effort. The main problem plaguing the parameterization of peptidoglycan were mismatches in the end-to-end distance. This problem arose from our CG peptidoglycan model getting stuck in an hairpin structure for far too long. While this structure did exist in the all-atom simulations, as shown in Fig. 3.26, it would leave this conformation in the scale of tens of ns. In contrast, the CG model would get stuck in this conformation and only leave it in the hundreds of ns scale. As an example, the longest hairpin structure in AA simulations lasted for ~40 ns, whilst the longest hairpin in CG was over 400 ns long.

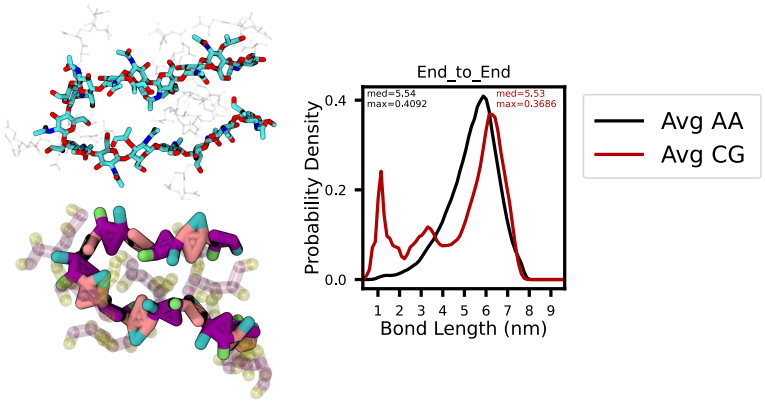

Figure 3.26: **VMD rendering of a simulation frame showcasing the AA (top left) and CG (bottom left) glycan backbone in a hairpin structure. Distribution of the monitored end-to-end distance in the peptidoglycan model after parameter transfer (right).** In the AA simulations, the hairpin state was only a fleeting state which disappeared in the tens of nanoseconds scale. In the CG simulations, this state lasted for longer periods of times which led to unrealistically prolonged short end-to-end distances, as shown by the distribution.

Many approaches were taken to try and fix this problem. We first thought of enforcing angles between the VS beads of all disaccharide rings to straighten the glycan chain structure. However, this would be incompatible with the AA distributions we would be recreating due to the angle between NAG's VS and its subsequent disaccharides (the *NAG_NAM_NAG* angle). As shown in Fig. 3.27, this angle has two well defined minima, each corresponding to one to the two previously identified states. As angles in our force field are defined by a harmonic potential, they can only have one well defined minima, making it impossible to faithfully recreate this angle, especially without interfering with the flips between states.

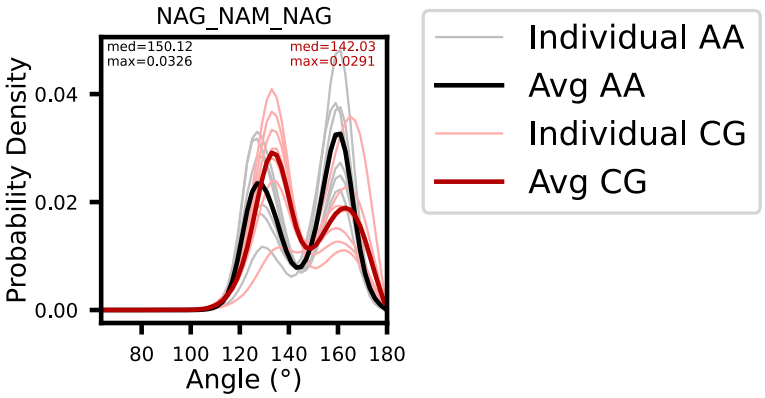

Figure 3.27: **Distribution of the monitored angle between the VS beads of consecutive sugars (starting at NAG).** This was one of the ways proposed to try and fix the hairpin problem in the CG model. Unfortunately, due to the harmonic nature of angles in the Martini 3 forcefield, it is impossible to correctly reproduce a bimodal distribution such as this one. This impossibility makes enforcing the *NAG_NAM_NAG* angle an invalid strategy for fixing the hairpin problem.

If a bonded interaction approach was impossible, we decided to try a non-bonded one. Here, we set up a scheme of repulsions between the VS beads of each sugar (i) and its subsequent monosaccharides at i+2 or i+3 distance. These repulsions were created using pairs with negative σ values (-2.30), which GROMACS interprets as a potential with no dispersive forces [36]. Using a very small ϵ value (0.001) creates a very sharp potential that is purely repulsive (Fig 3.28) and keeps the center of each sugar away from each other. As seen in Fig. 3.28, this approach did eliminate the hairpin problem and achieved a more linear structure. However, not only was this structure too straight, which failed to capture the more exact details of the end-to-end distance, it also heavily interfered with the possibility of flips from occurring. When flips occur the distance between sugar residues decreases and, by preventing this, the repulsion scheme brought the flip rate all the way down from 3.05 to 0.49 Flips/DU/μs. Given this problem, we decided to move forward with a new approach.

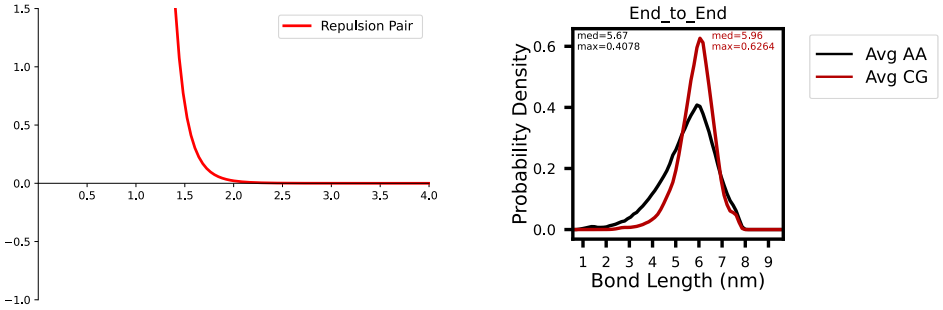

Figure 3.28: **Plotting of a purely repulsive Lennard-Jones potential with a -2.30 σ and 0.001 ϵ value (left). Distribution of the monitored end-to-end distance in the peptidoglycan model using the repulsive pair approach (right).** The second approach used to try and fix the hairpin problem in the CG model was using a repulsive pair potential to keep consecutive sugars apart. Although it was successful in preventing short end-to-end distances, the pair overcorrected for the fact and stunted the flip rate between states in the NAG-NAM disaccharide. This overcorrection led to a less faithful recreation of the end-to-end distance, as some intermediate end-to-end distances disappeared.

As the new variable in our simulations, we thought that the hairpin problem arose from too much self-interaction happening between peptide strands, leading to excessive aggregation and a short end-to-end distance. It has been pointed out before that IDP proteins need an increase in their interaction forces with the solvent to accurately capture their disordered behavior [66]. Given the lack of an overall structure due to the short nature of the peptide strand, it could be argued that peptidoglycan's peptide behaves similarly to an IDP. Thus, we decided to try to artificially increase the interactions between the peptide strand and water as a fix to the hairpin problem by mimicking the GōMartini 3 strategy [66]. This entailed creating a VS on top of the pre-existing backbone in each amino acid, which would exclusively and additively interact with water, boosting the attraction of each amino acid for the solvent. As per GōMartini 3 guidelines, these new beads were given a LJ potential exclusively for water with a σ value equal to the original

interaction and ϵ of 0.500 kJ/mol. Unfortunately, as shown in Fig. 3.29, this approach did not work as there is still excessive aggregation going on in the peptidoglycan CG simulations. While other simulations were ran with increasing ϵ values, they all resulted in mismatched end-to-end distributions as well.

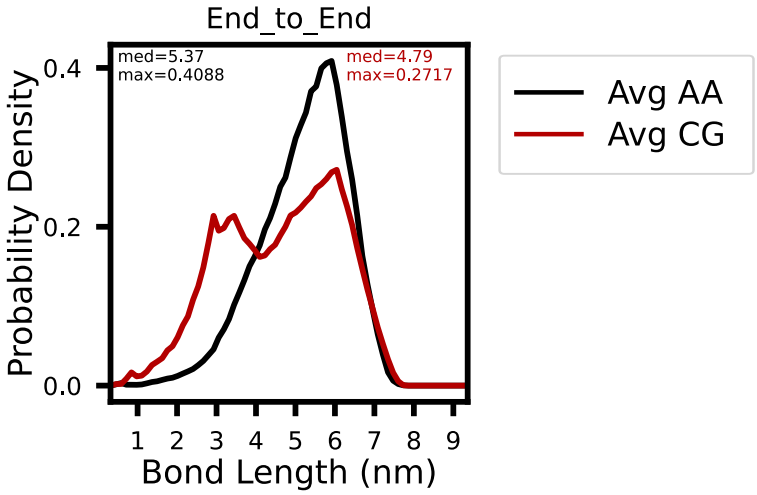

Figure 3.29: **Distribution of the monitored end-to-end distance in the peptidoglycan model using a GōMartini-like approach.** The new 1-body VS were placed on top of the preexisting amino acids backbone. The size of the backbone bead was matched and a LJ potential with water was attributed using a corresponding σ value and a ϵ of 0.500 kJ/mol.

Up until now, in all produced distributions both the individual and average distributions are plotted to show that the behavior of each individual parameter follows the same over-arching pattern. This possibility, however, is only a recent development. Beforehand, and, more specifically, while we were fixing the hairpin problem, only the average behavior was being plotted, which allowed for outliers to go unnoticed. Once the plotting script was updated to show both the average and individuals distributions, the problem with our model became apparent.

In all simulations where the end-to-end distance was off, there was a clear outlier in the individual distributions for the main dihedral controlling the NAM-NAG torsion, *M3_M1_+G2_+G3*. Back in the glycan parameterization process, the positive state of NAM-NAG was deliberately given more freedom than the distributions showed. Back then, this was not a problem as the torsion would quickly enter and leave this minority state due to its fast kinetics. However, when this parameter was directly translated to the peptidoglycan, we failed to account for the slower kinetics caused by the peptide strand. As a result, in all simulations with a end-to-end mismatch, there was also a NAM-NAG torsion stuck in the positive state for as long as the hairpin lasted, which caused the whole peptidoglycan structure to be stuck in the hairpin conformation. As the positive state in NAM-NAG's torsion is miniscule, for the sake of simplicity and consistency, we decided to completely remove its possibility by sculpting a new potential, which mostly fixed the hairpin problem.

Fig. 3.30 shows the final end-to-end distance as well as the change to the aforementioned dihedral. Even if rarer and reversible, the hairpin problem still persists, which means there is still some combinations of parameters that lead to an unrealistically short end-to-end distance. We believe there could also be some contribution coming from excessive aggregation in the default Martini 3 protein models, as was previously mentioned [66], which has yet to be tested with the new fix. Nonetheless, as our model will be mainly used in mesh-like structures, where the cross-links lock the glycan structure at higher end-to-end distances, this hairpin problem will be insignificant and unnoticeable in most use cases of the peptidoglycan model.

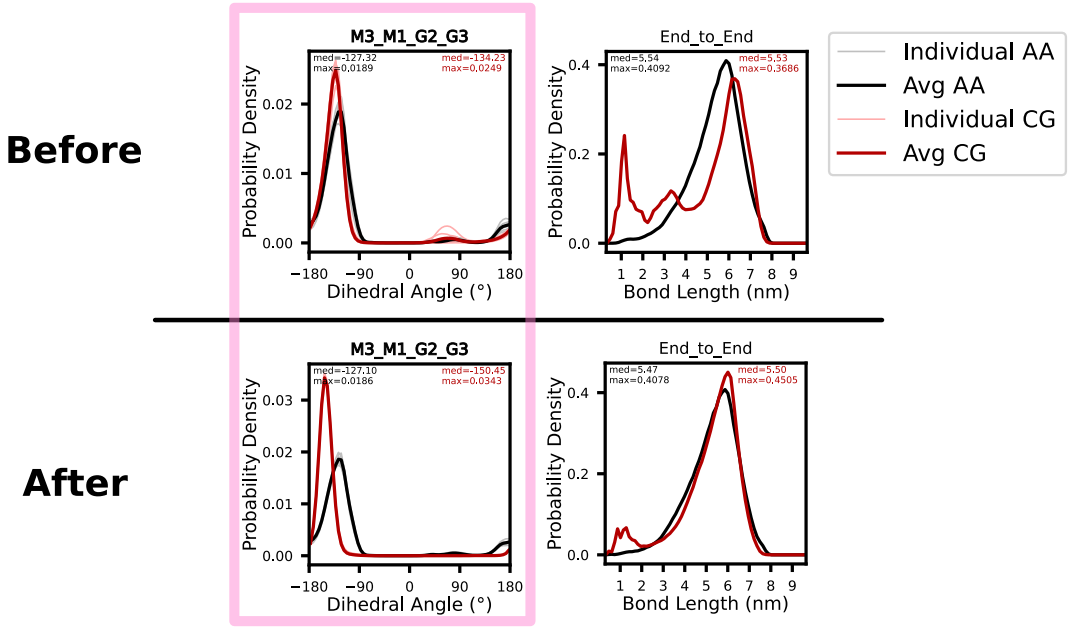

Figure 3.30: **Comparison of the distribution of the main torsion dihedral of NAM-NAG (left) before and after removing its "kinked" state. Comparison of the resulting end-to-end distance distributions (right).** Due to the slower kinetics of the peptidoglycan molecule, some NAM-NAG disaccharides were getting trapped for too long in the "kinked" state. This resulted in long-lasting hairpin conformations that contributed to prolonged short end-to-end distances in the distributions. Although they were still present, these hairpin conformations became much rarer, after removing the possibility of a "kinked" state from NAM-NAG.

The final results for the CG model's flip kinetics are shown in Fig. 3.31. We see that the Flip Rate values are higher in the CG model due to the reasons previously highlighted in section 3.6.3. Just like the glycan chain before it, both the AA and CG peptidoglycan chain spend about ~50% of the time in each state. Interestingly, the fraction of path used by flips seems to have switched fractions compared to the glycan, with the 0^o path being clearly preferred by the peptidoglycan chain, which the CG model was able to accurately recreate without any additional tweaking. Being able to capture this switch in the fractions without any additional tinkering in the exclusion-pair scheme speaks to its accuracy in being able to correctly capture the non-bonded interactions occurring around the NAG-NAM link.

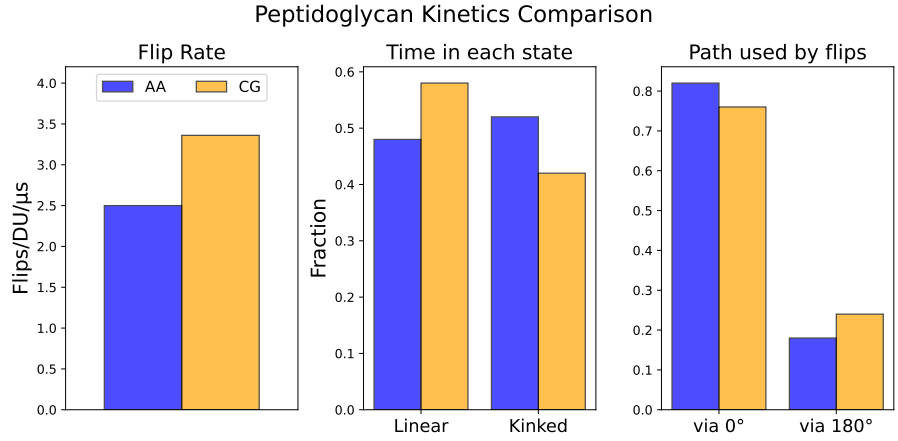

Figure 3.31: **Comparison of the final kinetic results from the AA and CG full peptidoglycan chain simulations** Only NAG-NAM flip kinetics were accounted for. We found there was good agreement in the kinetic results of all evaluated parameters.

3.10.3 Peptide Stem Rotation

The torsion angle between peptide stems was not analysed during the parameterization process of peptidoglycan as we were working with an unfinished version of the peptide parameters. Furthermore, disregarding the NAM-NAG link, an appropriately 50/50 chance of any NAG-NAM link being kinked means that the chance of a 8-mer peptidoglycan molecule being completely straight at any given point is only ~0.4%. Even if peptidoglycan is usually depicted as a linear structure, the AA simulations show that a free-roaming molecule in solution is very rarely completely straight for it to make sense to measure the angle between peptides. Nonetheless, if all bonded and non-bonded parameters capture the AA behavior, it follows that the angle between peptides will also be accurately recreated in the CG peptidoglycan simulations over time. Defining the angle between peptides is most important in the creation of mesh structures for simulation, not in the free-roaming simulations where this angle is not forced to a configuration by other links to other peptide stems.

In the end, we were able to create a peptidoglycan model capable of faithfully recreating both the average and individual distributions obtained from AA mapped simulations. Despite the bulkiness of the peptide stem introduced, only minimal changes had to be made to the bonded interactions. The exclusion and pair scheme was able to once again recreate the flip rate with no additional changes having to be made to it. Furthermore, the model was able to capture the switch that happened in the flip directionality preference without any further changes to achieve such a case. All of these results speak to the ability of the developed peptidoglycan model to capture both the bonded and non-bonded interactions of the peptidoglycan molecule. Nonetheless there are known problems relating to the end-to-end distance of the model in free peptidoglycan simulations, but the hairpin problem has become rarer and reversible. Furthermore, in peptidoglycan mesh simulations, the hairpin problem is also inconsequential as the cross-links do not

allow for such low end-to-end distance conformations to appear. Even so, options such as mixing the removal of NAM-NAG kinks with the GōMartini 3 mimic have yet to be tested that could finally lead to a perfect agreement in the end-to-end distances in the free peptidoglycan simulations.

3.11 Future Perspectives

The Ayappa group was able to use the parameters we developed to run a simulation with several peptidoglycan molecules that are not connected with cross-links. Using the data obtained from this simulation, cross-links were established based on the number of contacts made between peptide stems to create a mesh such as the one shown in Fig. 3.32. This approach was analogous to the Martini 2 parameterization steps of peptidoglycan.

While the constructed mesh simulates the cell wall of a Gram-negative bacteria (mono-layered and direct cross-links), the model can be easily adapted to recreate the cell wall of a Gram-positive bacteria. In this future work, the peptide stem will be switched for the appropriate amino acids, several layers will be introduced and a pentaglycine bridge will be used to establish cross-links.

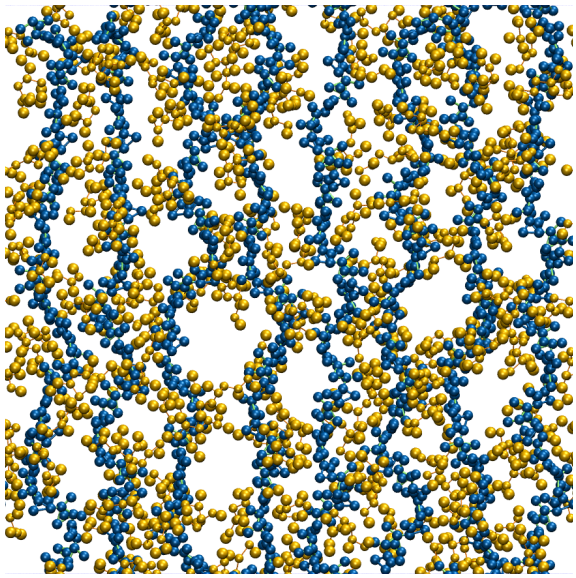

Figure 3.32: **Coarse-grain model of a peptidoglycan mesh.** 4-3 cross-links were established based on the frequency of contact in a free peptidoglycan simulation. After validation, this Gram-negative model of the bacterial cell wall can be used to run several tests. The sugar backbone is coloured in blue and the peptide stems in yellow.

Just as in the previous parameterization effort, the mesh model will be validated based on several properties, such as density distribution, cavity size distribution and potential mean force calculations using thymol and many others. These tests will gauge the accuracy of the peptidoglycan mesh model and allow for further studies into its properties. In the future, this mesh model can allow for further studies into drug permeability in bacteria, cell wall orientation studies, protein-cell wall interactions and many others.

Furthermore, the Martini 3 coarse-grain peptidoglycan parameters will be publicly available at The Melo Lab's GitHub. There, tools using PolyPly [48] and CGsmiles [60] will be available that generate a topology for a peptidoglycan molecule whose size depends on user input. Later on, this tool will be expanded using PolyPly's *gen_coords* [48] function to generate coordinates as well from the user's desired topology file.

Additionally, due to an overall lack of information, we plan on using NMR spectroscopy to study the biological relevance of the two identified torsional states in the NAG-NAM and NAM-NAG disaccharide units.

4

Conclusion

The bacterial cell wall plays a major role in all bacteria by helping maintain a well-defined cell shape and by protecting them from environmental effect. The main polymer behind the cell wall's properties is the peptidoglycan (PG).

Simulating these meshes could provide invaluable insight into their properties as well as allow for many different studies into the efficacy of potential novel drugs. However, the scale of these meshes makes coarse-grain (CG) molecular dynamics a more appropriate choice to carry out simulations, instead of the more detailed all-atom (AA) methods. In this work, we carried out the parameterization process of peptidoglycan for the Martini 3 CG force field.

In AA simulations of the peptidoglycan's glycan chain, two relevant ψ torsional states, which had a profound impact in the structure of the glycan backbone, were identified in the NAG-NAM disaccharide units. Replicating these flips between torsional states was one of the targets of our CG parameterization process, which sets it apart from most parameterization processes.

To create a CG structure, beads were precisely placed using a center of weights approach to ensure the best possible match in AA and CG solvent accessible surface area (SASA) by maintaining the discrepancy between AA and CG SASAs smaller than ~5%, as according to Martini 3 guidelines [54].

It was found that clashes between beads surrounding the linkage were responsible for the absence of flips between the identified torsional states in this initial CG model. We successfully utilized GROMACS' exclusion and pair interaction mechanisms to reintroduce flips into the CG model, while also matching the AA flip rate, directionality and the fraction of time spent in each state. When transferred to the entire glycan chain, these new parameters were also successfully capable of reintroducing flips into the CG glycan model.

While minimal adjustments were required to account for the restraints imposed by the peptide stem on the glycan chain, specific dihedrals needed refinement to reflect the increased difficulty of transitions. Ultimately, the CG peptidoglycan model successfully reproduced the AA flip rate, directionality, and the fraction of time spent in each state.

Although the overall AA distributions were matched, the parameterization of the peptidoglycan model was hindered by a short end-to-end distance caused by a metastable hairpin structure. Initial attempts to resolve this issue, including introducing inter-sugar repulsions and adopting a strategy similar to GoMartini 3, were unsuccessful. In the end, the main torsional dihedral in NAM-NAG had to be adjusted to prevent individual disaccharide units in the peptidoglycan chain from becoming trapped in certain conformations.

While this fix significantly reduced the persistence of the hairpin structure and made it reversible, the CG model's end-to-end distance still does not perfectly match the AA model. Even so, as this hairpin conformation was deemed inconsequential in peptidoglycan mesh simulations, due to cross-links not allowing for this shorter end-to-end distances to appear, we deemed the parameterization of peptidoglycan finished for now.

The Ayappa group was able to use the parameters we developed to simulate peptidoglycan molecules without cross-links, using contact data between peptide stems to establish cross-links and create a mesh model resembling the Gram-negative bacterial cell wall. This model, featuring a monolayer structure with direct cross-links, can be adapted for Gram-positive bacteria by modifying the peptide stem, adding layers, and introducing pentaglycine bridges.

In the end, our parameterization effort was one of the first in Martini 3 to take into account and correctly capturing the kinetics of intramolecular dynamics. In the future, the peptidoglycan model can be used for simulations testing diverse properties of the bacterial cell wall such as drug permeability, cell wall orientation studies, protein-cell wall interactions and many others.

Bibliography

[1] J. M. Lourenço. *The NOVAthesis LaTeX Template User's Manual*. NOVA University Lisbon. 2021. URL: https://github.com/joaomlourenco/novathesis/raw/main/template.pdf (cit. on p. i).

[2] C. Mayer et al. "Bacteria's different ways to recycle their own cell wall". In: *International Journal of Medical Microbiology* 309.7 (2019), p. 151326. ISSN: 1438-4221. DOI: https://doi.org/10.1016/j.ijmm.2019.06.006. URL: https://www.sciencedirect.com/science/article/pii/S1438422119301936 (cit. on p. 1).

[3] S. Garde, P. K. Chodisetti, and M. Reddy. "Peptidoglycan: Structure, Synthesis, and Regulation". In: *EcoSal Plus* 9.2 (2021). DOI: 10.1128/ecosalplus.ESP-0010-2020. eprint: https://journals.asm.org/doi/pdf/10.1128/ecosalplus.esp-0010-2020. URL: https://journals.asm.org/doi/abs/10.1128/ecosalplus.esp-0010-2020 (cit. on pp. 1–4).

[4] M. Pazos and K. Peters. "Peptidoglycan". en. In: *Subcell. Biochem.* 92 (2019), pp. 127–168 (cit. on pp. 1–6).

[5] A. Mai-Prochnow et al. "Gram positive and Gram negative bacteria differ in their sensitivity to cold plasma". en. In: *Sci. Rep.* 6.1 (2016-12) (cit. on p. 1).

[6] D. L. Nelson and M. Cox. *Lehninger principles of biochemistry*. 8th ed. New York, NY: W.H. Freeman, 2021-03 (cit. on p. 2).

[7] W. Vollmer, D. Blanot, and M. A. De Pedro. "Peptidoglycan structure and architecture". In: *FEMS Microbiology Reviews* 32.2 (2008), pp. 149–167. DOI: https://doi.org/10.1111/j.1574-6976.2007.00094.x. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1574-6976.2007.00094.x. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1574-6976.2007.00094.x (cit. on p. 2).

[8] A. J. Apostolos, S. E. Pidgeon, and M. M. Pires. "Remodeling of cross-bridges controls peptidoglycan cross-linking levels in bacterial cell walls". en. In: *ACS Chem. Biol.* 15.5 (2020-05), pp. 1261–1267 (cit. on pp. 3, 6).

[9] S. Porfírio, R. W. Carlson, and P. Azadi. "Elucidating Peptidoglycan Structure: An Analytical Toolset". In: *Trends in Microbiology* 27.7 (2019), pp. 607–622. ISSN: 0966-842X. DOI: https://doi.org/10.1016/j.tim.2019.01.009. URL: https://www.sciencedirect.com/science/article/pii/S0966842X19300228 (cit. on p. 4).

[10] S. J. Kim, J. Chang, and M. Singh. "Peptidoglycan architecture of Gram-positive bacteria by solid-state NMR". en. In: *Biochim Biophys Acta* 1848.1 Pt B (2014-06), pp. 350–362 (cit. on pp. 4, 5).

[11] S. O. Meroueh et al. "Three-dimensional structure of the bacterial cell wall peptidoglycan". en. In: *Proc. Natl. Acad. Sci. U. S. A.* 103.12 (2006-03), pp. 4404–4409 (cit. on pp. 4, 5).

[12] M. V. Kelemen and H. J. Rogers. "Three-dimensional molecular models of bacterial cell wall mucopeptides (peptidoglycans)". en. In: *Proc. Natl. Acad. Sci. U. S. A.* 68.5 (1971-05), pp. 992–996 (cit. on p. 4).

[13] J. C. Gumbart et al. "Escherichia coli peptidoglycan structure and mechanics as predicted by atomic-scale simulations". en. In: *PLoS Comput Biol* 10.2 (2014-02), e1003475 (cit. on pp. 5, 8, 26).

[14] J. D. Durrant and J. A. McCammon. "Molecular dynamics simulations and drug discovery". en. In: *BMC Biol.* 9.1 (2011-12), p. 71 (cit. on p. 6).

[15] R. Vaiwala et al. "Developing a Coarse-Grained Model for Bacterial Cell Walls: Evaluating Mechanical Properties and Free Energy Barriers". In: *Journal of Chemical Theory and Computation* 16.8 (2020). PMID: 32628849, pp. 5369–5384. DOI: 10.1021/acs.jctc.0c00539. URL: https://doi.org/10.1021/acs.jctc.0c00539 (cit. on pp. 6, 8, 9, 26, 27, 33).

[16] S. A. Hollingsworth and R. O. Dror. "Molecular Dynamics Simulation for All". In: *Neuron* 99.6 (2018), pp. 1129–1143. ISSN: 0896-6273. DOI: https://doi.org/10.1016/j.neuron.2018.08.011. URL: https://www.sciencedirect.com/science/article/pii/S0896627318306846 (cit. on pp. 6, 7).

[17] A. R. Leach. *Molecular modelling : principles and applications*. eng. 2nd ed. Harlow, England ; New York: Prentice Hall, 2001. ISBN: 0582382106 (cit. on pp. 6, 7, 11–13, 16, 18–21).

[18] L. Borges-Araújo et al. "Pragmatic Coarse-Graining of Proteins: Models and Applications". In: *Journal of Chemical Theory and Computation* 19.20 (2023). PMID: 37788237, pp. 7112–7135. DOI: 10.1021/acs.jctc.3c00733. eprint: https://doi.org/10.1021/acs.jctc.3c00733. URL: https://doi.org/10.1021/acs.jctc.3c00733 (cit. on pp. 7, 13).

[19] P. C. T. Souza et al. "Martini 3: a general purpose force field for coarse-grained molecular dynamics". In: *Nature Methods* 18.4 (2021-04), pp. 382–388. ISSN: 1548-7105. DOI: 10.1038/s41592-021-01098-3. URL: https://doi.org/10.1038/s41592-021-01098-3 (cit. on pp. 7–9, 13, 23, 24, 30).

[20] F. Grünewald et al. "Martini 3 Coarse-Grained Force Field for Carbohydrates". In: *Journal of Chemical Theory and Computation* 18.12 (2022). PMID: 36342474, pp. 7555–7569. DOI: 10.1021/acs.jctc.2c00757. eprint: https://doi.org/10.1021/acs.jctc.2c00757. URL: https://doi.org/10.1021/acs.jctc.2c00757 (cit. on pp. 7, 24, 28, 30, 34–36, 38).

[21] L. Borges-Araújo et al. "Assessing the Martini 3 protein model: A review of its path and potential". In: *Biochimica et Biophysica Acta (BBA) - Proteins and Proteomics* 1872.4 (2024), p. 141014. ISSN: 1570-9639. DOI: https://doi.org/10.1016/j.bbapap.2024.141014. URL: https://www.sciencedirect.com/science/article/pii/S1570963924000219 (cit. on p. 7).

[22] L. Borges-Araújo et al. "Martini 3 Coarse-Grained Force Field for Cholesterol". In: *Journal of Chemical Theory and Computation* 19.20 (2023). PMID: 37796943, pp. 7387–7404. DOI: 10.1021/acs.jctc.3c00547 (cit. on p. 7).

[23] L. Borges-Araújo et al. "Improved Parameterization of Phosphatidylinositide Lipid Headgroups for the Martini 3 Coarse-Grain Force Field". In: *Journal of Chemical Theory and Computation* 18.1 (2022). PMID: 34962393, pp. 357–373. DOI: 10.1021/acs.jctc.1c00615 (cit. on p. 7).

[24] R. Alessandri et al. "Martini 3 Coarse-Grained Force Field: Small Molecules". In: *Advanced Theory and Simulations* 5.1 (2022), p. 2100391. DOI: https://doi.org/10.1002/adts.202100391 (cit. on pp. 7, 30, 36).

[25] J. A. Stevens et al. "Molecular dynamics simulation of an entire cell". In: *Frontiers in Chemistry* 11 (2023). ISSN: 2296-2646. DOI: 10.3389/fchem.2023.1106495. URL: https://www.frontiersin.org/journals/chemistry/articles/10.3389/fchem.2023.1106495 (cit. on p. 7).

[26] R. Alessandri et al. "Pitfalls of the Martini Model". In: *Journal of Chemical Theory and Computation* 15.10 (2019). PMID: 31498621, pp. 5448–5460. DOI: 10.1021/acs.jctc.9b00473. eprint: https://doi.org/10.1021/acs.jctc.9b00473. URL: https://doi.org/10.1021/acs.jctc.9b00473 (cit. on p. 7).

[27] W. F. van Gunsteren et al. "On the Effect of the Various Assumptions and Approximations used in Molecular Simulations on the Properties of Bio-Molecular Systems: Overview and Perspective on Issues". In: *ChemPhysChem* 22.3 (2021), pp. 264–282. DOI: https://doi.org/10.1002/cphc.202000968. eprint: https://chemistry-europe.onlinelibrary.wiley.com/doi/pdf/10.1002/cphc.202000968. URL: https://chemistry-europe.onlinelibrary.wiley.com/doi/abs/10.1002/cphc.202000968 (cit. on p. 11).

[28] Juan José Nogueira. *Molecular dynamics - chapter 2: Force fields*. en (cit. on pp. 11, 12).

[29] L. Monticelli and D. P. Tieleman. "Force fields for classical molecular dynamics". en. In: *Methods Mol Biol* 924 (2013), pp. 197–213 (cit. on pp. 12–16).

[30] González, M.A. "Force fields and molecular dynamics simulations". In: *JDN* 12 (2011), pp. 169–200. DOI: 10.1051/sfn/201112009. URL: https://doi.org/10.1051/sfn/201112009 (cit. on p. 12).

[31] H. M. Waidyasooriya, M. Hariyama, and K. Kasahara. "An FPGA Accelerator for Molecular Dynamics Simulation Using OpenCL". In: *International Journal of Networked and Distributed Computing* 5.1 (2017-01), pp. 52–61. ISSN: 2211-7946. DOI: 10.2991/ijndc.2017.5.1.6. URL: https://doi.org/10.2991/ijndc.2017.5.1.6 (cit. on p. 12).

[32] F.-Y. Lin and A. D. MacKerell Jr. "Force Fields for Small Molecules". en. In: *Methods Mol Biol* 2022 (2019), pp. 21–54 (cit. on p. 13).

[33] K. Vanommeslaeghe et al. "CHARMM general force field: A force field for drug-like molecules compatible with the CHARMM all-atom additive biological force fields". en. In: *J. Comput. Chem.* 31.4 (2010-03), pp. 671–690 (cit. on p. 13).

[34] J. Wang et al. "Development and testing of a general amber force field". In: *Journal of Computational Chemistry* 25.9 (2004), pp. 1157–1174. DOI: https://doi.org/10.1002/jcc.20035. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/jcc.20035. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/jcc.20035 (cit. on p. 13).

[35] A. Mirzanejad and S. A. Varganov. "Derivation of Morse potential". In: *Molecular Physics* 0.0 (2024), e2360542. DOI: 10.1080/00268976.2024.2360542. eprint: https://doi.org/10.1080/00268976.2024.2360542. URL: ttps://doi.org/10.1080/00268976.2024.2360542 (cit. on p. 13).

[36] M. Abraham et al. *GROMACS 2024.2 Manual*. Version 2024.2. 2024-05. DOI: 10.5281/zenodo.11148638. URL: https://doi.org/10.5281/zenodo.11148638 (cit. on pp. 14, 16–23, 26, 27, 59).

[37] M. Bulacu et al. "Improved angle potentials for coarse-grained molecular dynamics simulations". en. In: *J. Chem. Theory Comput.* 9.8 (2013-08), pp. 3282–3292 (cit. on pp. 14, 15, 17).

[38] S. J. Marrink et al. "The MARTINI Force Field: Coarse Grained Model for Biomolecular Simulations". In: *The Journal of Physical Chemistry B* 111.27 (2007-07), pp. 7812–7824. ISSN: 1520-6106. DOI: 10.1021/jp071097f. URL: https://doi.org/10.1021/jp071097f (cit. on p. 14).

[39] C. Tan et al. "A singularity-free torsion angle potential for coarse-grained molecular dynamics simulations". In: *The Journal of Chemical Physics* 153.4 (2020-07), p. 044110. ISSN: 0021-9606. DOI: 10.1063/5.0013089. eprint: https://pubs.aip.org/aip/jcp/article-pdf/doi/10.1063/5.0013089/14721726/044110_1_online.pdf. URL: https://doi.org/10.1063/5.0013089 (cit. on p. 15).

[40] P. Atkins et al. *Physical chemistry for the life sciences*. en. 3rd ed. London, England: Oxford University Press, 2023-03 (cit. on p. 16).

[41] K. Roy, S. Kar, and R. N. Das. "Chapter 5 - Computational Chemistry". In: *Understanding the Basics of QSAR for Applications in Pharmaceutical Sciences and Risk Assessment*. Ed. by K. Roy, S. Kar, and R. N. Das. Boston: Academic Press, 2015, pp. 151–189. ISBN: 978-0-12-801505-6. DOI: https://doi.org/10.1016/B978-0-12-801505-6.00005-3. URL: https://www.sciencedirect.com/science/article/pii/B9780128015056000053 (cit. on pp. 18, 20).

[42] E. Braun et al. "Best practices for foundations in molecular simulations [article v1.0]". en. In: *Living J. Comput. Mol. Sci.* 1.1 (2019) (cit. on pp. 18, 21).

[43] A. T. Gisbert and N. Piovella. "Multimode Collective Atomic Recoil Lasing in Free Space". In: *Atoms* 8 (2020-12), p. 93. DOI: 10.3390/atoms8040093 (cit. on p. 18).

[44] W. F. V. Gunsteren and H. J. C. Berendsen. "A Leap-frog Algorithm for Stochastic Dynamics". In: *Molecular Simulation* 1.3 (1988), pp. 173–185. DOI: 10.1080/08927028808080941. eprint: https://doi.org/10.1080/08927028808080941. URL: https://doi.org/10.1080/08927028808080941 (cit. on p. 19).

[45] M. S. Smyth and J. H. Martin. "X ray crystallography". en. In: *Mol. Pathol.* 53.1 (2000-02), pp. 8–14 (cit. on p. 19).

[46] D. Marion. "An introduction to biological NMR spectroscopy". en. In: *Mol. Cell. Proteomics* 12.11 (2013-11), pp. 3006–3025 (cit. on p. 19).

[47] J. Jumper et al. "Highly accurate protein structure prediction with AlphaFold". en. In: *Nature* 596.7873 (2021-08), pp. 583–589 (cit. on p. 19).

[48] F. Grünewald et al. "Polyply; a python suite for facilitating simulations of macromolecules and nanomaterials". In: *Nature Communications* 13.1 (2022-01), p. 68. ISSN: 2041-1723. DOI: 10.1038/s41467-021-27627-4. URL: https://doi.org/10.1038/s41467-021-27627-4 (cit. on pp. 19, 26, 29, 35, 64).

[49] E. Garcia. "End-functionalized rodlike colloid suspensions under shear flow". PhD thesis. 2013-10. DOI: 10.13140/2.1.3032.7363 (cit. on p. 22).

[50] A. Laio and M. Parrinello. "Escaping free-energy minima". en. In: *Proc. Natl. Acad. Sci. U. S. A.* 99.20 (2002-10), pp. 12562–12566 (cit. on p. 23).

[51] A. Laio and F. Gervasio. "Metadynamics: A method to simulate rare events and reconstruct the free energy in biophysics, chemistry and material science". In: *Reports on Progress in Physics* 71 (2008-12), p. 126601. DOI: 10.1088/0034-4885/71/12/126601 (cit. on p. 23).

[52] Henar Mateo de la Fuente. *Enhanced Sampling Methods - Chapter 4: Metadynamics*. en (cit. on p. 23).

[53] J. Clayton, L. Baweja, and J. Wereszczynski. "Peptide Dynamics and Metadynamics: Leveraging Enhanced Sampling Molecular Dynamics to Robustly Model Long-Timescale Transitions". In: *Computational Peptide Science: Methods and Protocols*. Ed. by T. Simonson. New York, NY: Springer US, 2022, pp. 151–167. ISBN: 978-1-0716-1855-4. DOI: 10.1007/978-1-0716-1855-4_8. URL: https://doi.org/10.1007/978-1-0716-1855-4_8 (cit. on p. 24).

[54] R. Alessandri. *Parametrization of a new small molecule*. URL: https://cgmartini.nl/docs/tutorials/Martini3/Small_Molecule_Parametrization/ (cit. on pp. 25, 26, 38, 65).

[55] H. I. Ingólfsson et al. "The power of coarse graining in biomolecular simulations". In: *WIREs Computational Molecular Science* 4.3 (2014), pp. 225–248. DOI: https://doi.org/10.1002/wcms.1169. eprint: https://wires.onlinelibrary.wiley.com/doi/pdf/10.1002/wcms.1169. URL: https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/wcms.1169 (cit. on p. 25).

[56] R. Alessandri. *Free energy techniques*. URL: https://cgmartini.nl/docs/tutorials/Legacy/martini2/free_energy.html (cit. on pp. 25, 26).

[57] M. J. Abraham et al. "GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers". In: *SoftwareX* 1-2 (2015), pp. 19–25. ISSN: 2352-7110. DOI: https://doi.org/10.1016/j.softx.2015.06.001. URL: https://www.sciencedirect.com/science/article/pii/S2352711015000059 (cit. on p. 26).

[58] PLUMED consortium. "Promoting transparency and reproducibility in enhanced molecular simulations". en. In: *Nat. Methods* 16.8 (2019-08), pp. 670–673 (cit. on p. 26).

[59] G. A. Tribello, M. Bonomi, G. Bussi, C. Camilloni et al. *PLUMED Tutorials: a collaborative, community-driven learning ecosystem*. URL: https://www.plumed-tutorials.org/ (cit. on p. 26).

[60] F. Grünewald. *CGsmiles*. https://github.com/gruenewald-lab/CGsmiles/branches/all. 2024 (cit. on pp. 26, 64).

[61] T. Lütteke. "Analysis and validation of carbohydrate three-dimensional structures". en. In: *Acta Crystallogr. D Biol. Crystallogr.* 65.Pt 2 (2009-02), pp. 156–168 (cit. on p. 28).

[62] N. Michaud-Agrawal et al. "MDAnalysis: a toolkit for the analysis of molecular dynamics simulations". en. In: *J Comput Chem* 32.10 (2011-04), pp. 2319–2327 (cit. on p. 29).

[63] C. R. Harris et al. "Array programming with NumPy". In: *Nature* 585.7825 (2020-09), pp. 357–362. DOI: 10.1038/s41586-020-2649-2. URL: https://doi.org/10.1038/s41586-020-2649-2 (cit. on p. 29).

[64] P. Virtanen et al. "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python". In: *Nature Methods* 17 (2020), pp. 261–272. DOI: 10.1038/s41592-019-0686-2 (cit. on p. 29).

[65] J. D. Hunter. "Matplotlib: A 2D graphics environment". In: *Computing in Science & Engineering* 9.3 (2007), pp. 90–95. DOI: 10.1109/MCSE.2007.55 (cit. on p. 29).

[66] P. C. T. Souza et al. "GoMartini 3: From large conformational changes in proteins to environmental bias corrections". In: *bioRxiv* (2024). DOI: 10.1101/2024.04.15.589479. eprint: https://www.biorxiv.org/content/early/2024/04/16/2024.04.15.589479.full.pdf. URL: https://www.biorxiv.org/content/early/2024/04/16/2024.04.15.589479 (cit. on pp. 59, 61).

A

Terminal Residue Analysis

The ~400 ns long mapped glycan simulation was used to create these plots. In all of the analysed parameters, there was not a significant enough difference for the terminal distributions to justify the parameterization of specific terminal residues.

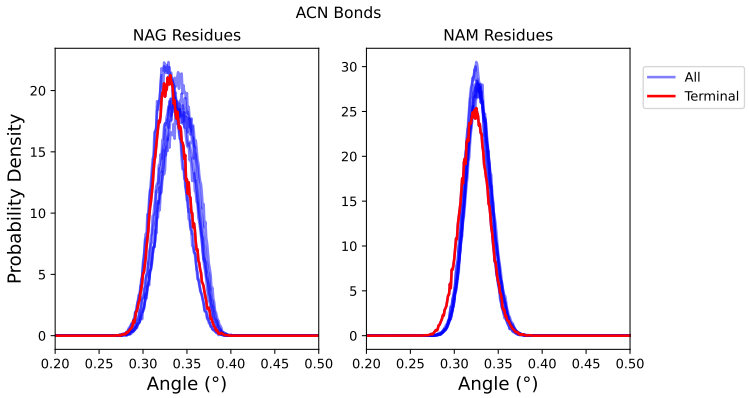

Figure A.1: **Comparison of the bond controlling ACN functional group in both monosaccharides.**

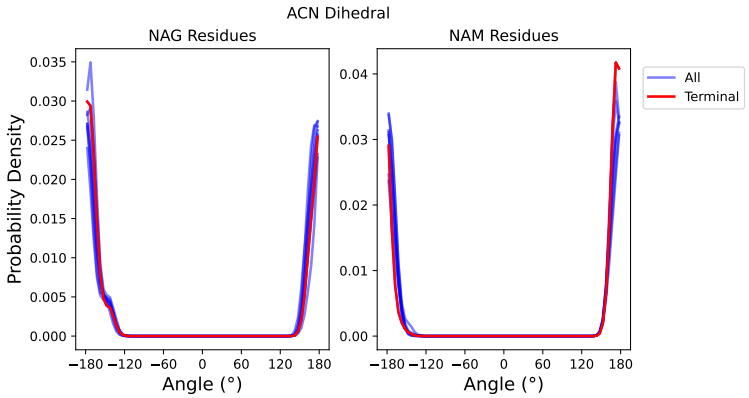

Figure A.2: **Comparison of the dihedral controlling ACN functional group in both monosaccharides.**

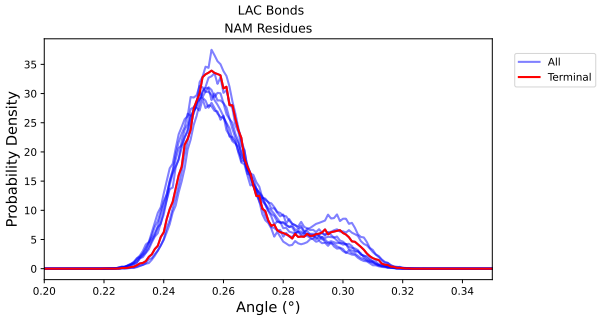

Figure A.3: **Comparison of the bond controlling LAC functional group in the NAM monosaccharides.**

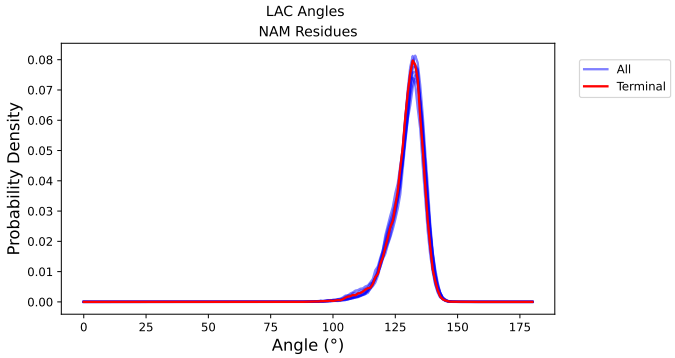

Figure A.4: **Comparison of the angle controlling LAC functional group in the NAM monosaccharides.**

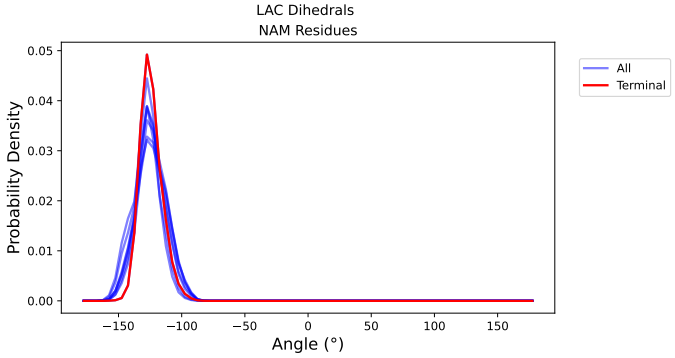

Figure A.5: **Comparison of the dihedral controlling LAC functional group in the NAM monosaccharides.**

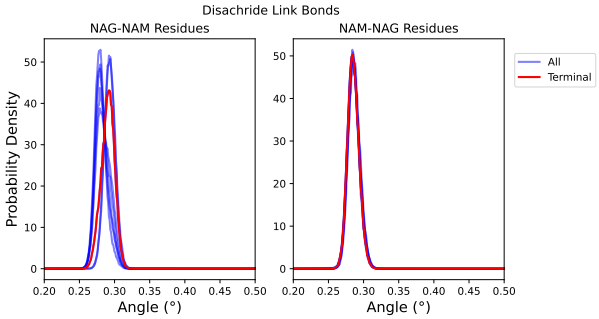

Figure A.6: **Comparison of the bond controlling the link between both disaccharide units.**

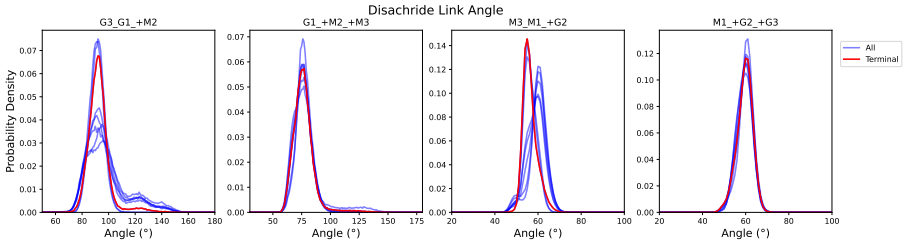

Figure A.7: **Comparison of the angles controlling the link between both disaccharide units.**

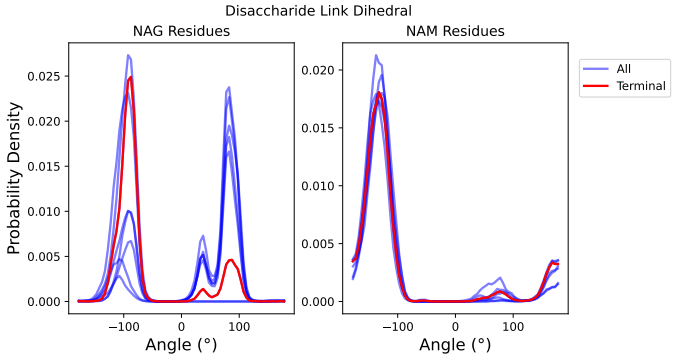

Figure A.8: **Comparison of the main torsional dihedral controlling the link between both disaccharide units.**

B

Exclusion and Pair Scheme

Table B.1: **Exclusions used in the exclusion and pair scheme for the NAG-NAM disaccharide.**

Bead 1	Bead 2
G1	M3 , M4 , LAC
G3	M2 , M3 , LAC
G4	M2
ACN	M3 , LAC

Table B.2: **Pairs used in the exclusion and pair scheme for the NAG-NAM disaccharide.** Epsilon values were unchanged.

Bead 1	Bead 2	Default σ	New σ
G1	LAC	0.43	0.39
G3	M3	0.41	0.355
G3	LAC	0.43	0.39
ACN	M3	0.41	0.39
ACN	LAC	0.43	0.42

Table B.3: **Exclusions used in the exclusion and pair scheme for the NAM-NAG disaccharide.**

Bead 1	Bead 2
M1	G3 , G4
M3	G2 , G3
M4	M2
ACN	G3

Table B.4: **Pairs used in the exclusion and pair scheme for the NAM-NAG disaccharide.** Epsilon values were unchanged.

Bead 1	Bead 2	Default σ	New σ
ACN	M3	0.41	0.39

UNIVERSIDADE NOVA
DE LISBOA