From Molecular-Ion Interactions to Gel Properties: A Combined Computational–Experimental Investigation of Alginates

Angharad Elizabeth Wood

This thesis is submitted for the degree of
Doctor of Philosophy

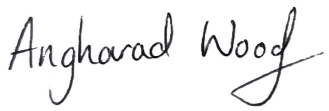
Department of Chemical and Process Engineering
University of Strathclyde

2025

Declaration

This thesis is the result of the author's original research. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree.

Signed:



Date: 29-09-25

Acknowledgements

My first and biggest thank you goes to Dr Karen Johnston, my primary supervisor in this PhD. I cannot tell you how much I appreciated your guidance, consistent availability, and effort in keeping me on track each time I got lost in a rabbit-hole of research, during this whole project. I would also like to thank Dr Paul Mulheran who aided in this effort and was always ready with kind words when I was struggling.

My appreciation also goes to Dr Charlie Bavington, Dr Ian Vallance, and Dr Vitor Magueijo, who helped in creating the storyline that this thesis follows, and were endlessly helpful in guiding my experimental work. To the staff at IBioIC, thank you for always being available for a chat, and for encouraging me throughout this project.

A special thanks goes to Elliot and Nia, my biggest supporters, who championed through all the highs and lows of this project with me. I dread to think what state I would have been in during this PhD without you both. To Liam and Naomi, thank you for being honest about what a PhD entails, and for being very supportive even when I had ignored your advice. To Mum and Dad, thank you for always encouraging me to pursue my aspirations and allowing me the freedom to choose them; I couldn't have done this without you.

I'd like to mention my friends and colleagues at the university and OCEANIUM®, who made life at work a lot more fun and showed me I wasn't alone in my struggles. Special thanks go to Thomas, Aidan, Michaela, and Alfie for this. To all my friends and family, in- and outside of work, thank you so much for your support and guidance.

My final acknowledgement goes to Mr Williams, my secondary school chemistry teacher, who will probably never see this, but deserves an acknowledgement anyway, as all outstanding teachers do.

Abstract

Sodium alginate is a sustainable polymer that gels on contact with calcium ions. Alginate is a polysaccharide composed of two monomers: α-L-guluronate (G) and β-D-mannuronate (M). The aim of this thesis is to investigate the impact of monomer type and cation environment on alginate behaviour. Molecular dynamics was used to investigate the effects of these factors on alginate trimers in solution. G trimers formed more buckled structures than M, as evidenced by the 5-8 % increase in end-to-end distances. These values decreased in systems with calcium ions by as much as 3 %. In systems with calcium ions, mixing of G and M trimers resulted in a 550 % increase in trimer clustering compared to pure G or M systems, as shown by the number and size of clusters that formed. To extend this work, the G:M ratio and cation environment were examined experimentally. A design of experiments study investigated the effects of different curing conditions on calcium alginate gel strength. Introduction of excess sodium ions had a significant ($p<0.05$) effect on calcium alginate gel strength, inducing an average decrease of 30.7 %. To examine this effect further, molecular dynamics was used to compare 24-monomer G and M chains in stoichiometric and excess sodium ion environments. Analysis of the radius of gyration and chain-chain radial distribution functions showed that an increase in sodium ion concentration created more open G structures with more G-G interaction and more closed M structures with less M-M interactions. Systems with low sodium ion content presented with more favourable G-M interactions, compared with G-G or M-M, as shown in the distribution functions. This work has shown that excess sodium ion content and blended G/M polymer interactions, which have previously been overlooked in alginate research, contribute extensively to sodium and calcium alginate properties.

Contents

List of Figures

List of Figures

List of Tables

List of Tables

Chapter 1

Introduction

1.1 Plastics in daily life: virtues and vices

Plastics are one of the most widely used materials in production today, with some marking this era as the "Plastic Age"[2]. First manufactured in the 1860s, the annual production of plastic is continually increasing, reaching over 400 million tonnes in 2023[3].

There are seven main types of plastics that see the most use worldwide: polyethylene terephthalate (PET), high-density polyethylene (HDPE), polyvinyl chloride (PVC), low-density polyethylene (LDPE), polypropylene (PP), polystyrene (PS), and "other" which includes polycarbonates[4]. The prefix "poly" in all these names refers to the large number of repeating units which make up the molecules of a plastic. These molecules are, more generally, called polymers.

Polymers are long-chain molecules composed of repeating units called monomers. There is a multitude of ways in which the properties of a plastic can be altered: (i) using different polymer lengths; (ii) combining different monomer types to form copolymers, where two or more monomers are bonded together to form a combined polymer, or blended polymers, in which the individual polymer types are mixed; (iii) introducing additives such as fillers, which strengthen a polymer network, or plasticisers, which increase flexibility; and (iv) through external processes such as heating or mechanical manipulation[5, 6].

The ability to alter the properties of the polymer system, of which a plastic is

composed, gives them numerous benefits as materials and makes them preferable over other materials to manufacturers. Some of these beneficial properties include that plastics are commonly easily processed, light-weight, tough, flame-retardant, insulating, readily produced in a range of colours and transparencies, and weather resistant[7]. While more traditional materials such as metal, wood, and glass may each have a few of these advantageous properties, plastics can combine them all.

The preference for plastic can extend to the consumer. It has been shown multiple times in multiple cultures that consumers prefer plastic packaging and bags over alternatives for the same reasons as manufacturers[8–10]. Additional advantages reported by consumers include convenience, option for resealability, and hygienic properties[8–10].

Despite all these advantages, anti-plastic campaigns are growing, with many organisations such as Greenpeace, the UN Environment Programme, the European Environment Agency, and the WWF advocating for a reduction or removal of plastics from our every day lives[11–16]. The basis of these aims lies within one of the coveted advantages of plastic products: their durability. Plastics are so resilient against degradation that they can exist in nature for hundreds of years without ever fully degrading. They instead form smaller and smaller particles called microplastics. In 2023, less than 9 % of commercial plastics were recycled into new products, meaning that plastic waste most commonly ends up in landfill, or discarded into nature. Microplastics have been found at 8 km above and 10 km below sea level, on Mount Everest and in the Mariana Trench which shows the extensive reach of these pollutants and how quickly they have dispersed into isolated environments since their manufacture less than 200 years ago[17, 18].

The start of a plastic's life can be just as problematic as its end: in 2023 over 90 % of plastics were petrochemically derived, i.e. they are produced from fossil fuels, a finite resource[3]. Estimates on how much fossil fuel is left give only decades before there is no option but to find an alternative.

It is vital, therefore, that alternatives to plastic production, waste management, and products themselves are found and optimised.

1.2 Moving away from fossil fuels

As a response to the United States of America's Pollution Prevention Act 1990, and the subsequent rise in awareness of eliminating waste by improved design, instead of through treatment and disposal, Paul T Anastas and John C Warner published their twelve principles of "Green Chemistry" in 1998[19]. While all the principles should be applied to any manufacturing process, some are of particular interest when discussing plastics: (i) use of renewable feedstocks in place of finite raw materials, and (ii) design for degradation so that the product and its degradation products are not toxic, bio-accumulative, or environmentally persistent.

Following the principles of "Green Chemistry", research has become increasing prevalent in finding alternative plastic production methods and products. This research takes many forms. Recycling methods now include mechanical, chemical, and biological for the seven most common plastic types with much discussion on the advantages and disadvantages of each and how they can be improved[20–22]. The common plastic types are also being partly or wholly produced from renewable sources, such as through bacterial excretion or from processing glucose from sugar cane[20, 23].

Another way in which research is driving the replacement of fossil fuels as a plastic feedstock is by replacing commonly used plastics altogether. With the aim of replacing petrochemical plastics, polymers found naturally in the environment are being studied. These polymers are varied in their abundance and sources. Cellulose is the most abundant organic polymer in nature and is found in aquatic and terrestrial plants and provides most of the structural integrity in the latter[24, 25]. Chitosan is found in the shells of crabs, lobsters, prawns, and less commonly, in the walls of some fungus species[24]. Pectin can be extracted from the skins of fruits such as lemons and apples, and carrageenan is extracted from red seaweed[24].

Of the bio-derived polymers, alginate is one of the least researched for replacing petrochemical plastics.

1.3 Alginate as a petrochemical replacement

Alginate is a non-toxic and readily biodegradable substance, making it an ideal candidate for petrochemical plastic substitution[26]. Alginates are polysaccharides which are wholly or partly deprotonated on their carboxylic groups; alginic acid is the protonated form. Some of the salts, such as Na^+-alginate, are water-soluble. Alginic acid and other salts, such as Ca^{2+}-alginate, are water-insoluble. They are primarily derived from brown algae genera such as *Laminaria hyperborean* and *Ascophyllum nodosum*, both of which are native to the coasts of the United Kingdom [27–29]. It can constitute up to 40 % of the dry matter in these plants and exists most commonly as a mix of Ca^{2+}, Mg^{2+}, K^+, and Na^+ salts[30, 31]. Alginate can also be produced by two genera of bacteria: *Pseudomonas* and *Azotobacter*, although these are used much less commonly compared with algal sources due to their lesser abundance and extraction efficiency, and the common occurrence of O-acetylation (a $CCOH_3$ group on a hydroxyl group) on the molecule which does not occur in algal-sourced alginate[32–34].

1.3.1 Alginate structure and function in algae

Alginate is a linear co-polymer that consists of two constituent monomer types which are both uronic acid residues: α-L-guluronate (G) and β-D-mannuronate (M). The residues are epimers of each other with the stereogenic centre on the ring carbon adjacent to the carboxylate group as shown in Figure 1.1.

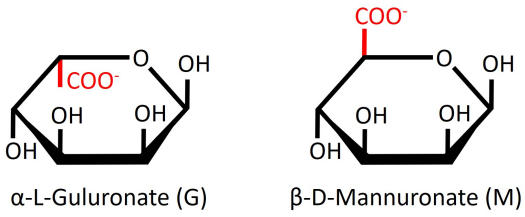


Figure 1.1: α-L-guluronate (G) and β-D-mannuronate (M) which combine to create alginate chains. The carboxylate groups, which present with different configurations between the monomers, are highlighted in red.

The "L" and "D" labelling refers to the direction of a hydroxyl group in the

monomers' linear Fischer projections. "L" refers to the latin *laevus* meaning left, and "D" denotes *dexter* meaning right. The direction is based on the same property in glyceraldehyde, a simple molecule used for comparison in the labelling process. The group of interest is the C-O-C oxygen atom (shown in Figure 1.1): for M molecules, the projection matches that of D-glyceraldehyde and for G it matches L-glyceraldehyde.

The α and β labelling of these molecules refers to how they bond. Alginate monomers bond through 1-4 glycosidic links, a covalent bonding type which links the C1 atom of the 1'-end monomer to the C4 of the 4'-end monomer. When the OH group on the C1 carbon is below the monomer ring it is an α-glycosidic bond, and when it is above the monomer ring, it is a β-glycosidic bond. Examples of these different bonding types, and the atom and monomer labelling used, are shown in Figure 1.2.

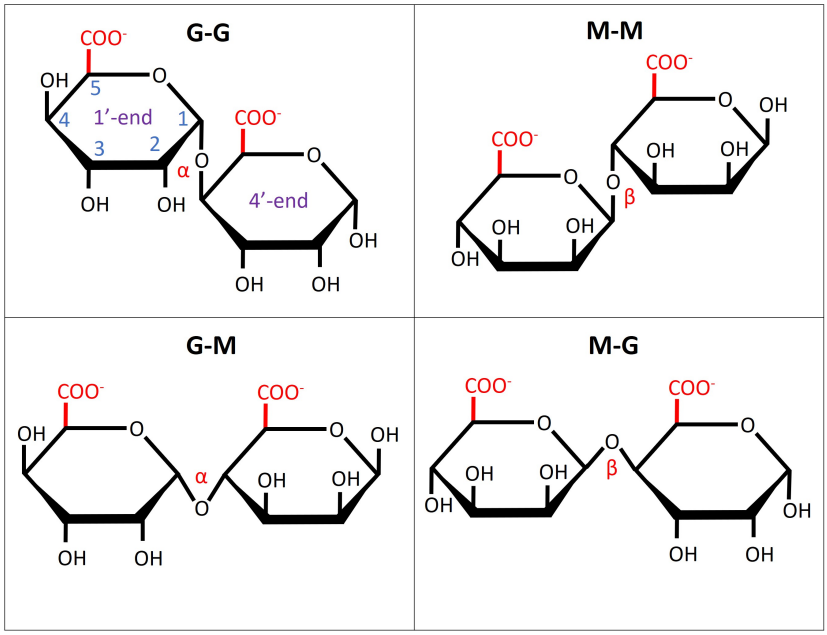


Figure 1.2: Examples of α- and β-glycosidic links with atom and monomer labelling.

While the exact biosynthesis pathway which leads to alginate creation in algae is not fully understood, it is believed that alginate is first produced as a poly-M chain, with no G residues[35, 36]. These chains are then epimerised to include G monomers[35, 36]. The epimerases involved in this process create different M/G patterns and ratios in the alginate chains depending on which specific enzyme is active; some of these enzymes

produce only G blocks, others produce alternating G-M blocks[35, 36].

It is also not fully understood why algal plants produce different forms of alginate in this way but it has been found to be dependent on many factors. Different G/M ratios and patterns were found in algae of different species and ages, in different parts of the plant, and in algae that was harvested in different locations and seasons and where the water had different levels of turbulence and salinity[35, 37].

Due to their growth occurring solely in water, the molecules that give brown algae their structure need to be more flexible than those on land. Cellulose, which does occur in algae in relatively low quantities but is the primary structure-providing molecule in terrestrial plants, cannot provide the level of flexibility needed, and so alginate takes its place. One of alginate's greatest benefits is its customisability that allows the algae plants to fine-tune their flexibility and stiffness. G blocks bond to form buckled structures whereas M blocks form flat, ribbon-like structures, and G-M bonding is a mixture of these two. This is shown in Figure 1.3, which shows a G-G-G-M-M-M chain. It is thought that the buckled structures in G blocks are less flexible than the ribbon-like structures of M blocks and offer more strength than their M counterparts, but in contrast, the M blocks provide a much needed flexibility that can help the plant cope with, for example, high water turbulence[1, 30].

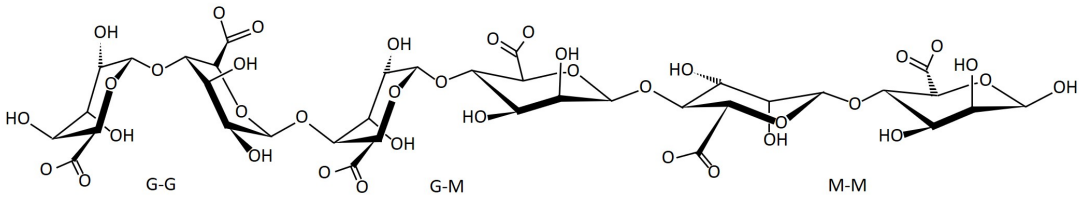


Figure 1.3: A G-G-G-M-M-M alginate chain showing the relatively buckled and straight-chain structures in G and M blocks, respectively, and the combination structure of a G-M linkage [adapted from [1]].

1.3.2 Alginate film forming behaviour

Most of the attention and research around alginate exists because of its gelling and film-forming abilities which are what make it an interesting candidate for petrochem-

ical plastic replacement. Alginate can form films via two distinct mechanisms: acid precipitation and crosslinking with cations. Both gel types are thermo-irreversible, insoluble in water, and can form over a wide temperature range[38–40].

Acid precipitation occurs through protonation of the carboxylate groups along the alginate chains[40]. This is achieved by reducing the pH of the alginate solution to below the disassociation constant, pK_a, which has been found to be 3.65 and 3.38 for G and M residues, respectively[40, 41]. This process forms alginic acid, collapses the electrostatic repulsion between the chains and instead creates a hydrogen-bond crosslinked network[40].

The second method is through multivalent cation crosslinking. The negatively charged groups along the alginate chain, primarily the carboxylate groups, on multiple different chains interact with a multivalent cation and form a network through these electrostatic interactions. It has been noted that the affinity between alginate chains and divalent cations increases in order of Hg^{2+}, $Mg^{2+} < Mn^{2+} < Zn^{2+}$, Ni^{2+}, $Co^{2+} < Fe^{2+} < Ca^{2+} < Sr^{2+} < Ba^{2+} < Cd^{2+} < Cu^{2+} < Pb^{2+}$[40, 42–44]. The most commonly used ion for this purpose is Ca^{2+} due to it's non-toxicity and availability[40].

Both gelling methods present with gel strength proportional to the G content of the alginate but this effect is less pronounced in acid precipitation than in cation crosslinking[38–40]. This is thought to be due to the bonding structures which G and M residues form, discussed above, and shown in Figure 1.3[1, 30, 38–40]. Along with forming stronger but less flexible structures than M blocks, G blocks also have the ability, through their buckled bonding structure, to form junction zones which enable more secure chain-cation-chain interactions to form but also aid in producing strong interchain hydrogen bonds in the acid precipitation gelling method[38–40].

It is widely believed that these junction zones allow the formation of the "egg-box" model, proposed in 1973[45], of alginate-cation interaction as demonstrated in Figure 1.4. The "egg-box" model consists of the buckled poly-G chains forming dimers in which each junction zone has a Ca^{2+} ion which mediates the chain-chain electrostatic interactions. By using an Atomic Force Microscope to bring oligomers of G blocks together in the presence of Ca^{2+} ions and then pull them apart after different amounts

of time, Bowman *et al.* (2016) showed that a minimum length of eight monomers is required in two poly-G chains to create a stable G-Ca^{2+}-G interaction, which was presumed to represent the "egg-box" model[46].

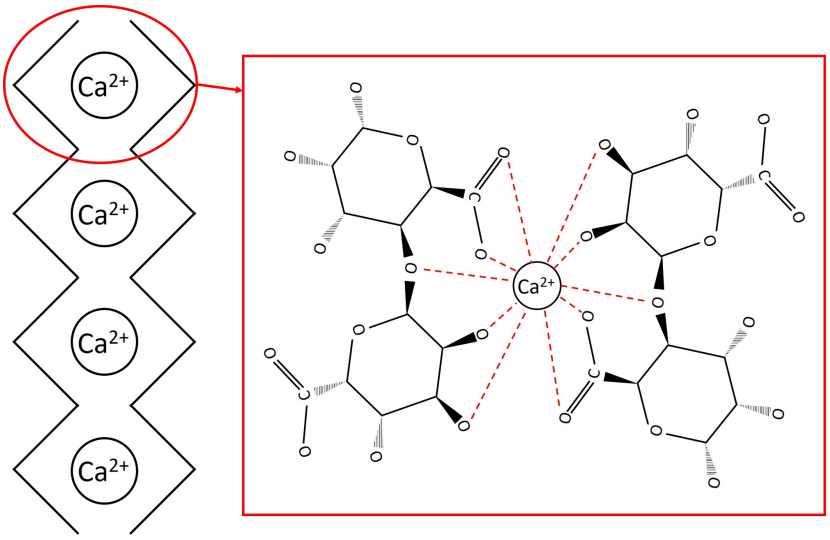


Figure 1.4: The "egg-box" alginate-Ca^{2+} ion interaction model which shows the buckled bonding structure of G block alginate forming junction zones which accommodation Ca^{2+} ions well.

Alginate is most commonly distributed as Na^{+}-alginate. There are several methods of converting a solution of Na^{+} alginate to a Ca^{2+} alginate gel. Perhaps the simplest is by putting a Na^{+}-alginate solution into contact with a Ca^{2+} source and allowing the Ca^{2+} ions to diffuse through the alginate network[38]. On contact with Ca^{2+}, alginate will form a gel instantaneously which can make forming uniform, defect-free gels difficult. A Ca^{2+} chelator such as ethylenediaminetetraacetic acid (EDTA) can prevent the Ca^{2+} interacting with the alginate until it has been fully dispersed at which point the solution is acidified which releases the Ca^{2+} ions and allows gelling to occur[38, 47]. Another way to introduce Ca^{2+} to alginate without having instantaneous gelling is to heat the solutions; this keeps the molecules in thermal motion until the Ca^{2+} has fully dispersed, at which point the solution can be cooled [38].

Water is retained within the network through hydrogen bonding between alginate and water molecules. During the process of alginate-Ca^{2+} crosslinking, the network contracts and water is pushed out of the gel. This is called syneresis and the extent

to which it occurs is dependent on the M:G ratio and bonding pattern, the molecular weight of the alginate, and the Ca^{2+} concentration[38].

1.3.3 Current uses of alginate

Alginate has been in commercial use for approximately a hundred years and is currently used in a variety of different industries including food and medical[48].

Some of alginate's most common uses are as a thickener, stabiliser, or gelling agent in food products such as jellies, mousses, and dairy products[38]. Alginate has also been widely studied as an edible film and food coating in which it extends shelf-life of fruit[27, 49–55], raw fish[27, 56–62], raw meat[27, 63–66], preserved meat[27, 67–69], and cheese[27, 70]. The alginate itself is used as a hygroscopic barrier between the food and external environment to prevent dehydration and slow oxygen access to the food. Alginate cannot be used non-specifically as shown in some studies that found that an alginate coating on it's own aided microbe growth and therefore food degradation[62, 71]. In most of these studies, the alginate coating itself was shown to have no or minimal effects in reducing degradation in the food product[49–61, 64–66, 69, 70, 72]. Therefore, active ingredients are added which can boost this effect and provide beneficial properties.

The ability of alginate to hold and slowly release active ingredients in this way also makes it useful in the medical and material industries for encapsulation and controlled release, and as a carrier for material-improving properties. The active ingredients used in the food and medical industries are often chemicals which posses anti-bacterial, -fungal, -microbial, -oxidant, -mutagenic, or -carcinogenic properties but are volatile or unstable, hence their requirement for controlled, sustained release. Other active ingredients include sensory additives (that smell, taste, or look good), vaccines, vitamins, drugs, probiotics, cosmetic ingredients, and flame retardants[32, 73–93].

Alginate's non-toxicity and hydrogel properties are often utilised in medical applications where it can be used for 3D printing of synthetic vascular tissue, bone, and cartilage, wound dressing, dentistry moulds, and in products which prevent gastric reflux[32, 89, 94–97]. Alginate also has antiviral properties due to its ionic polysaccha-

ride nature which can affect viral integrity and disrupt virus-host interaction[32, 98]. Outside of these more prominent uses in food, materials and medicine, alginate has been used as a binder for 3D printing colourant, as a biosorbent for heavy metal and methylene blue removal, and in the fabrication of metal hollow spheres[32, 99–101].

Alginate possesses many good qualities which have already been discussed: it's sustainable source from seaweed, non-toxicity, biodegradability, storage and slow release of active ingredients, and ease of film-forming. Alginate's main obstacles to being used in even more applications, and being used more commercially, lie in its non-ideal mechanical properties and its hygroscopicity[102, 103]. While the hygroscopicity of alginate can be a strength, as in reducing moisture loss of coated fruit, it also causes swelling, weakening, and can allow the transport of water-soluble substances through the gel[104–106]. The non-ideal mechanical properties of alginates are more wide-ranging. The main issues are the brittleness and lack of toughness which are present in pure alginate gels and films which is most obvious when they are dried, although the instability over time which arises from environmental sensitivity (moisture, ion presence, pH) is also a contributing factor[107–109]. Research has been conducted to reinforce alginate products and reduce their hygroscopicity by introducing another polymer (in a co-polymer or a blend), plasticisers, and/or fillers into the system but this is far from fully described or optimised[102, 110–117].

While these studies and alginate's use in the food industry show a practical understanding of alginate and its functions, there is still a lot which is not known about it.

1.3.4 Gaps in the knowledge of alginate

From the start of an alginate product's lifetime, there are issues and unknowns. To extract alginate from an algal source there are several processing steps which are now being analysed for their sustainability[118]. From the balance of transportation cost saving when the biomass is dried compared to including a cost-intensive drying step in the process, to the long extraction times, high solvent use, and use of toxic chemicals common in conventional extraction techniques, alginate extraction has become a focus

of "greener" processing research[118–120]. Novel alginate extraction techniques have been shown to reduce extraction time, reduce solvent use, keep conditions closer to ambient for the duration of the process, produce high purity alginate, and use green solvents in place of traditional toxic ones such as formaldehyde and chloroform[118, 120–128]. There is no standardised extraction process, however, and a scalable, green, cost-effective process is yet to compete with the conventional methods[118].

From extraction to manufacturing, the customisability of alginate, while a very interesting characteristic, provides challenges in product creation. Currently there does not seem to be a way to analyse alginate batch-to-batch to describe its composition further than the percentage G content and molecular weight[129]. The way alginate is processed and extracted from algae can affect it's composition, as can the algae itself, and its environment. It is therefore essential to have a description of what the final alginate product is so that it can be treated in the relevant way[129–131].

Once the product is created, there are knowledge gaps in current uses of alginate which need further investigation before it can be implemented as widely as alternative materials. These unknowns include biological processes such as controlled biodegradation in mammals who do not have alginate lyases, and information on how alginate interacts and reshapes the human microbiome[132–135]. There are also questions about ion exchange, swelling, and active ingredient binding in alginate drug carrying [136], and a lack of mapping of alginate properties versus desirable product characteristics in this industry [137, 138].

Some examples of this missing knowledge come from a lack of understanding of the very basic structural and molecular properties of alginate, including the behaviour of both G and M chains in the system, gelling mechanisms, ion interactions, and structural models, such as the "egg-box" model. While it is clear from practical use that G chains offer a more viscous liquid and a stronger gel, the molecular basis for this in Na^+ and Ca^{2+} alginate systems is unclear. The differences in behaviour of alginate systems with Na^+, Ca^{2+}, or both ions are essential in applications such as drug release and product stabilisation [136], and yet these systems are not well understood at a molecular level. In particular, the "egg-box" model, which is widely discussed and referenced, is more

of a useful picture rather than a definitive and carefully studied binding method, and it is still debated in the literature[139–146].

These fundamental properties can be difficult or impossible to study in an experimental setting due to the high resolution and level of detail required. Computational methods, which are becoming increasingly useful and accessible, allow a molecular, or even atomistic, view of a system and allow the analysis of these fine details such as atom movements, bonds, and interactions. Some computational simulations of alginate have been completed, but there are relatively few of them and in many cases there is a lack of consensus between the studies[141, 146–152].

1.3.5 Thesis aims and objectives

This project was funded by the Biotechnology and Biological Sciences Research Council (BBSRC) and Industrial Biotechnology Innovation Centre (IBioIC). The IBioIC pairs academic research facilities with companies to help foster industrially relevant partnerships and research. Through this collaboration, this work was partnered with OCEANIUM®, an innovative seaweed ingredient developer. The simulation work was proposed to help OCEANIUM® understand alginate better, a product they produce in large quantities and have less use for than other products. The experimental work was carried out with, and guided by, OCEANIUM® with the aim of helping them develop new and better products.

In this thesis I aim to investigate the impact of monomer type and the presence and concentration of Na^+ and Ca^{2+} ions on the behaviour of alginate. These aims were realised through the following objectives.

1. The collection of alginate, water, and ion parameters coupled with the calculation and optimisation of partial atomic charges on alginate molecules to produce a validated general alginate model for use in computational simulations.

2. Molecular dynamics simulations to investigate the differences between systems with G, M, or both residue types, and with either Na^+ or Ca^{2+} ions.

3. Extending this computational work into an experimental study which investigated

similar systems in addition to conditions used to cure Na^{+}-alginate to Ca^{2+}-alginate in a systematic and comparative study.

4. Using the previous simulation and experimental work to guide further computational investigations of systems with differing G, M, and ion compositions.

This thesis follows the same structure as described in the objectives above, with a general methodology section following this introduction. The methodology outlines the underlying processes and calculations used in both the computational and experimental studies with the specific methodologies included in the relevant chapters. Chapter 3 describes the work done in calculating and optimising the alginate charges for use in simulation work and Chapter 4 presents the first set of simulation work which investigates different alginate systems. To guide further simulation work, the next study, presented in Chapter 5, was experimental and details the effects of different curing conditions on Ca^{2+}-alginate gel strength. The second simulation study focuses on the most interesting results from both these previous chapters, and is presented in Chapter 7. Between the first and second simulation studies, the software used for these studies was changed. The details of, and the reasons behind this change are presented in detail in Chapter 6.

Chapter 2

Methodology

2.1 Overview of methodology

This thesis uses both experimental and computational methods, with Chapters 3, 4, 6, and 7 presenting details of molecular dynamics (MD) simulations, and Chapter 5 presenting a design of experiments (DOE) study. In this chapter, the fundamental processes and calculations involved in MD and DOE are presented. More specific methodologies are given in Chapters 3-7.

2.2 Molecular dynamics simulations

This section will outline the general MD methodology applied within Chapters 4, 6, and 7, and discussed in Chapter 3. We start with the theoretical foundation of MD, how it works, and what we need to create a system, before briefly discussing the analysis techniques and software that were chosen for this project.

2.2.1 Foundation of molecular dynamics

MD is a computational method which allows the simulation of atoms, molecules, and systems. It is a useful tool to look at the fine details of a system, allowing the visualisation and analysis of each atom and bond as required. To begin a simulation, several inputs are required: (i) initial positions and velocities of all atoms in the system, (ii) atom properties including types, masses, radii, and interaction affinities, (iii) properties

of bonds between 2-4 atoms including equilibrium lengths and angles, and resistance to change, and (iv) details of what calculations and processes should occur during the simulation. The details of points (ii) and (iii) in this list are contained in the *force field* which is used to model the species within the system. Force fields will be discussed in more detail in Section 2.2.2 following an overview of how systems develop in MD.

Initial positions of all atoms in the system should be reasonably realistic, with no atom overlaps or molecule entanglements. This prevents overly large initial forces being created which can cause instabilities in the system. This is relatively simple as most MD software packages have commands which allow the addition of molecules randomly in a system without overlap with existing atoms[153, 154].

Initial velocities are often assigned according to the Maxwell-Boltzmann distribution, shown in Equation 2.1. T is the target temperature of the system, P_i is the probability distribution of the velocities and k_B is the Boltzmann constant. Each component of the velocity sampled is assumed to be statistically independent, so each is sampled from a one-dimensional Gaussian distribution which is related to the target temperature.

$$\begin{aligned} P_i &= \left(\frac{m\beta}{2\pi}\right)^{\frac{3}{2}} \exp\left(-\frac{m\beta}{2} v_i^2\right) \\ \beta &= \frac{1}{k_B T} \\ k_B &= 1.38 \times 10^{-23}\ m^2\ kg\ s^{-2}\ K^{-1} \end{aligned} \tag{2.1}$$

Newton's Equations of Motion (EOM, shown in Equation 2.2) are used to determine, for each atom i, the (i) force ($\mathbf{F}$), (ii) resulting velocity ($\mathbf{v}$), and (iii) destination ($\mathbf{r}$) after a short time step. $\mathbf{a}$ is the acceleration, m is the atom mass, and $\mathbf{F}$, $\mathbf{r}$, $\mathbf{v}$, and $\mathbf{a}$ are vectors each with x, y, and z components.

$$\begin{aligned} \mathbf{F}_i &= m_i \frac{d^2\mathbf{r}_i}{dt^2} \\ \mathbf{v}_i &= \frac{d\mathbf{r}_i}{dt} \\ \mathbf{a}_i &= \frac{d^2\mathbf{r}_i}{dt^2} \end{aligned} \tag{2.2}$$

There are several different methods used to solve the EOM in MD. Perhaps the simplest is the standard Verlet integrator which uses the current and previous positions and current forces to compute new positions with Equation 2.3[155]. Velocities are not explicitly used in the integration step and must be estimated afterwards. The standard Verlet integrator is therefore not ideal for use with thermostats and barostats which require velocities; this is discussed in more detail in Section 2.2.3. It is, however, simple and stable when used with a small time step (Δt) and is good for energy conservation over long timescales.

$$\mathbf{r}(t + \Delta t) = 2\mathbf{r}(t) - \mathbf{r}(t - \Delta t) + \frac{\mathbf{F}(t)}{m}\Delta t^2 \tag{2.3}$$

The Verlet algorithm evolved into the Leapfrog Verlet algorithm, which calculates the velocities at half time steps ($t + \frac{1}{2}\Delta t$) and the positions at full time steps ($t + \Delta t$)[156, 157]. This reduces numerical error propagation and works well at preventing energy drift over long simulations[156, 157].

The most common integration algorithm used in MD simulations is the Velocity Verlet algorithm which updates the positions and velocities together at each time step, as shown in Equation 2.4[156, 158]. The benefits found in the standard Verlet algorithm are preserved in the Velocity Verlet algorithm however the Velocity Verlet algorithm is more compatible with thermo- and barostats due to its velocity tracking.

$$\begin{aligned} \mathbf{r}(t + \Delta t) &= \mathbf{r}(t) + \mathbf{v}(t)\Delta t + \frac{1}{2}\mathbf{a}(t)\Delta t^2 \\ \mathbf{v}(t + \Delta t) &= \mathbf{v}(t) + \frac{1}{2}\left[\mathbf{a}(t) + \mathbf{a}(t + \Delta t)\right]\Delta t \end{aligned} \tag{2.4}$$

Other MD integration algorithms include stochastic integration or multiple time step integration. Stochastic algorithms use random noise and friction terms, which disrupt system dynamics, so that temperature is controlled directly in the algorithm[156]. Multiple time step integrators are often used in large biomolecular simulations to reduce computational cost. In these methods, fast and slow forces are integrated separately: bonded interactions are calculated on short time step intervals and non-bonded forces use a longer time step[156].

For all of the above integration algorithms, the choice of time step is vital. If a time step is too big, the particles can move too far in a single step, produce artificial effects such as particle overlap and unrealistically large forces, and the system can become unstable. If the time step is too small, computational power is used in calculating atom movements, velocities, and forces which are not different enough from the previous time step to be interesting. Beyond a certain point, reducing the time step does not, therefore, improve accuracy, and instead it can cause rounding errors which may accumulate over long time scales.

It is common in all-atom MD to use a time step of 1-2 fs. In systems with less detail, a larger time step may be useful. A 1 fs time step might be used when no atoms are constrained so that the fastest motions (usually hydrogen atoms) are captured. A 2 fs time step is more common when fast moving bonds are constrained because simulation speed is increased while maintaining bond vibration details.

It is also important to decide how many time steps will be integrated over within the simulation. Simulation timescales can range from femtoseconds to microseconds depending on the resources available and the interesting timescales applicable to the system. In systems with relatively small polymer chains in water, a timescale range of 10-200 ns is common[150, 151, 159]. Ideally a total simulation time is chosen which is long enough to produce ergodicity, in which the system explores all accessible states and time-averaged properties reflect true physical properties of the system. Scales into microseconds would be ideal for polymers as this means the system would be reaching full relaxation but available computational power can limit this[160, 161].

Energy minimisation to remove unfavourable conformations, and equilibration MD are common precursors to production MD runs in which data is extracted for analysis. Equilibration is the point at which all memory of the initial configuration has been lost[156]. There is no definitive equation to follow to ensure that equilibration has been reached but using thermodynamic quantities and allowing them to stabilise around the target value is common practice[156]. Ideally, all system qualities are analysed to ensure equilibration is reached, where stabilisation is usually the equilibration marker.

The outputs from simulations are commonly the trajectory which details the x, y,

and z coordinates of each atom at different time steps in the simulation, the thermodynamic properties such as temperature, pressure, volume, and energy, and any other user-defined outputs specific to the system and methodology being applied.

As mentioned, the force experienced by each atom is key in progressing MD methodology. A collection of parameters and equations, unique to each system, is employed to calculate the forces acting on each atom. This collection is called the *force field*.

2.2.2 Force Fields

The force on each atom in a simulated system is found by calculating the negative gradient of the potential energy surface for that atom at that time step[156]. This is shown in Equation 2.5 where V is the potential energy at position $\mathbf{r}$.

$$\mathbf{F}_i = -\nabla_{\mathbf{r}_i} V(\mathbf{r}) \tag{2.5}$$

There are three classes of force field, each with its own functional form to solve Equation 2.5[156]. Class II and III force fields offer better accuracy where, for example, there is deviation from ideal harmonic approximation or the inclusion of reaction chemistry. Both these classes, however, come at a increased computational cost. This work uses only Class I force fields, which balance speed and accuracy, and take the form of Equation 2.6 in which the energy potential is composed of the bonded and non-bonded energies[156].

$$\begin{aligned} V_{total} &= V_{bonded} + V_{non-bonded} \\ V_{bonded} &= V_{bond} + V_{angle} + V_{dihedral} + V_{improper} \\ V_{non-bonded} &= V_{Lennard-Jones} + V_{Coulombic} \end{aligned} \tag{2.6}$$

Atom types

Atom types are used throughout force fields to distinguish what type of atom, not just which element, is being described. Many different "types" of one element can be described by a force field, where the differences usually arise due to their bonding

environment. Every atom in a simulated system is designated an atom type which complies with the force field's description of that atom. Each parameter is then specified based on these atom types.

Bonded potentials

Bonded potentials arise when atoms are bonded to each other. There are four ways in which they are described: (1) stretching of a bond between two atoms, (2) bending of an angle between three atoms, (3) torsion in a proper dihedral (commonly referred to as just a dihedral) between four atoms bonded linearly, and (4) torsion in an improper dihedral (improper) between four atoms in which three atoms are bonded to a central atom. Visuals of these bond descriptions and examples of their different states are given in Figure 2.1.

Class I force fields, as used in this work, implement Equations 2.7-2.9 for the bonded potentials where Equation 2.9 is used for both proper and improper dihedrals[156]. k, with its various subscripts, is the force constant and represents the resistance to change within the bond length, angle, or rotation. A large k value means that the bond is "stiff" and resists deformation. l^0 and θ^0 represent the equilibrium values of the bond length and angle. The force field parameters used for dihedrals and impropers are the periodicity (n) and phase shift (γ).

$$V_{bond} = \frac{1}{2}k_b(l - l^0)^2 \tag{2.7}$$

$$V_{angle} = \frac{1}{2}k_a(\theta - \theta^0)^2 \tag{2.8}$$

$$V_{dihedral} = \frac{1}{2}k_{d,n}\left[1 + \cos(n\omega - \gamma)\right] \tag{2.9}$$

Non-bonded potentials

Non-bonded potentials describe the repulsive and attractive forces which exist between atoms that are not bonded to each other. The Lennard-Jones (LJ) potential (Equation 2.10) describes the short-term (Pauli) repulsion and longer-range attractive forces

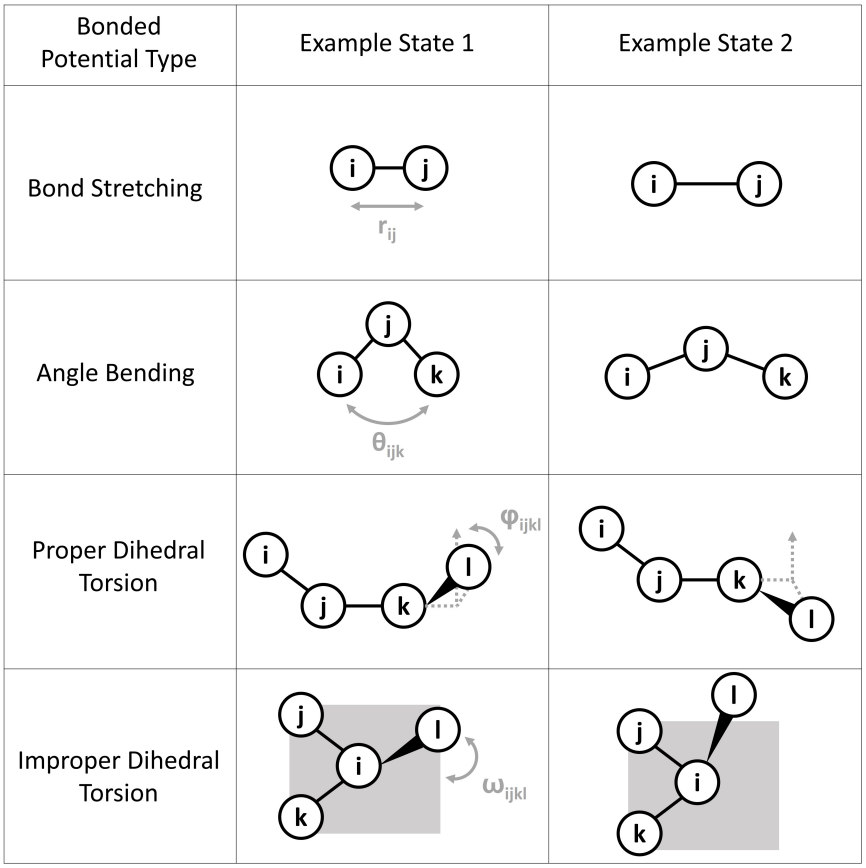


Figure 2.1: Example states of each of the different bonding potential types, with their defining parameter labelled.

(dispersion) which arise from the atom size and polarisability, where r_{ij} is the distance between the centres of atoms i and j[156]. σ and ϵ are shown in Figure 2.2 where σ is a representation of an atom's size and ϵ, the potential well depth, is a measure of the attraction between two atoms.

$$V_{LJ} = 4\epsilon_{ij}\left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6}\right] \tag{2.10}$$

The parameters σ and ϵ are described in force fields as the distance and attraction between two of the same atom. When the interaction is with a different type of atom, the parameters for the two interacting atom types need to be combined to describe

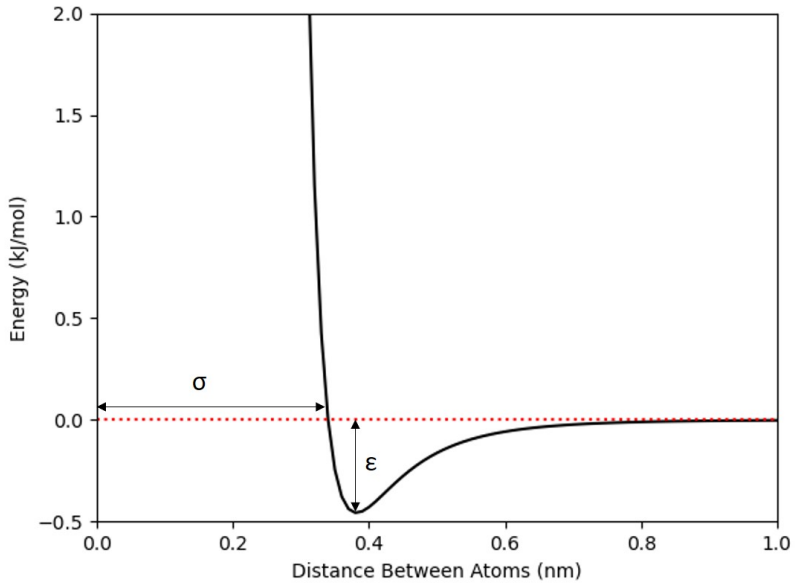


Figure 2.2: A depiction of the Lennard-Jones potential showing σ and ϵ. There is an area of repulsion when the two atoms in question are too close to each other and experience overlap, followed by attraction until the atoms are too far apart to "see" or influence each other.

this relationship. While there are a few methods of combining σ and ϵ for unlike atom types it is dependent on what the force field being used was parameterised with. The Lorentz-Berthelot combining rules, shown in Equation 2.11, are the most common; σ_{ij} is calculated using the arithmetic mean and ϵ_{ij} is calculated using the geometric mean which means that the size of the two interacting atoms is the average of the individual atom sizes and that the interaction strength is proportional to the compatibility of the two atom types. These rules are commonly applied due to their simplicity, widespread compatibility, and because they generally give acceptable results for biomolecules, and small molecules in solution[162].

$$\begin{aligned} \sigma_{ij} &= \frac{\sigma_{ii} + \sigma_{jj}}{2} \\ \epsilon_{ij} &= \sqrt{\epsilon_{ii}\epsilon_{jj}} \end{aligned} \tag{2.11}$$

The Coulombic potential, shown in Equation 2.12, describes the attractive and re-

pulsive forces felt between two atoms due to their charge; q is the atom charge, and ϵ_0, separate from ϵ in Equation 2.10, represents the vacuum permittivity or electric constant which is a measure of the ability of space to permit electric field interactions[156].

$$V_{Coulombic} = \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}} \tag{2.12}$$

The charges in Equation 2.12 refer to the atomic partial charges on each atom. Atomic partial charges arise due to asymmetry of electrons involved in atom bonding. Typically these are treated as fixed in MD simulations which allows the simulation of much bigger systems compared with density-functional theory (DFT) or other quantum methods which include the simulation of electron movements. For well studied molecules such as protein, nucleic acids, and amino acids, the partial atomic charges are commonly included in the force field documentation. For less studied molecules, such as alginate, these charges need to be calculated by the user. Chapter 3 details the calculation of partial atomic charges for alginate.

Cut-offs and constraints

To increase speed and reduce unnecessary calculations while maintaining accuracy within the simulation, it is common to apply cut-offs to the non-bonded potentials. LJ interactions are usually not calculated past a cut-off of 9-12 Å because, at this distance, both the attractive and repulsive terms become very small and are often smaller than $k_B T$ which represents the thermal noise[163].

It is also common to split short- and long-range electrostatic potentials to maximise efficiency. Short-range values, usually also up to a cut-off of 9-12 Å, are calculated using Equation 2.12[163–165]. Long-range values, at distances above the cut-off, are commonly calculated in Fourier space where waves account for repetitions and reduce the calculation time[164, 165]. Short-range potentials are not calculated in Fourier space due to the higher variation which means Fourier methods are slower and less accurate[164, 165]. Fourier methods for calculating electrostatic potentials include the Particle Mesh Ewald (PME) and Particle-Paricle Particle-Mesh (PPPM, P^3M)

methods[165]. These methods are essentially the same although PME does offer some efficiency advantages over PPPM.

For similar reasons of reducing computational expenditure while maintaining accuracy, it is also common to constrain bonds which do not add necessary detail to the system. When a time step is chosen which does not allow the very fastest motions in the system to be described well, it is common to "freeze" or constrain these motions. Common motions to constrain are hydrogen atom bonds such as C-H or O-H, including in water molecules. Removing the bond and angle vibrations in water molecules can massively increase simulation speeds especially if there are numerous water molecules in the system.

There are a few common algorithms which are used to add constraints to a system. SHAKE and LINCS correct each constrained bond in each time step. LINCS is faster and more stable than SHAKE for large systems because it is non-iterative and typically requires one calculation per time step; SHAKE is iterative[166, 167]. LINCS is, however, less accurate for very stiff bonds. While SHAKE and LINCS can both be used to simulate rigid water molecules, it is often preferred to use the SETTLE algorithm which is an analytical solution specifically for rigid water molecules[168]. SETTLE calculates the water's oxygen atom position and velocity at each time step as it would any other particle in the system following the chosen integration algorithm. It then computes the rotation of the hydrogen atoms around the oxygen based on angular momentum conservation before correcting the positions of the hydrogen atoms relative to the oxygen atom to conserve the rigid bond and angle parameters[168].

Force fields used in this project

Throughout this work, the GLYCAM-06j force field[169] was used to model alginate. GLYCAM is part of the Assisted Model Building with Energy Refinement (AMBER)[162] repository and is specific to carbohydrates. This force field was chosen due to its previous use with alginate[170–172] and its inclusion of pre-defined uronic acids[169]; alginate is not included in this but it does show that the force field has been validated extensively for alginate-like molecules. Other force fields have been used in

literature for alginate systems such as CHARMM[173–175] and OPLS[152, 176, 177], although these force fields are more applicable to systems containing proteins, nucleic acids, and lipds, and organic liquids, respectively.

The parameters used to model Na^+, Ca^{2+}, and Cl^- ions in this work are recommended by AMBER because they were used in the parametrisation of GLYCAM-06j[162]. These parameters were presented in work by Li *et al.* which used free-energy perturbation in different explicit water models and evaluated ion-oxygen distances to produce ion parameters which replicated experimental values[178, 179].

It is possible to model water implicitly as a featureless continuous medium which reproduces net thermodynamic and screening effects. This method has been used in alginate work before, primarily to simplify the system[148, 149]. This is a debated method as water molecules are essential in alginate-alginate and alginate-ion interaction mediation. Most previous work, therefore, has employed an explicit water model in which each water molecule is modelled individually[150–152, 180–182]. An explicit water model was therefore used in this work.

AMBER used the well-established, explicit, Transferable Intermolecular Potential with 3 Points (TIP3P) water model[183] in parametrising the GLYCAM-06j force field. TIP3P was first proposed to allow more efficient simulation of water. The model assumes that each atom on the molecule has a fixed point charge, that the molecule is rigid, with fixed bonds and angle, and that the interactions involved are solely non-bonded[183].

The parameters used in the GLYCAM-06j, ion, and water models are listed in Appendix A.

2.2.3 Ensemble selection

In MD, three thermodynamic variables are controlled to fully describe the macroscopic state of a system in statistical mechanics. The chosen three variables correspond to the natural parameters of thermodynamic potentials such as internal energy, Helmholtz free energy, or Gibbs free energy[156]. Each set of three thermodynamic variables, when held constant in a simulation, is described as an ensemble. These include the

microcanonical ensemble which controls the number of particles (N), the volume (V), and the energy (E); the canonical ensemble (N, V, and temperature, T); the isothermal-isobaric ensemble (N, T, and pressure, P); and the grand canonical ensemble (V, T, and chemical potential, μ)[156].

Each of these ensembles can be used to represent a different real-world experiment or process. For example, the canonical ensemble approximates an experiment undertaken in a sealed container submerged in a heat bath, and many industrial, lab-scale, and biological process use the same conditions as the isothermal-isobaric ensemble. To apply these two ensembles, the N, V, T and N, P, T need to be able to be controlled, respectively.

If the box which contains the simulated system had impenetrable sides, and the atoms were enclosed, it would be hard to approximate or compare these small systems to experimental or industrial work which occurs on much larger scales, as the interface-volume ratio would be considerable higher than in an experimental system. To overcome this challenge, we employ periodic boundary conditions (PBC)[156] which are used to approximate bulk material behaviour by embedding the finite simulation box within an infinite lattice of identical box replicas. Each particle, therefore, interacts with the particles inside the primary simulation box and with the nearest periodic images of particles in neighbouring cells[156]. When viewing the simulated system on it's own this phenomenon is seen as an atom leaving the system with a specified direction and velocity, and re-entering the same system at the opposite side of the box with the same direction and velocity. Volume is still controllable with PBC employed by altering the size of the simulation box; molecules have a fixed space to move about in causing pressure changes within the box.

Number of particles in the system is controlled simply by not adding or removing any particles from the system during the simulation. Pressure in the simulation box is calculated using the virial theorem, which describes how the kinetic energy and interatomic forces contribute to stress within the simulation volume. Pressure and temperature are controlled by using thermostats and barostats, respectively. There are many different methodologies used in thermo- and barostats, each with advantages and

disadvantages.

Thermostats

The Berendsen thermostat employs velocity rescaling[184]. This allows control of the kinetic energy which directly links velocity to temperature. This is shown in Equation 2.13 where $\langle K \rangle$ is the total system kinetic energy and N is the total number of particles in the system. The Berendsen thermostat suppresses fluctuations and therefore does not sample the canonical ensemble correctly. The V-rescale thermostat applies the Berendsen thermostat but with a stochastic term which adds random noise but samples the canonical ensemble correctly; a velocity rescale is performed on the system, usually every specified number of time steps and therefore allows the temperature to fluctuate but produce an average of the target temperature value[185].

$$T = \frac{2\langle K \rangle}{3Nk_B} \tag{2.13}$$

Extended system dynamics methodology, used by the Nose-Hoover and Andersen thermostats, couples the system with a fictitious thermal reservoir. In the Nose-Hoover thermostat, this reservoir is coupled to the system's kinetic energy and alters it using a friction-like term (ξ) as seen in Equation 2.14. This term changes with the system's kinetic energy; if the systems kinetic energy is too high, for example, ξ will increase to slow the particles down, and thus cool the system. The Andersen thermostat works in a slightly different way, occasionally randomizing the velocities of a subset of the system's particles to mimic collisions with a heat bath. This disrupts the real dynamics within the system and is therefore not suitable for all systems.

$$\frac{d\mathbf{v}_i}{dt} = \frac{\mathbf{F}_i}{m_i} - \xi \mathbf{v}_i \tag{2.14}$$

Other thermostat methodologies include dissipative particle dynamics, configurational, and hybrid methods. Configurational thermostats use positions and restraints as well as velocities coupled to the temperature; this is most useful in a system where

kinetic temperature is misleading, for example in non-equilibrium, or very constrained systems.

Barostats

Most barostats work by allowing the volume (isotropic) and sometimes shape (anisotropic) of the simulation box to fluctuate to maintain a target pressure. The system's atom positions can then be scaled with the box size to maintain consistency. This is how both the Parrinello-Rahman and Nose-Hoover barostats operate. Both of these barostats handle realistic volume and pressure fluctuations well but only if they are correctly tuned using the input parameters relevant to them.

Other barostats may use methods similar to the extended system dynamics thermostat methodology detailed above, with the fictitious variable coupled with box size rather than kinetic energy, or Monte Carlo methods in which a random new box size is proposed and accepted or rejected based on an energy/pressure balance probability. These methods are particularly suitable for non-cubic simulation boxes, coarse-grained models, dense systems, or when Monte Carlo is used throughout the simulation.

2.2.4 Equilibration and production

As discussed above, it is important to ensure equilibration of the system has taken place before simulating a production run. Equilibration is commonly split into several stages including energy minimisation, NVT, and NPT runs, although this is dependent on the system being studied. The goal of equilibration runs is to remove any unphysical aspects of the system, such as atoms being too close or overlapping, to allow the system to relax into a stable thermodynamic state, and for the system to "forget" its initial configuration[156]. It is a good idea to, at minimum, make sure the three thermodynamic properties which are held constant in the chosen ensemble have stabilised around the target value before the simulation is deemed equilibrated.

Energy minimisation of a system is commonly the first step as it can remove severe steric clashes without crashing the simulation. Steepest Descent is an algorithm which moves the system "downhill" along an energy gradient, putting the system into the

energetically closest local minimum. It is not, therefore, suitable for finding the system's global energy minimum in a single simulation. The Conjugate Gradient algorithm works in a similar, but more efficient, way which converges on the minimum a lot quicker due to it avoiding the "zig-zagging" behaviour used by the Steepest Descent algorithm in narrow energy valleys. The Conjugate Gradient algorithm is more efficient, especially in big systems, but it does not handle bad geometries or large overlaps as well as the Steepest Descent algorithm.

Following energy minimisation, in systems which will be run for production in the NPT ensemble, it is common to apply a short NVT MD run. Large particles, such as polymers, can be positionally restrained to allow the water and ions, and temperature of the system to stabilise. By holding the volume of the system constant, the system is prevented from "flying apart" while the temperature stabilises. The NPT simulation can then begin, with care being taken to ensure that the system has equilibrated before results are extracted.

Sampling sufficiency of the production run is mostly determined by a comparison of the system to real-world effects or literature, convergence monitoring, and replica analysis. For example, it is known that polymer chain relaxation happens in the timescale of microseconds. However, since the chains in simulations do not commonly represent the full size of experimental polymer chains, simulations run for <200 ns are common due to the use of energy-minima-finding techniques such as simulated annealing, and computational limitations [146, 148–150, 152]. Convergence can be monitored in almost all of the output gained from a simulation; if the values do not stabilize, perhaps the simulations need to be run for longer. Running multiple replicas of the simulations, either with different starting configurations and/or velocities, can help make the results gained from the simulations more robust, or show that the results from one or more simulations may be a product of, for example, their initial configurations.

Production runs are the part of the simulation where results which will be used in data processing and analysis are extracted. There are a few analysis techniques which are common to Chapters 4, 6, and 7 which are discussed here to prevent repetition between chapters. Properties of the system such as the temperature, pressure, volume,

density, and energy are either output automatically by the simulation software or are easily accessible in post-processing. Other properties can be calculated during or following the simulation with specific commands within the software being used. These properties help describe the polymer conformation, the distribution of species in the system, and the species mobility.

Polymer conformation

Radius of gyration (R_g) is a measure of how compact a molecule is, for example how much does a polymer chain curl up on itself. A relatively small R_g shows that the molecule is more compact than one with a larger R_g. The calculation involves finding the average distance between the atoms and centre-of-mass (COM) of the molecule as shown in Equation 2.15.

$$R_g = \sqrt{\frac{1}{N}\sum_{i=1}^{N} |\mathbf{r}_i - \mathbf{r}_{\mathbf{COM}}|^2} \tag{2.15}$$

Persistence length (L_p) is a measure of polymer stiffness where a greater persistence length describes a stiffer chain. The calculation to find persistence length involves comparing a bond vector on each monomer to the same vector on another monomer in the chain. Usually this is the bond between monomers as this gives a clear indication of the directionality of the polymer chain. The formulae used in calculating persistence length is given in Equation 2.16 in which s is the distance between the bond vectors, $\mathbf{t}$ is the bond vector of monomer i, and L_p is the persistence length.

$$\begin{aligned} \langle \cos\theta(s) \rangle &= \left\langle \frac{\mathbf{t}_i \cdot \mathbf{t}_{i+s}}{|\mathbf{t}_i||\mathbf{t}_{i+s}|} \right\rangle \\ \langle \cos\theta(s) \rangle &= e^{-s/L_p} \end{aligned} \tag{2.16}$$

The values of $\langle \cos\theta(s) \rangle$ are plotted against those of s, averaged over the polymer for every value of s, and this produces a graph which plots the decay of the directionality of the polymer. The faster $\langle \cos\theta(s) \rangle$ drops to zero, the more flexible the chain is.

Distribution of one species around another

Radial distribution functions (RDFs) are used to look at the particle density of a target species at different distances from a reference (origin) particle. The RDF ($g(r)$) is calculated using Equation 2.17 where ρ is the average number density of the target species, $dN(r)$ is the number of particles in the shell at distance r from the origin particle, and dV is the volume of the spherical shell at distance r from the origin particle.

$$g(r) = \frac{1}{\rho} \left\langle \frac{dN(r)}{dV} \right\rangle \tag{2.17}$$

The average number density of the target species is included in Equation 2.17 to normalise the data. Once the normalisation is complete, an RDF of 1 shows that the number density of the target species at that distance from the origin species is the same as if the target species was uniformly distributed. An RDF of greater than 1 shows some ordering of the target species towards this area. As distance from the origin particle increases, $g(r)$ tends to 1 because the system becomes uniform at large distances.

It is common to see multiple diminishing peaks as the distance increases from the origin particle, showing that there are structured "shells" of the target species around the origin species. This makes RDFs an ideal analysis technique for investigation of ion and water structure around a large molecule, such as a polymer.

When RDFs are being compared over systems with different species and concentrations it can be useful to plot the coordination numbers. These values are the number of target species that were found at the distance being investigated from the origin species. They are usually plotted as cumulative plots at the same distances as the RDFs.

Species mobility

Mean squared displacements (MSDs) are used to study particle diffusion and mobility. MSDs use different time "lag" values (τ) to calculate how far a particle moves on average

over different time intervals as shown in Equation 2.18. α is the gradient of the log-log τ versus MSD graph and gives information about the diffusivity. An $\alpha = 1$ value shows that the species is presenting with normal diffusion, $\alpha < 1$ shows sub-diffusion, and $\alpha > 1$ shows super-diffusion[186]. The MSD is related to the diffusion coefficient; the diffusion coefficient can be taken as the gradient of the linear section of the MSD (when $\alpha = 1$) plotted against τ divided by 6 (which accounts for the 3-dimensional system). When $\alpha \neq 1$, an effective diffusion coefficient can be calculated in the same way as when $\alpha = 1$ except that the diffusion coefficient will have units of $[distance]^2/[time]^\alpha$ (for example, $\text{Å}^2/ns^\alpha$).

$$\begin{aligned} MSD(\tau) &= \langle |\mathbf{r}(t+\tau) - \mathbf{r}(t)|^2 \rangle \\ MSD &\propto \tau^\alpha \\ MSD(\tau) &= 6D_\tau \end{aligned} \tag{2.18}$$

When plotting MSDs, the data at high τ values can often look noisy or less linear than the data at low τ values. This is because for large τ there are less origin t values to use in the calculation: if the simulation being analysed is 200 ns long, there is only one data point to use per particle to find out how far a single particle would move in 200 ns. This is why MSD plots can often look linear up to a point. It is common practice to discard the non-linear area of an MSD plot and focus only on the highly sampled, lower τ region.

2.2.5 Software & computational resources

There are many software and computational resources available for MD techniques, each with their own specialities. In this project, two packages of MD software, LAMMPS and GROMACS, were used. Other MD simulation software includes AMBER[187], NAMD[188], and CHARMM[173], all of which are particularly good for simulating biomolecular systems of proteins, nucleic acids, or membranes.

The Large-Scale Atomic/Molecular Massively Parallel Simulator (LAMMPS)[153] was designed for uses involving materials science and polymers. It is particularly use-

ful when using coarse-grained models and custom force fields but supports atomistic simulations too. LAMMPS is used widely in polymer research and allows for more customisation by the user than other simulating software. The GROningen MAchine for Chemical Simulations, known as GROMACS[154, 189], has highly efficient methods for simulating dilute systems and is designed for use in biological, membrane, and polymer systems. GROMACS is also highly optimised for use with GPU which can grant the user large speed increases.

The size and length of the simulations used in this project were only possible due to the use of the Academic Research Computing High-performance Integrated Environment - West of Scotland (ARCHIE-WeSt) supercomputer. ARCHIE-WeSt provides access to large-scale CPU clusters and GPU nodes which were essential for this project. Although LAMMPS was not fully integrated to work with the ARCHIE-WeSt GPU nodes when these simulations were carried out, the large-scale CPU nodes allowed significantly bigger and longer simulations to be carried out than would have been possible on a domestic computer.

2.2.6 Limitations of molecular dynamics

Computational simulation can be seen as the link between theoretical predictions and models, and between models and experimental work, but there are limitations to these methodologies[156]. Limitations lie within the accuracy of force fields, which are constantly being updated and made to produce results which more closely resemble experimental work, and in the treatment of long-range interactions, for which there are many different methodologies, each with its own advantages and disadvantages[156]. The timescale and size of the systems accessible by molecular dynamics are a common point of discussion because they currently do not compare to experimental systems[190]. However, due to the ever-increasing speed of computers and model development, systems as large as a trillion atoms and simulations as long as a microsecond can now been achieved[156, 191, 192].

The following sections discuss the underlying theory and calculations used in a design of experiments methodology. This methodology was implemented in Chapter 5.

2.3 Design of experiments

Design of experiments (DOE) methodology is used to systematically investigate the effects of different levels of multiple factors on a response variable, and identify which factors have a statistically significant influence on that response variable[193, 194]. DOE methodology encompasses the planning, execution, and statistical analysis of experimental work. The output of a DOE is (i) an assessment of the effect each factor had on the response variable, (ii) a model fitted to the data which describes the data and predicts new data, (iii) and a description of the model's ability to describe and predict observed and new data[193, 194].

In this section we will first discuss the design and experimental portions of the DOE before discussing the statistical analysis performed on the resulting data.

2.3.1 Designing a DOE

Screening or probing designs are usually employed first. These allow some information to be gained about an unknown system and suitable level ranges for each factor to be determined. For example, if there is no response change when using concentration levels of 1 and 2 mol L^{-1} but there is a significant change when using values of 1 and 10 mol L^{-1}, then the screening tests would identify the 1-10 mol L^{-1} range as more suitable.

It is important that the factors, and levels of these factors, not only change the response variable in a measurable range but are themselves easy to control and measure accurately[194]. This ensures that the factors are being set to the proposed levels and that the results from the DOE accurately describe the behaviour within the system. Factors are classified as qualitative (also known as discrete or categorical) or quantitative (continuous).

Similarly, the response variable(s) should be accurately measurable to ensure correct results and capture meaningful variation in the experiments[194]. The variable(s) should also give a good description of the system and be directly relevant to the system's behaviour or performance goals.

Perhaps the most common DOE approach is a 2-level full factorial design in which each factor uses a low and high level and each combination of factor and level is evaluated. This results in a total number of runs equal to the number of levels to the power of the number of factors (for 2 levels and five factors: $2^5 = 32$). For statistical robustness, it is usually recommended for a design to contain 20-50 runs which equates to 20-50 data points in each replicate.

If there is a large number of factors and/or limited resources, it is common to reduce the full factorial design to a fractional design. Reducing a full factorial design removes half or three-quarters of the runs required while still allowing a description of the system to be made. Of course, it removes detail from this description and should only be done when necessary or when a detailed and full description is not required.

If deemed necessary, in order to optimise the model, a response surface methodology (RSM) design can be implemented. These are often used when factors show signs of curvature within the factorial design. Usually, only the most interesting factors are taken forward into the RSM to allow a deeper look at the effects these have on the response.

Curvature can only be fully modelled when a RSM design is employed. Centre-points are often added to full factorial designs so that curvature can be identified but not modelled. In a centre-point run, all quantitative factors are set to the mid-point between their low and high levels. This is often used to guide the design of an RSM which analyses the system in more detail following a DOE.

2.3.2 Experimental work in a DOE

Environmental factors can impact the results of a DOE; for example, if the experimental methodology is carried out in a busy lab in which the activity increases during the day and causes an increase in the lab temperature, the experiments performed in the morning and afternoon could produce different results. It is therefore common to randomise and block experiments in a DOE. Randomisation ensures that environmental factors are randomly dispersed in the data instead of, for example, lab temperature increases coinciding with concentration increasing and therefore producing a falsely

exaggerated result. Blocking ensures that experiments performed on, for example, different days, are noted so that an environmental factor which was present on one day but not the other is taken into account.

To reduce the effects which environmental factors have on the dataset, and to help identify any outliers within the data, it is important to replicate and repeat an appropriate amount of times. Replication involves performing the whole experiment, from start to finish, multiple times; repetition means performing a section of the experiment multiple times. It is recommended that repetition happens in triplicate at a minimum, but larger datasets (made of replicates and repeats) tend to improve statistical robustness.

2.3.3 Statistical analysis

During the methodology employed in Chapter 5, Minitab 2.0 software[195] was used to plan and analyse the statistics of the experiment, the details of this and the other methods which were used to aid this are given in Chapter 5. The following sections give the methodology which is used by Minitab. More information about the methodologies, parameters, and descriptions can be found on the Minitab website and accompanying documentation.

DOE methodology is commonly used to fit general linear models (GLM)[196]. The GLM is defined in Equation 2.19 in which Y is the response variable and β_i is the estimated effect factor X_i had on the response. The effect β_{ij} of factor X_{ij} represents compounded effects, or the effects which arise from different factors interacting with each other[193, 194]. Theoretically, all compounded effects up to and including the compound of all factors can be included in the model. This is unwise however, because it is unlikely that the design will have the degrees of freedom to accommodate them, and they rarely mean anything practical. For most designs, two-factor compounds are enough to provide detail of the factors' effects and any possible coupling between the factors. ϵ is the random error term, it is the difference between the observed data and the part of the data that is described by the model and it quantifies the random noise

that the model cannot explain[194].

$$Y = \beta_0 + \sum [\beta_i X_i] + \sum [\beta_{ij} X_{ij}] + \epsilon \tag{2.19}$$

Analysis

Fitting a GLM methodology employs matrices to assess all factors at once. Two matrices are required, X which tells us which level each factor was at for each response variable, and Y which lists the response variables. The responses are formatted into a column matrix Y, with the response for each run aligned to matrix X. Factor levels are coded for the calculations where a value of -1 in X denotes that the factor was at its low level, 0 shows the factor was at its centre-point, and +1 shows that the factor was at its high level. An additional column is added into X as the first column, and represents the intercept (whose effect is β_0), in which all entries in the column set to 1. For each compounded effect, an additional column is added to X with the level code being the sum of each of the components' level codes. For example, if the compounded factor for run 8 involved factor X_1 being at its low level and factor X_2 being at its high level, the 8^{th} row of the column which represented the compounded factor $X_1 \times X_2$ in X would have the value $-1 \times +1 = -1$.

Calculating effects of factors on responses

Using these matrices and Equation 2.20, we can find the empirical effect ($\hat{\beta}$) each factor produced[194]. If $X^T X$ cannot be inverted as required for Equation 2.20 this can indicate that the model is over-parametrised (i.e. a factor plus a combined factor containing the individual factor such as X_1 and $X_1 + X_2$ are included in the model), or that all interaction and polynomial terms are included but there is not enough data to fully evaluate this within the model. Removing redundant terms and combining correlated predicators can solve this.

$$\hat{\beta} = (X^T X)^{-1} X^T Y \tag{2.20}$$

We also calculate the standard error (SE) on these effect values using Equation 2.21 where the subscript ii denotes the row and column, respectively, which should be pulled from the resulting matrix $(X^TX)^{-1}$ to access the value related to the factor being assessed. MS_E is the mean squared error, SS_E is the sum of squares error, DF is the degrees of freedom, n is the number of runs, and p is the number of parameters (intercept, factors, compounded factors). The effect of each parameter quantifies how much changing the value of this parameter changes the response.

$$\begin{aligned} SE(\hat{\beta}_i) &= \sqrt{MS_E \times (X^TX)^{-1}_{ii}} \\ MS_E &= \frac{SS_E}{DF} \\ SS_E &= \sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 \\ DF &= n - p \end{aligned} \tag{2.21}$$

An example of this calculation is given in Appendix B. Statistical significance testing is now employed to ensure the effects have physical meaning in the system.

Evaluating statistical significance

A t-test is used to test for significance in each effect: the null hypothesis is that $\beta = 0$ and the factor has no effect on the response; the alternative hypothesis is $\beta \neq 0$ and the factor does have an effect on the response[194]. It is common to use a confidence level of 95%, $\alpha = 0.05$ in DOE and this is suitable for most tests where the consequences of a false positive are not severe as it balances sensitivity and specificity. When the consequences of a false positive are severe, for example when testing if a new drug has fewer side effects than an existing one, a 99% confidence level can be more suitable. If the work is exploratory or the data is limited, and the main goal is narrowing down candidates based on a set of requirements, a confidence level of 90% may be more suitable.

A t-test is a mean comparison - it tests if the mean response changes when the factor's level changes - which is why it is applicable to DOE methodology. The DF

and α values are used as the row and column, respectively, in a t-table to extract the critical t-value appropriate to the current calculation. A two-tailed test is used because this test is not concerned with the direction (positive or negative) of the effect, simply if the effect is present or not, as described in the null and alternative hypotheses.

The t-statistic for each of the effects found earlier is then calculated using Equation 2.22. If $-t_{critical} \leq$ t-statistic $\leq +t_{critical}$ then the null hypothesis is accepted and it can be concluded that changes in the factor in question do not affect the response variable in a statistically significant way. Alternatively, if the t-statistic is outwith the range of $-t_{critical}$-$+t_{critical}$, then the null hypothesis is rejected, and it is concluded that changes in the factor in question do affect the response variable in a statistically significant way.

$$t = \frac{\hat{\beta}_i}{SE(\hat{\beta}_i)} \tag{2.22}$$

A GLM is fitted to the data by using the above statistical test in an iterative methodology to build the model. The method can begin with an empty model or a full one (all effects included) and then each step can either add a significant factor, remove an insignificant factor, or both. The result is a GLM as shown in Equation 2.19 in which all effects are statistically significant as per the t-test.

It is common to use a hierarchical system when using stepwise regression which involves ensuring that the individual main effect is present in the model if any interaction effect containing the same main effect is also present. For example, if the compounded factor $A \times B$ is in the model, both individual factors A and B must also be included. This ensures that the model remains logical as you can't truly understand the effect of the interaction between factors if there is no understanding of the individual factors themselves.

Evaluating the suitability of the resulting fitted model

Due to the use of experimental data, and shown by the inclusion of an error term in the model (ϵ), the model presented here is not a deterministic physical model, but is

empirical. This means that randomness from experimental data will be included in the model and must be analysed to assess the validity of the model.

There are several ways to check the suitability of a fitted GLM and its ability to represent the dataset or predict new data. These include values which give details focussing on (1) the model's fit to current and predictive data, (2) statistical significance of the data, and (3) residual diagnostics. Each of these categories is presented below with explanations of the values used in their testing. An explanation is then given of a method which can be applied if the residual diagnostics highlight problems in the model.

Residuals are used often in diagnostic testing and are the difference between the observed value Y and the model's predicted value $\hat{Y}$ as shown in Equation 2.23.

$$e_i = Y_i - \hat{Y}_i \tag{2.23}$$

The model's fit to current and predicted data The values in Table 2.1 and Equations 2.24-2.29 describe the values which can be used to test the model's fit to current data, its level of overfitting, and its power to predict new data.

$$\sqrt{MS_E} = \sqrt{\frac{SS_E}{n-p}} \tag{2.24}$$

$$R^2 = 1 - \frac{SS_E}{SS_T} \tag{2.25}$$

$$R^2_{adj} = 1 - \left(\frac{SS_E/(n-p)}{SS_T/(n-1)}\right) \tag{2.26}$$

$$R^2_{pred} = 1 - \frac{SS_P}{SS_T} \tag{2.27}$$

$$SS_E = \sum_{i=1}^{n} \left(Y_i - \hat{Y}_i\right)^2 = \sum_{i=1}^{n} e_i^2 \tag{2.28}$$

$$SS_T = \sum_{i=1}^{n} \left(Y_i - \bar{Y}_i\right)^2 \tag{2.29}$$

Statistical significance of the data To test whether the explanation of the data

Table 2.1: Values used in determining the suitability of the fitted model to describe the current data and predict new data.

Diagnostic Value	Symbol	What It Shows	Equation	What To Look For
Standard deviation of residuals	$\sqrt{MS_E}$	How close or far the predicted values are from the observed values on average.	2.24, 2.28	A smaller MS_E indicates a better fitting model.
R-squared value	R^2	Proportion of variation in the response which the model explains.	2.25, 2.28, 2.29	A value of 1.00 shows a model which describes 100% of the data variation.
Adjusted R-squared value	R^2_{adj}	Adjusted for the number of terms in the model and so can help identify over-fitting.	2.26, 2.28, 2.29	An increase when the number of terms in the model is changed shows a better model.
Predictive R-squared value	R^2_{pred}	Shows how well the model predicts new data.	2.27, 2.28 with Y_i removed from dataset, 2.29	A value of 1.00 shows a model which predicts new data perfectly.
Prediction sum of squares	SS_P	Mimics testing the model on new data by removing the current observed value Y.	2.28 with Y_i removed from dataset	A smaller value indicates better predictive power and less over-fitting.

given by the model is statistically significant or due to chance, an F-test is performed. an F-statistic is a comparison of the variance between groups with the variance within the groups, and in the case of DOE this is the explained variance in the model divided by the unexplained variance, i.e the residuals. An F-statistic of approximately 1 indicates that the model does not explain any more variation than random noise, while a large enough F-statistic shows that the model explains significantly more than just noise.

Residual diagnostics Residual diagnostics allow the validity and trustworthiness of the model to be determined by determining not just how well the model fits but

Table 2.2: Residual diagnostics plots used to determine if a model is appropriately fitting the data.

Plot	Defining Feature of Appropriate Model	Potential Problems & What They Might Indicate
Residuals vs Fitted Values	A random scatter of points around the horizontal axis.	Curves: non-linear effects are not being captured.
		Funnels (widening/narrowing spreads): non-constant variance (heteroscedasticity).
		Outliers: large individual errors.
Normal Probability	Residuals fall along the predicted (straight) normality line.	Curved or S-shaped patterns: residuals are not normally distributed.
Histogram of Residuals	A symmetric, bell-shaped distribution.	Skewed or multimodal: misfit of model of transformation of model required.
Residuals vs Order of Data	No obvious trends.	Trend present: autocorrelation or unmodeled time-based patterns.

whether or not the fit is appropriate. Even if R^2 is high, if the residuals are bad, the model is bad. Good models have residuals which look like random noise with no patterns, trends, or structure. There are commonly four different indicator plots which can be used to determine the suitability of the model fit using residuals: residuals vs fitted values, normal probability plot, histogram of residuals, and residual vs order of data. For a well-fitting and appropriate model, each of these plots will have the defining features presented in Table 2.2.

Heteroscedasticity (mentioned in Table 2.2) refers to a non-constant variance in the errors, for example, if the errors are small for low predicted responses and large for high predicted responses, there is heteroscedasticity. Having roughly the same-sized errors throughout the dataset is called homoscedasticity and implies a better fitting model. Another point mentioned in Table 2.2 is autocorrelation. This term refers to external factors which were not documented or included in the statistical analysis but have affected the results. An example is performing a temperature dependent experiment in a room in which the temperature rises and falls throughout the day. The results

may change with what seems like no reason within the parameters of the experiment because the room temperature is affecting the results. This is why experimental run order is important in DOE, having the order noted allows patterns like this to emerge.

Transformation to improve model's suitability For problems with heteroscedasticity and a skewed or multimodal histogram of residuals, performing a transformation on the data can reduce these problems and produce a more suitable model. This involves altering the response dataset to gain a new dataset with the aim of improving the model's fit. During these transformations it is possible that terms which were significant pre-transformation, especially higher-order terms, will no longer be significant and are therefore removed from the model. This happens because the transformation can make the variance in the data more consistent and therefore reduces the need for complex interaction terms such as second- or third-order terms.

A common transformation methodology is the Box-Cox transformation, the formulae for which can be seen in Equation 2.30 where Y' is the transformed response. Maximum Likelihood Estimation (MLE) is commonly used to estimate a value of λ which best fits the assumption of normality and homoscedasticity. The relevant formula from Equation 2.30 is applied to the response variable dataset, and the model is refitted with the new response variable data. The model fit and testing can then be repeated on the transformed data to, hopefully, produce a model with better residual diagnostics.

$$\begin{aligned} Y' &= \frac{Y^\lambda - 1}{\lambda} \text{ for } \lambda \neq 0 \\ Y' &= \log(Y) \text{ for } \lambda = 0 \end{aligned} \tag{2.30}$$

While transformations such as the Box-Cox can create better residuals in a model, they should be used with care. Things to consider include: (i) once transformed, the response variable data is no longer in the original scale and may be more difficult to interpret, (ii) the need for transformation could also be an indication of a more serious underlying problem such as a missing interaction term or use of the wrong model, and (iii) λ is often approximated from a broad range of possible λ values and an extreme

outlier can shift the optimised λ value within this range.

2.3.4 Strengths and weakness of DOE methodology

DOE methodology offers many strengths. The systematic approach of DOE allows the analysis of the effects of multiple factors and their interactions, and the comparison between the magnitude of these effects. This means that a good description of a system is possible through one set of experimentation. For experiments which heavily consume resources (for example those that take a long time, use expensive reactants, or consume large volumes of feedstocks), a DOE can be made more efficient by reduction of the number of runs (fractional design) while still gaining a clear description of the system. Following the DOE methodology, there are quantifiable effects and uncertainties which formalise the results and leave less for interpretation. DOE methodology also allows the optimisation of a system to specific criteria through the model which is produced and can increase the reproducibility of results.

The weaknesses of DOE methodology by no means outweigh the strengths, but they are important to note. DOE methodology can be resource intensive even with a fractional design, and the resources required extend beyond the experimental set-up, into planning and statistical knowledge. DOE methodology often assumes linear models, such as a GLM, which can miss detail and result in misleading outcomes. Due to this, a follow up RSM is often required to fully describe the system or create a predictive model. Overall, even with its weaknesses, DOE is a powerful tool for learning about a system and the ways its variables can affect the final product.

Chapter 3

Calculating & Optimising Alginate Atomic Partial Charges For Simulation

3.1 Introduction

As outlined in Section 2.2.2, the generation of partial atomic charges for alginate molecules is required before simulation can begin. In this chapter, we aim to determine a set of partial charges that can be used for alginate chains of any length and composition.

Monomers in alginate chains are rarely uniform, as can be seen in the example given in Figure 3.1, where each monomer has a slightly different configuration which will change as the chain moves and interacts with its environment. If a general alginate atomic partial charge list can be produced, these configurational differences must have minimal effect on charges.

Alginate chains can be considered to be constructed from three separate monomers: 4'-end, middle, and 1'-end as shown in Figure 3.2. If a general alginate atomic partial charge list is realistic, it is important that each of these three monomer types retains the same atomic partial charges in any chain length.

There are, therefore, three main ways in which an alginate chain can be altered which could affect the atomic partial charges: (1) geometric fluctuations in individual monomers, (2) the number of monomers which make up the chain, and (3) whether the chain is made of G or M residues. M-G bonding was not used in this project and so is

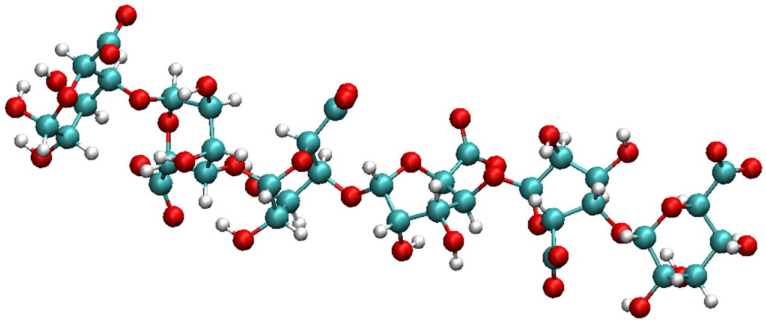


Figure 3.1: A screenshot from VMD showing a hexa-G alginate chain as it looks in a simulation where cyan spheres represent carbon atoms, red spheres represent oxygen atoms, and white spheres represent hydrogen atoms.

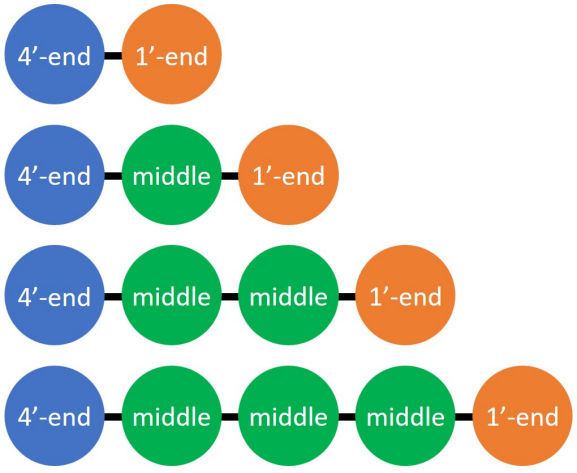


Figure 3.2: From top to bottom: di-, tri-, tetra-, and pentamer constituent monomers.

not expressly investigated in this study. To ensure full compatibility for any alginate study, the total charges for the 4'-end and 1'-end monomers combined must be -2 and the middle monomers must have a charge of -1 each. On average, therefore, the alginate chains have a charge of -1 per monomer. This chapter aims to determine whether this is feasible without losing the charge details of each monomer in the chain. If this is not deemed feasible, each chain used in simulation will require its own partial charge calculation.

We will first discuss the methodology used to achieve this aim before presenting the results from studies of the geometric monomer fluctuations, chain length, and G vs M effects of the partial atomic charges.

3.2 Methodology

This section will present the methodology for calculating partial atomic charges before discussing the collection of alginate structures for use in this study.

3.2.1 Methodology for calculating partial atomic charges

The AM1-BCC method was chosen to calculate partial atomic charges. This method employs the Austin Model 1 (AM1), a semi-empirical quantum chemistry method, followed by a Bond Charge Correction (BCC). AM1-BCC is an estimation of full quantum methods like the Restrained Electrostatic Potential (RESP), offering advantages of speed and lowered computational expense[197–199]. This method was designed for, and tested on, biological and organic systems, including carbohydrates similar to alginate, and was used in the development of the GLYCAM-06j force field, making it ideal for implementation in this work[197, 198].

The AM1-BCC method is made up of two steps, as discussed above. First, the AM1 method is applied to calculate the Mulliken charges, and then a correction is applied which produces charges that better represent those calculated using RESP methodology.

The Mulliken charge on an atom is defined as shown in Equation 3.1 where $q_i^{Mulliken}$

is the Mulliken charge, Z_i is the atomic number of atom i, μ and v are basis functions (atomic orbitals), $P_{\mu v}$ is the density matrix elements, and $S_{\mu v}$ is the overlap matrix elements.

$$q_i^{Mulliken} = Z_i - \sum_{\mu \in i} \sum_{\nu} P_{\mu\nu} S_{\mu\nu} \tag{3.1}$$

The basis functions in Equation 3.1 are mathematical functions used to approximate atomic orbitals and build molecular orbitals[197, 200, 201]. Each orbital is represented by one basis function. For simplicity, only the 2s and 2p orbitals will be discussed here. The 2s orbital is spherical and so the function models the electron density radially. The 2p orbital is dumbbell shaped and is therefore split into $2p_x$, $2p_y$, and $2p_z$ so that the bonding in each direction of the orbital can be modelled.

A simplified version of the Schrödinger equation is used; the full Schrödinger equation is not solvable for any molecules larger than diatomic hydrogen due to the number of electrons and complex electron-electron interactions[197, 200, 201]. The general time-dependent Schrödinger equation is given in Equation 3.2 where $\hat{H}$ is the Hamiltonian operator which describes kinetic energy and repulsive and attractive forces, Ψ is the wavefunction and describes the behaviour of electrons, and E is the eigenvalue and describes the total energy of the system[197, 200, 201]. An example calculation of Mulliken charges is given in Appendix C.

$$\hat{H}\Psi = E\Psi \tag{3.2}$$

The AM1 method simplifies Schrödinger's equation using experimental data and pre-parameterized integrals. This is where much of the speed increase in AM1 methodology compared with quantum methods arises. The downfall of Mulliken charges is that the method shares the overlap density equally between the two atoms in the bond and therefore can give false estimates of partial charge[197]. To address this, Jakalian *et al.* added the Bond Charge Correction (BCC) into the methodology which helps make the Mulliken charges match the RESP charges more closely.

Jakalian *et al.* published a list of bond charge corrections (BCC) which can be applied directly to the atom charges in question as shown in the example calculation

in Appendix C[198]. Where one atom is involved in multiple bonds, the sum of all applicable BCC is added to the atom charge. This results in a list of partial charges for every atom which more closely resembles the results produced in a RESP methodology.

The atomic partial charges were calculated using AM1-BCC in the AMBERTools21 antechamber software[187]. The software requires the user to define a net molecule charge which was taken to be -N where N is the number of monomers in the chain being studied. This assumes that alginate is completely deprotonated and each carboxylate group bears a -1 charge which is consistent with its presence as a salt with monovalent ions such as Na^+ and with its ability to cross-link in the presence of divalent ions such as Ca^{2+}.

3.2.2 Obtaining alginate structures with monomer geometric fluctuations

To study the effects of monomer geometric fluctuations on the partial atomic charges, structures of alginate were retrieved from the Protein Data Bank (PDB). The PDB is an online database containing experimentally determined structures of large biological molecules such as DNA, RNA, and proteins, and their systems (which sometimes contain alginate oligomers) [202]. The PDB codes for G and M residues are "LGU" and "BEM", respectively. Since these structures are taken from experimental systems, the monomers have natural geometric differences from bonded and non-bonded interactions with their environment.

The longest chain lengths available for the two residues were a 6-mer of G and a 7-mer of M; using longer chains means more monomer configurations and therefore a more extensive study. Details of the two systems used are given in Table 3.1.

To extract the alginate chains from these systems, the structure files available on the PDB website were uploaded to the Visual Molecular Dynamics (VMD) software[206]. System 7BZ0, described in Table 3.1, is shown in VMD in Figure 3.3. The alginate chain was selected and saved separately. The alginate structure in the original 7BZ0 system did not include any hydrogen atoms so the alginate-only coordinate file was uploaded to the Avogadro2 software[207] which allows the easy addition of hydrogen atoms. The

Table 3.1: Protein Data Bank systems used to gain alginate chains with monomeric configurational differences.

Alginate Molecule in System	6-G	7-M
PDB System Identifier Code	7BZ0	7NCZ
Date Released	10-03-2021	02-03-2022
Published Work	[203]	Not Applicable
PDB Page	[204]	[205]

software fully protonated the molecules so it was necessary to delete the two carboxylate hydrogen atoms on each monomer to ensure the molecule represented alginate and not alginic acid. Lastly, the bonding details were added to the resulting structure file. Without the information of which atom is bonded to which, the antechamber software will estimate it based on bond lengths which could result in errors in the partial atomic charges.

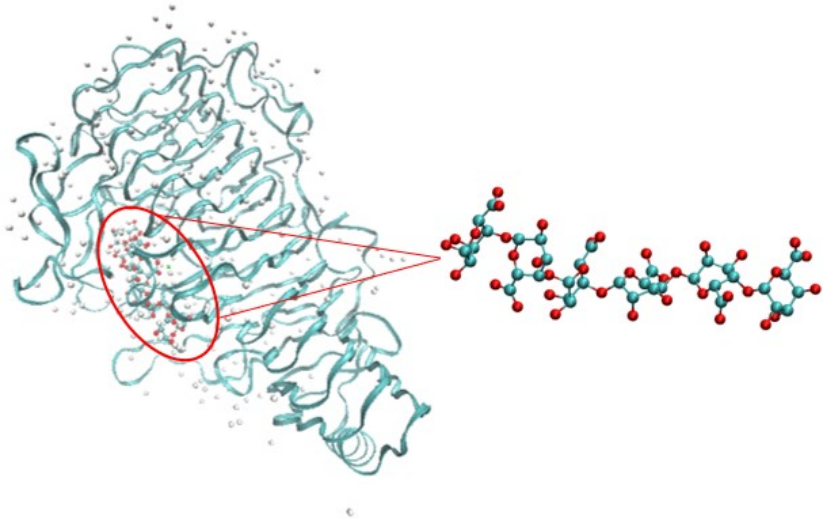


Figure 3.3: The full 7BZ0 PDB system visualised in VMD with the deprotonated hexa-G chain highlighted.

To allow the investigation of all four monomer types: full, 4'-end, middle, and 1'-end, these chains were split into monomers, dimers, and trimers of neighbouring monomers. Addition of hydrogen atoms where glycosidic bonds had been broken was achieved using the Avogagro2 software.

3.3 Results

The results are sectioned by the three main ways an alginate chain can be altered to affect the partial atomic charges: (i) monomer geometric fluctuations, (ii) chain lengthening, and (iii) G versus M residue chain identity. A final section is added following these investigations which describes how the results were applied to simulation work. The atom labelling used in all of the results of this study is as shown in Figure 3.4.

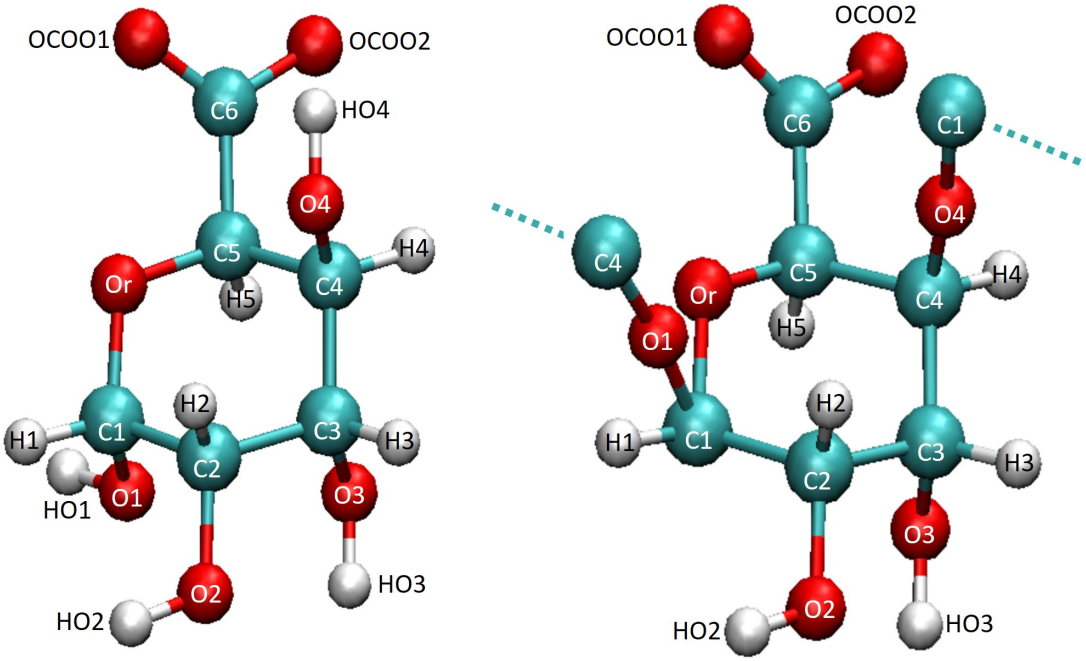


Figure 3.4: An alginate free and chain middle monomer with atom labelling.

3.3.1 The effects of geometric fluctuations

Examples of the monomer structures used in this study are shown in Figure 3.5; all of the structures can be seen in Appendix D, Figures D.1 and D.2.

The monomer shown in the top right of Figure 3.5 deviated the furthest from a planar ring formation of all monomers studied. The majority of the structural differences seen between all monomers was the positions of the hydrogens, especially those

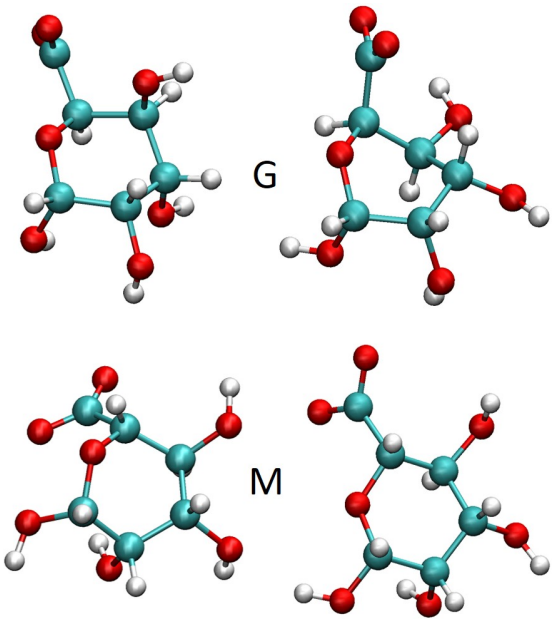


Figure 3.5: Examples of the (top) G and (bottom) M monomers used to the study the effects of monomer geometric fluctuations on the partial atomic charges.

in hydroxyl groups. These structural differences had minimal effects on the partial charges, as seen in Figure 3.6 in which the average and standard deviation (error bars) are plotted. The biggest standard deviation values were produced on the H2 and H5 atoms and most of this deviation from the average was contributed by the G monomer highlighted previously for its non-planar ring (Figure 3.5, G, right). The deviation of each individual monomer can be seen in the more descriptive data included in Appendix D, Figures D.3 and D.4. Due to the relatively small standard deviation values of the partial atomic charges across the different monomers studied, it can be concluded that the effect of geometric fluctuations on the partial atomic charges of free monomers is minimal and can be disregarded.

Examples of the different dimer configurations used in this study are shown in Figure 3.7; all of the structures studied can be seen in Appendix D Figures D.5 and D.6.

In addition to the hydrogen atom position differences and deviations from a planar ring group seen in the monomers, the dimer structures also included monomer-monomer configurational differences. The angle between the direction of each monomer's car-

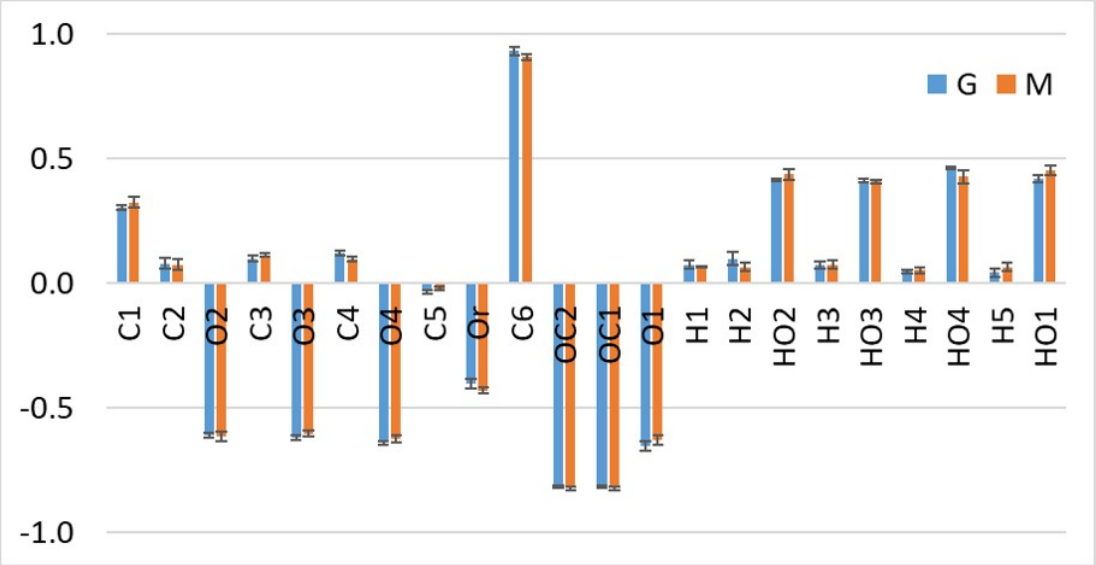


Figure 3.6: Average partial atomic charges of all G and M free monomers studied.

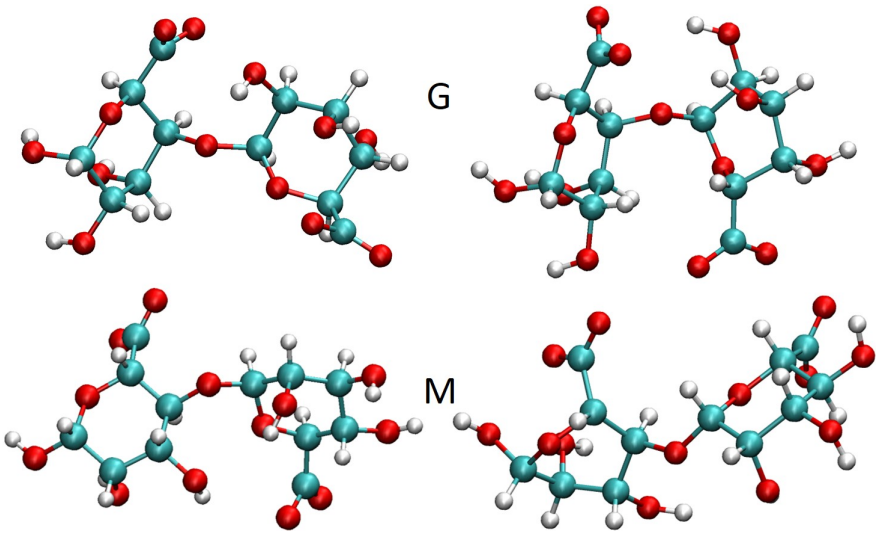


Figure 3.7: Examples of the (top) G and (bottom) M dimers used to the study the effects of monomer geometric fluctuations on the partial atomic charges.

boxylate group changed from dimer to dimer, as did the angle between the planes of each of the monomer's ring groups. This, however, did not produce any significant deviations from the average partial atomic charges seen on each atom as shown in the standard deviations plotted in Figure 3.8. The individual data for each dimer studied can be seen in Appendix D Figure D.7 and D.8. The effect of geometric fluctuations in

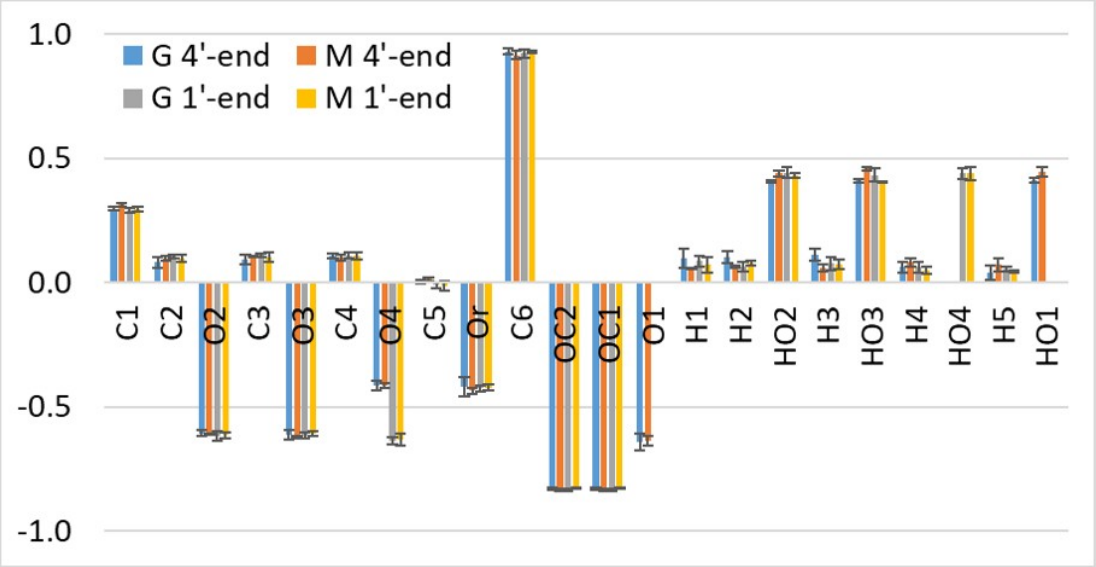


Figure 3.8: Average partial atomic charges of all G and M dimers studied. The 4'- and 1'-end monomers are presented separately.

dimers on the partial atomic charges was therefore deemed to be negligible.

Lastly, examples of the different trimer configurations used are shown in Figure 3.9; all of the structures studied can be seen in Appendix D Figures D.9 and D.10.

The additional monomer included in these structures, compared with the dimers, added a further level of complexity in the differences seen between each molecule and this did affect the hydrogen atoms' partial atomic charges more than seen in the monomer or dimer investigations. This is shown in Figure 3.10 where the hydrogen atoms have larger standard deviations (error bars) than in Figures 3.6 or 3.8. The partial atomic charges of all structures studied are included in Appendix D Figures D.11 and D.12. Since these larger deviations were only seen for hydrogen atoms, which are less influential in simulation than heavy atoms, the effect of configurational differences in trimers on partial atomic charges was deemed negligible.

For monomers, dimers, and trimers, the variation in G structures' partial atomic charges was less than that of M. This is theorised to be due to the carboxylate placement with respect to the hydroxyl groups. The M carboxylate direction allows it less interaction with the C4 hydroxyl group than the G residue. This decreased interaction allows for less differences to form in the bonding structure of each monomer and

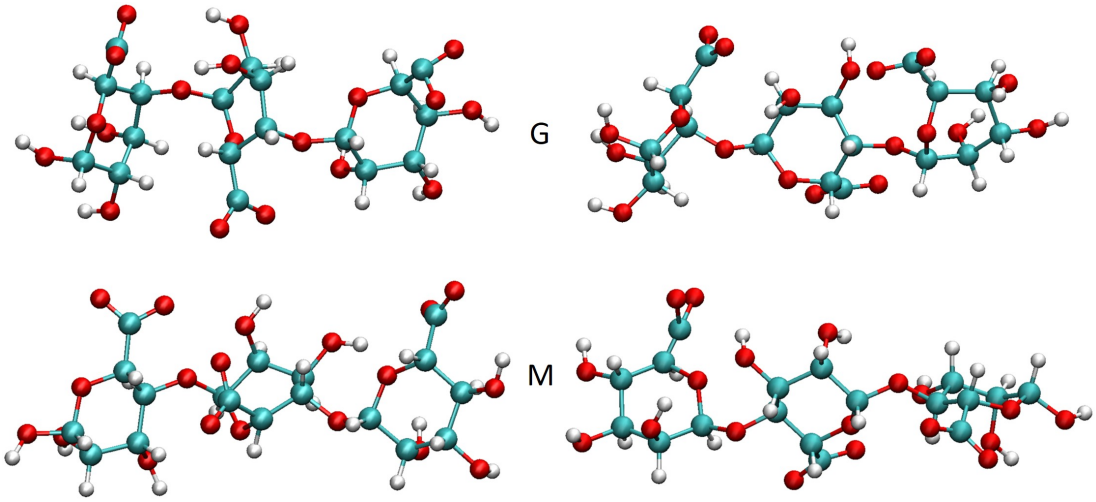


Figure 3.9: Examples of the (top) G and (bottom) M trimers used to the study the effects of monomer geometric fluctuations on the partial atomic charges.

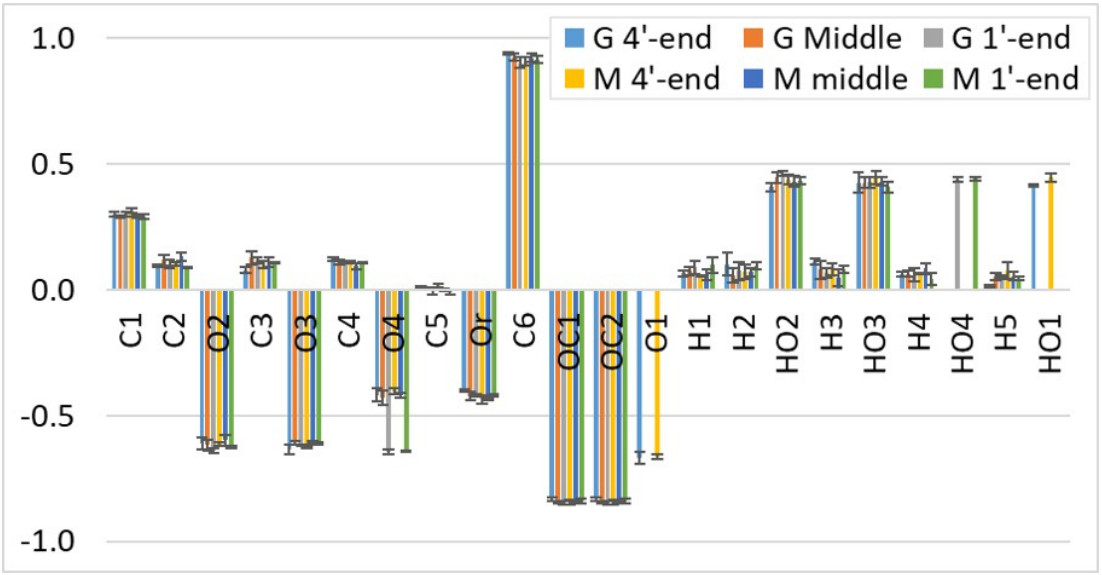


Figure 3.10: Average partial atomic charges of all G and M trimers studied. The 4'-end, middle, and 1'-end monomers are presented separately.

therefore the partial atomic charges.

Since no geometric fluctuations between the structures studied produced a significant effect on the partial atomic charges, it was decided that the affect could be deemed negligible. Due to this, to reduce the number of variables, and to allow free customisation of chain length, ordered chains were used in the remainder of this study. A python script was compiled which creates structural files of alginate chains. This ensured that reasonable initial configurations could be created for customisable (in chain length and residue) alginate chains. This script creates structured and straight chains which have a 180° flip in the direction the carboxylate group points between each monomer, as seen in Figure 3.11. The script is included in Appendix E and contains the charges that resulted from this chapter.

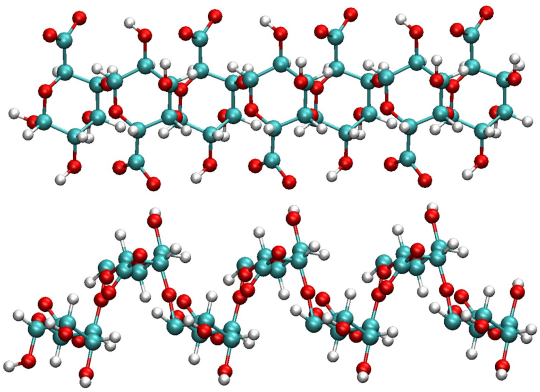


Figure 3.11: An example of the ordered and highly buckled chains which were produced using the python script included in Appendix E.

3.3.2 The effects of chain lengthening

This part of the study was conducted to test whether the charge distribution along a chain, and within the three different monomer types (4'-end, middle, and 1'-end), is different in chains of different lengths. Chains of length 2-7 monomers were created from both G and M monomers using the molecule building script described above.

The results from this part of the study can be seen in Figure 3.12. The variation in these values, as shown with the standard deviations (error bars) in Figure 3.12, was similar to the values seen in the geometric fluctuations analysis. The charges of each

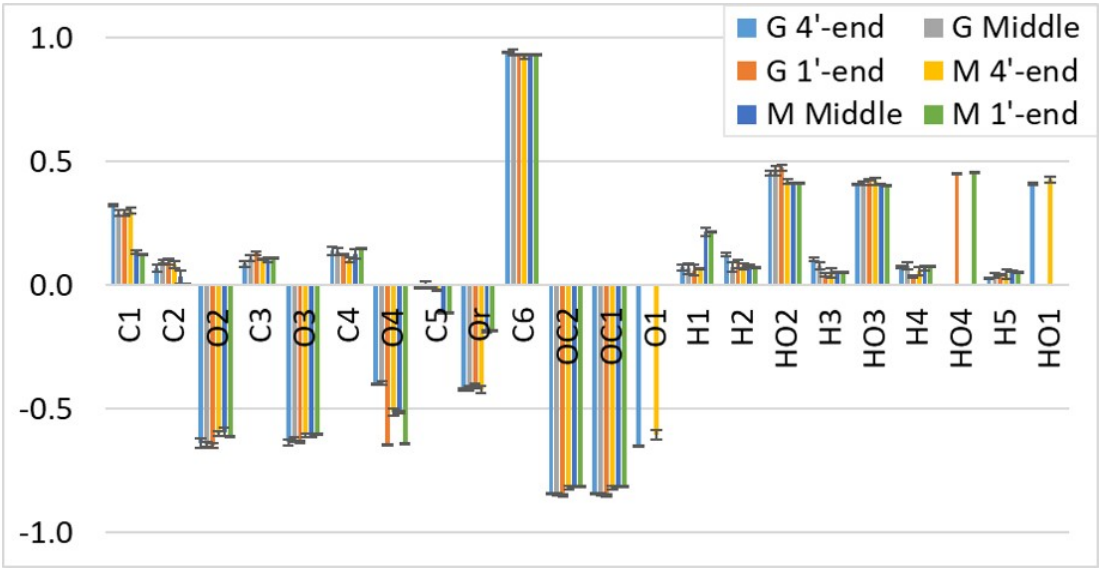


Figure 3.12: Average partial atomic charges of G and M 4'-end, middle, and 1'-end monomer types from chains of length 2-7 monomers.

individual structure studied are included in Appendix D Figures D.13- D.15. Although the effect of chain lengthening is minimal when looking at the full range of residues and chains, there were trends in some specific atoms as chain length increased. The most prominent of these trends is shown in Figure 3.13 which shows the partial atomic charge of the G 4'-end monomer C3 atom in chain lengths of 2-7; as chain length increased the 4'-end C3 atom's charge decreased from 0.108 to 0.076 in a way which suggests the charge of this atom would plateau as chain end effects were minimised. Similar trends occurred on several G atoms but were not as prominent in M atoms. While the ranges of charges produced in these trends are small, this should be considered when simulating alginate chains of less than 7 monomers in length.

While this trend is visible when presented as the data in Figure 3.13, overall the fluctuations are minimal, as can be seen in Figure 3.12 in which the standard deviation values are negligible in comparison to the partial atomic charge values. It can be concluded, therefore, that the effects of chain lengthening on each monomer type (1'-end, middle, and 4'-end) is negligible.

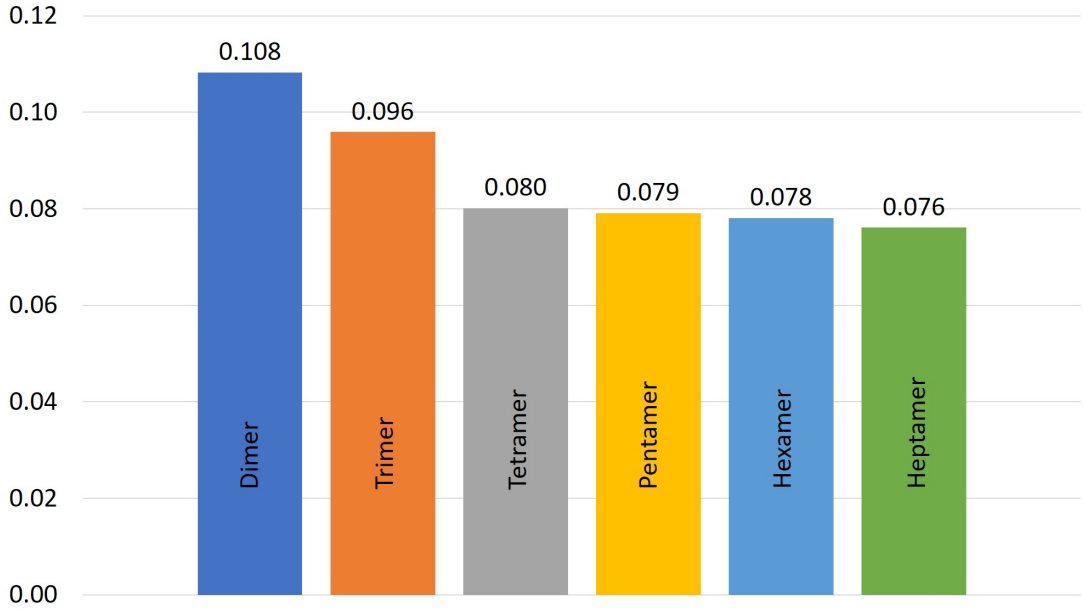


Figure 3.13: Average partial atomic charges on 4'-end monomer G C3 atoms in chains of length 2-7.

3.3.3 Differences between M & G residues

It is unclear from the configurational and chain lengthening effects whether or not G and M residue charges are interchangeable; a study was performed to compare the residues. The similarity of the charges on M and G monomers is vital in determining limitations in chain pattern design, e.g. whether an M-G-M-G chain can be built in the same way as an M-G-G-M chain, with emphasis on the differing or identical end monomers. While it is not required that M and G monomers have identical atomic partial charges, differences in total monomer charges would add complexity in creating mixed chains.

The charges from the geometric fluctuation and chain lengthening parts of this study were compiled and compared to determine the differences in atomic partial charges in M and G residues. The results can be seen in Figure 3.14. Most atoms have very similar charges between the two residues, for example, the full monomer atoms C1, O2, C3, C4, C5, OCOO1, and OCOO2 in Figure 3.14 are almost identical between the M and G residues. Atoms which have significant differences in the charges between the M

and G residues include O4 in the 4'-end and middle monomers, and H1 in the 1'-end monomers, where the biggest charge difference is about 0.25. Major differences were most common in the atoms closest to the carboxylate group (C1, C5, O1, O4, and Or) which is expected since this group's position differs between the two residue types.

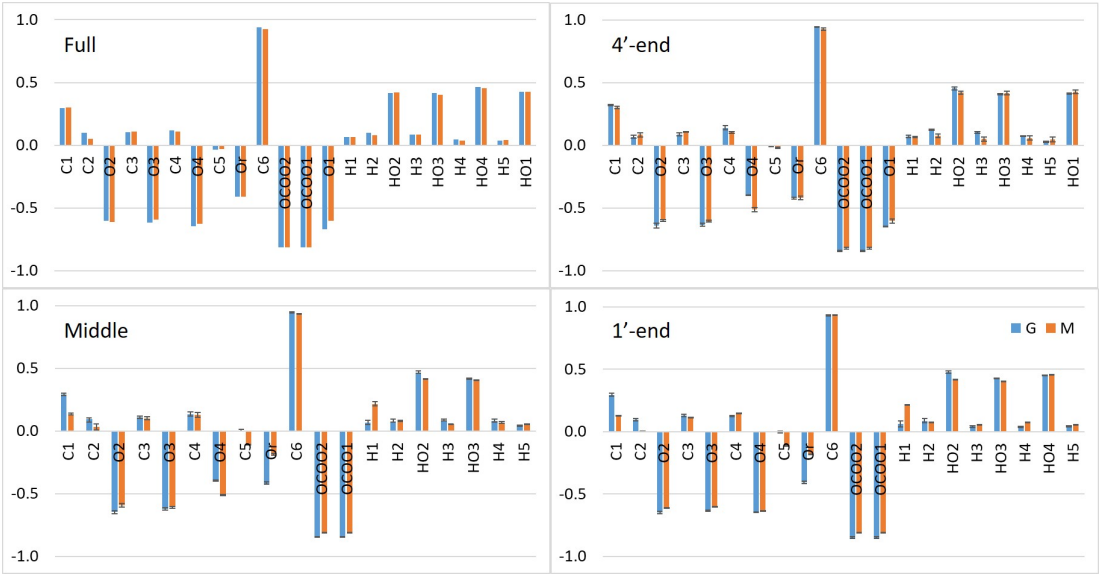


Figure 3.14: The atomic partial charges for each atom in the full, 4'-end, middle, and 1'-end monomers showing the differences in atomic partial charges between M and G residues. The lack of standard deviation on the full monomer graph is because only 1 structure was used for each.

Due to the scale of the biggest differences found, and in the interest of conserving the detail of differences between the residues, it was concluded that the atomic partial charges are not identical within M and G residues and that the atomic charge lists should be kept separate for the two residues.

3.3.4 Adapting the charges for use in simulation

To end this study in a way that allows the findings to be integrated directly into simulation work, the resulting atomic partial charge lists and total monomer charges were evaluated to maximise the usefulness of the charges. For simulation creation it is important that the two end monomers have a total combined charge of -2 and any middle monomers have a charge of -1; this way any chain length from a dimer upwards

can be created with no changes to the atomic partial charge lists. It can be seen in Table 3.2 under "Before Scaling" (which shows the averaged total monomer charges) that the middle monomers do lend themselves to being rounded to a -1 total monomer charge, and that the two sets of two end charges, while different, both round to a total of -2. However, with the charges presented in "Before Scaling" in Table 3.2, M and G residues could not be combined in a dimer or as end monomers in a chain because the 1'-end and 4'-end monomer total charges for M and G residues do not add up to -2 in any combination. This implements limitations within the simulation set-up of possible mixed residue chains, which would not be able to have different residue types at either end of the chain. Because of this, and because the net monomer charges are not significantly different, it was decided to scale both M and G end charges to a middle point so that they could be used in mixed residue chains. The adapted total monomer charges can be seen in Table 3.2 under "After Scaling". This scaling method was deemed the best way to optimise the atomic partial charge lists as it keeps M and G charges separate, and allows for the variation between M and G residues seen in Figure 3.14, but it also means that any length and pattern of M/G/MG chain can be created. This total monomer charge scaling is enacted by scaling each individual atomic partial charge.

Table 3.2: The averaged total charges for the three different types of monomer found in chains for M and G residues before and after scaling.

Monomer	**Before Scaling**		**After Scaling**	
	M	**G**	**M**	**G**
4'-end	-1.332	-1.201	-1.250	-1.250
Middle	-0.997	-0.982	-1.000	-1.000
1'-end	-0.674	-0.844	-0.750	-0.750

The free monomer charges are not included in Table 3.2 because the method of gaining the atomic partial charges on a free monomer will always produce a total monomer charge of -1. The full list of partial atomic charges for each residue and monomer type, scaled as described above, is presented in Table 3.3. A comparison between the charges calculated in this study and those found in literature is presented in Appendix D Table D.1.

Table 3.3: The eight charge lists for M and G monomers and chains calculated and optimised for use in simulated alginate systems. Blank spaces are left where the atom does not exist in the monomer type. The monomers are labelled as "Full", "1", "Mid", and "4" which stand for full, 1'-end, middle, and 4'-end, respectively.

Atom	**Atomic Partial Charges in Different Monomer Types**							
	G Full	**G 4**	**G Mid**	**G 1**	**M Full**	**M 4**	**M Mid**	**M 1**
C1	0.2955	0.3364	0.2978	0.2645	0.2996	0.2877	0.1356	0.1415
C2	0.0973	0.0621	0.0978	0.0797	0.0520	0.0860	0.0349	0.0046
O2	-0.6030	-0.6727	-0.6571	-0.5755	-0.6132	-0.5670	-0.5894	-0.6806
C3	0.1053	0.0810	0.1117	0.1157	0.1070	0.1007	0.1004	0.1272
O3	-0.6160	-0.6631	-0.6341	-0.5605	-0.5922	-0.5769	-0.6073	-0.6689
C4	0.1193	0.1556	0.1393	0.1065	0.1090	0.0942	0.1257	0.1674
O4	-0.6451	-0.4136	-0.4038	-0.5678	-0.6282	-0.4875	-0.5109	-0.7100
C5	-0.0370	-0.0045	0.0027	-0.0005	-0.0319	-0.0176	-0.1093	-0.1237
Or	-0.4094	-0.4322	-0.4239	-0.3528	-0.4132	-0.4027	-0.1871	-0.2012
C6	0.9404	0.9759	0.9578	0.8175	0.9268	0.8774	0.9328	1.0434
OC2	-0.8169	-0.8724	-0.8612	-0.7505	-0.8135	-0.7787	-0.8099	-0.9038
OC1	-0.8169	-0.8724	-0.8612	-0.7505	-0.8135	-0.7787	-0.8099	-0.9038
O1	-0.6691	-0.6685			-0.6012	-0.5765		
H1	0.0648	0.0810	0.0657	0.0442	0.0666	0.0619	0.2152	0.2406
H2	0.0989	0.1247	0.0768	0.0815	0.0806	0.0699	0.0796	0.0833
HO2	0.4158	0.4760	0.4719	0.4253	0.4226	0.3999	0.4146	0.4651
H3	0.0829	0.1087	0.0792	0.0338	0.0836	0.0427	0.0552	0.0610
HO3	0.4168	0.4186	0.4220	0.3750	0.4016	0.4013	0.4056	0.4494
H4	0.0478	0.0744	0.0795	0.0301	0.0377	0.0598	0.0670	0.0867
HO4	0.4659			0.3976	0.4545			0.5086
H5	0.0348	0.0271	0.0391	0.0367	0.0417	0.0464	0.0572	0.0632
HO1	0.4279	0.4279			0.4236	0.4077		
Total	-1.0000	-1.2500	-1.0000	-0.7500	-1.0000	-1.2500	-1.0000	-0.7500

3.4 Conclusions

It can be concluded from this study that the configuration and length of an alginate chain have negligible effects on the atomic partial charges within that chain. The effects of differences between M and G residues on the atomic partial charges cannot, however, be ignored, and these must be accounted for in the atomic partial charges used in simulation. Although the atomic partial charges in the M and G residues were too different to be deemed the same, their total monomer charges were similar enough to be averaged, which allows any pattern of M/G/MG chain to be created and used without a new atomic partial charge calculation needing to be performed.

The result of this study consists of atomic partial charge lists for M and G full, 1'-end, middle, and 4'-end monomers. The 1'-end and 4'-end monomers always combine to give a total charge of -2, and the full and middle monomers have a total charge of -1. From this study, any alginate chain, whether it be pure M, pure G, or mixed residue can be created in any length to be used in simulation work.

Once calculated and optimised these charges were implemented into molecular dynamics simulations of alginate systems.

Chapter 4

A molecular dynamics investigation of oligomer structure and cation behaviour in pure and blended alginate systems

4.1 Introduction

While the "egg-box" model is widely accepted in experimental work, many computational studies now dispute it, instead showing that cation coordination is generally un-specific and presenting as dense ion condensation around the alginate chains, with no well-defined coordination sites. Where simulated annealing is applied to find minimised structures, specific Ca^{2+} binding is seen, likely accounted for in the single- or limited-structure output of this method[149–152]. Many of these studies look at single alginate chains which are somewhat restricted by, for example, implementation of an infinite chain model which may hinder the formation of the "egg-box" model[150, 152]. These studies often report at least two shells of ions forming around the alginate chains with Na^{+} ions interacting more with M chains and Ca^{2+} interacting more with G, with a much weaker association for either cation with G-M chains[149]. Perić-Hassler and Hünenberger (2010)[151] found that chain conformation was insensitive to the ion environment however Agles and Bourg (2023)[182] found that the structure of the hydrogel system as a whole was highly sensitive to the identity of cations present, suggesting that the alginate chains themselves are unchanged but the structures formed by the chains together are altered by the ion environment.

Water is an important medium in alginate systems; dried sodium alginate solutions and calcium alginate gels are brittle because the flexibility and polymer-mobility increasing, plasticising effects of water are reduced or removed [27]. Agles and Bourg (2023)[182] found that water behaviour in alginate systems was only weakly sensitive to cation identity and that alginate-rich and -poor phases occurred no matter what counterion was present. Li *et al.* (2021)[152] found that Ca^{2+} presence excluded water from close proximity to the alginate chains, going some way to explaining the insolubility of calcium alginate in water.

Limited studies are published that employ molecular dynamics simulations to enable an atomistic view of alginate systems, the differences between M and G residues, the effects of ions, and the role of water. Previous studies using MD to investigate alginate systems have found that alginate chains are, in general, flexible, with regular helical structures being in the minority[150]. Some studies choose to analyse only poly-G chains due to their well-known crosslinking strength[151, 152], or only poly-M chains to exclude the stronger G-Ca^{2+} interactions[182]. Studies which compared the two residues found that increasing M content in alginate chains has been found to increase flexibility with poly-M and poly-G chains showing more rigidity than MG chains[148, 149]. It has been shown that aggregation of alginate chains is predominantly affected by the chain composition, with association clearly dependent on G content, although with M chains promoting this association[148, 149]. The role of MG blocks in cation binding is underestimated in comparison[181].

While these studies look extensively at co-polymer alginate, in which M and G residues are bonded to each other, there is a lack of analysis on alginate polymer blends, in which the G and M chains are mixed but not bonded. Pairing analysis of these blends with analysis of only G or only M chains could give more insight into the M-G interactions and how they help or hinder alginate interactions with other alginate chains, cations, and water.

This study used MD simulations to investigate the effect of i) monomer type and ii) ion type. First, three systems with either 10 G trimers, 10 M trimers, or 5 G and 5 M trimers, 30 Na^{+} ions, and 0.016 g alginate/g water concentration were compared

to analyse the effects of alginate environment on the trimer conformations, monomer bonding structures, water and ion placements around the trimers, trimer clustering, and species mobility. The same properties were then investigated for three further systems with 15 Ca^{2+} ions instead of 30 Na^{+} ions, and the same alginate and water environments to provide a comparison between the cation behaviours. The use of small chains in these systems allows the full flexibility of the glycosidic monomer linkages, and full accessibility of cations and water to all parts of the alginate chains. This allows the analysis of alginate conformation/structure, and detailed analysis of cation and water proximity to the chains. Keeping the G and M trimers separate in systems or with no bonding between the monomer types allows an analysis of how a polymer blend may impact the properties of alginate.

4.2 Methodology

Molecular dynamics (MD) simulations were run using the Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS)[153]. Six different systems were investigated in this study. Each had ten alginate trimers, either all G, all M, or a mixture of five of each G and M. The systems also contained either 30 Na^{+} or 15 Ca^{2+} counterions, and 18,851 water molecules to produce an alginate-in-water concentration of 0.016 g alginate per g water. This concentration is within the commonly used range of alginate in water concentration for creating alginate gels and films. The details of each system, along with the code assigned to it, can be seen in Table 4.1.

Table 4.1: The number of each type of trimer and ion in the six different systems studied. All systems also include 18,851 water molecules.

System	Trimers		Ions	
	G	M	Na^{+}	Ca^{2+}
G10-Na	10	0	30	0
M10-Na	0	10	30	0
G5M5-Na	5	5	30	0
G10-Ca	10	0	0	15
M10-Ca	0	10	0	15
G5M5-Ca	5	5	0	15

Alginate was modelled using the GLYCAM-06j carbohydrate-specific Amber force field which is discussed in more detail in Section 2.2.2. Ions were modelled using the Amber-recommended parameters from Li *et al.* [162, 178, 179], and the TIP3P model was used to model water. The parameters for all of these models are included in Appendix A. Atomic partial charges for the alginate trimers were computed as described in Chapter 3. The Na^+ and Ca^{2+} cations within the systems bring the total system charge to zero.

The trimers were numbered by their order of addition into the simulation box, 1 to 10, and are labelled as such throughout this study. Alginate trimers were inserted with random position and rotation into a cubic box of length 100 Å, followed by the water and ions, also randomly inserted. A single starting conformation was used for the G systems, and another for the M systems; these configurations were created using the python script included in Appendix E which is discussed in more detail in Section 3.3.1. Due to the small length of these molecules, each system was relaxed as a whole.

Each system was energy minimised using the LAMMPS Polak-Ribiere conjugate gradient algorithm with energy and force stopping tolerances of 1×10^{-4} kcal mol^{-1} and 1×10^{-6} kcal mol^{-1} Å^{-1}, respectively. Systems were first run with a 1 fs time step for 10 ps NPT in 1 ps groups followed by an unbroken 10 ps to allow the particle-particle particle-mesh (PPPM) solver grid to adapt to the reducing box size. The systems were then run for 41 ns NPT, with trajectories being output every 10 ps. The equations of motion were integrated using the velocity-Verlet algorithm. The SHAKE algorithm was used to fix the water molecules' bonds and angle using an accuracy tolerance of 1×10^{-4} and a maximum of 20 iterations in each SHAKE solution. Pairwise interactions were computed with distance cutoffs of 12 Å for both Lennard-Jones and Coulombic interactions. PPPM was used to compute long-range electrostatic interactions. The Nose-Hoover thermostat and barostat were employed to maintain a system temperature of 300 K and pressure of 1 atm.

End-to-end distances (R_e) of the trimers were calculated by averaging the distance between atoms O1 (4'-end) and O4 (1'-end) where atoms are labelled as shown in Figure 4.1. The angles between the glycosidic linking C4-O4-C1 bonds in the trimers were

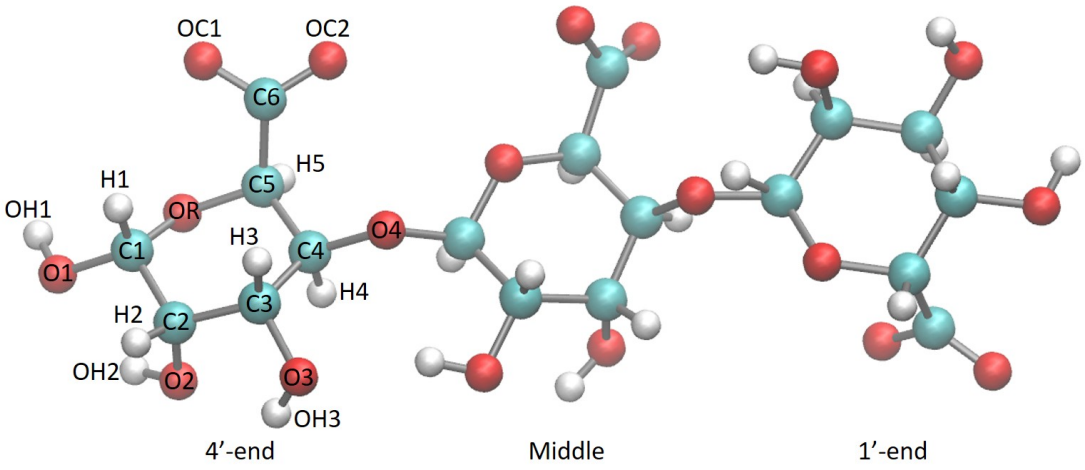


Figure 4.1: An example G trimer with atom numbering and monomer labelling.

also tracked, as were the dihedral torsion angles between the same C-O-C glycosidic link plus the C5 ring carbon connected to the carboxylate group. These three values were analysed to give details on the trimers' structures and flexibility.

Radial distribution functions (RDFs) were calculated with a bin size of 0.02 Å and a maximum distance of 12 Å. RDFs were calculated for alginate-alginate, alginate-ion, and alginate-water interactions. For the alginate-alginate RDFs, the OC1 and OC2 oxygens were used to determine the distribution of trimers around themselves. For the ion and water RDFs, oxygen atoms along the trimer were split into three groups and RDFs were calculated for each of these groups to the target species. The oxygen groups were labelled carboxylate (OC1, OC2), hydroxyl (O1 4'-end only, O2, O3, O4 1'-end only), and glycosidic and ring (OR, O4 (4'-end and middle only) as shown in Appendix G, Figure G.1.

Clustering of trimers was calculated based on C6-C6 atom distances. The distances between different trimers in each system were charted throughout the simulation. When two trimers remained closer than 10 Å from each other for at least 0.5 ns, they were considered a cluster, and were inspected in more detail using VMD.

The degree of clustering, $C(t)$, at time t, is defined as follows

$$C(t) = \frac{\sum_{i=1}^{10} iN_i(t)}{\sum_{i=1}^{10} N_{c,i}(t)} = \frac{\sum_{i=1}^{10} i^2 N_c(t)}{\sum_{i=1}^{10} N_{c,i}(t)} \tag{4.1}$$

where i is the cluster size in number of trimers, $N_{c,i}(t)$ is the number of clusters of size i at time t, and $N_i(t)$ is the total number of trimers involved in a cluster of size i at time t. This definition gives more weight to systems with fewer, bigger clusters, and has a minimum value of 1 for no clustering, and a maximum value of 100 for a system with all 10 trimers forming a single cluster. This definition allows a direct comparison of clustering in different systems.

The mean squared displacements (MSDs) in trimers, water molecules, and cations were calculated using multiple time origins. The simulations were split into two trajectories between 0-21 ns and 21-41 ns, and the MSDs were based on the last 20 ns of simulation time. The system centre of mass was subtracted from the MSDs during calculation. The MSDs are related to generalised diffusion as follows

$$\left\langle (r_i(t) - r_i(t_0))^2 \right\rangle \propto D\ t^{\alpha} \tag{4.2}$$

where the diffusion coefficient, D, and exponent, α, correspond to the intercept and slope of log-log MSD vs time plots according to the following.

$$\ln\langle \Delta r^2(t) \rangle = \ln K_\alpha + \alpha \ln t \tag{4.3}$$

The errors in the α and generalised diffusion coefficients were found using bootstrap resampling which is explained in Appendix F.

Calculation of end-to-end distances, C4-O4-C1 glycosidic angles, C5-C4-O4-C1 glycosidic dihedral torsion angles, RDFs, MSDs, and visualisation of the systems was performed using VMD software[206], either through in-built analysis tools or using TCL scripting.

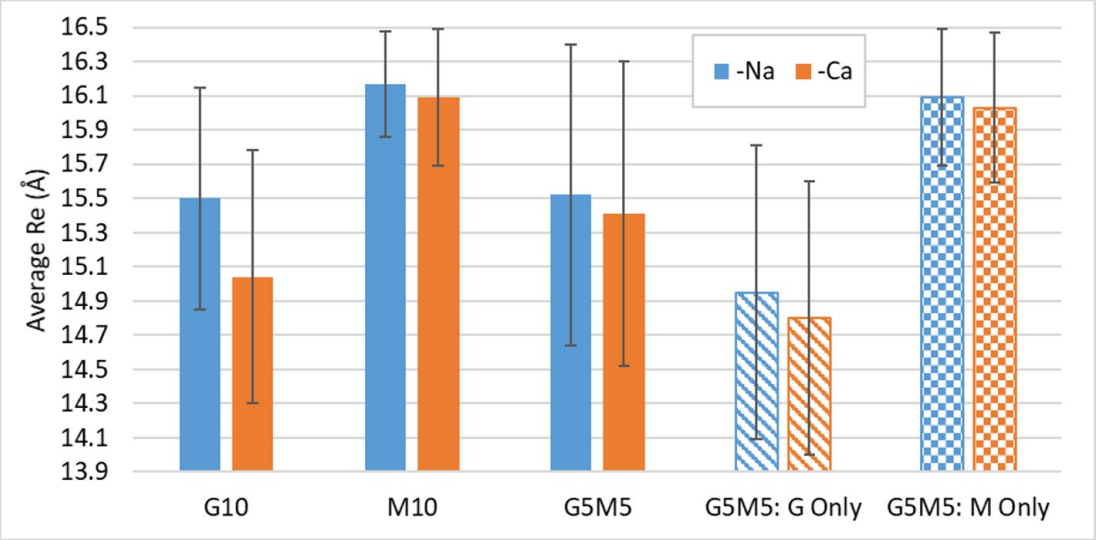


Figure 4.2: Average R_e and standard deviations (error bars) for all systems. The individual values for both the G (striped) and M (checkerboard) trimers in systems G5M5-Na and G5M5-Ca are included for comparison.

4.3 Results

To understand how the alginate and ion environment affects the systems' behaviours we present the trimer conformations, water and ion distributions around the alginate trimers, clustering of the trimers, and the mobility of the species within the systems.

4.3.1 Trimer conformations

The average R_e distances, and standard deviations for the Na^+ alginate systems are shown in Figure 4.2. M trimers have a larger R_e than G, in both the pure and mixed systems and in systems with Na^+ or Ca^{2+} ions, consistent with the well-established straight bonding of M and buckled bonding of G. The increase of R_e from G to M trimers was as much as 8 % (in system G5M5-Ca). The standard deviation, shown as error bars in Figure 4.2, indicates a much higher variation in the R_e values of the G trimers in both the pure and mixed systems. This differs from previous studies of longer (up to 30 monomer) alginate chains in which M chains presented with lower persistence lengths and therefore increased flexibility compared with G chains at 2-12 chains in the system [148], and suggests that conformational flexibility is not only a

function of monomer identity but also of chain length. Changing the system ion from Na^+ to Ca^{2+} universally results in a decrease in the average R_e value of up to 0.06-3 %. This effect is more prevalent in G trimers in both pure and mixed systems. It is likely that the more flexible G trimers have lower resistance to buckling which allows the negatively charged species along the trimers to come into closer contact with the Ca^{2+} ions.

The glycosidic C4-O4-C1 bonding angle distributions in the six systems showed minimal differences, with all systems having angles within a range of 119.3-121.4°; these distributions are included in Appendix G Figure G.2. The differences between systems can be seen more clearly by analysing the dihedral distributions. Figure 4.3 shows the 4'-end to middle and middle to 1'-end (labelled 4'-end and 1'-end, respectively) glycosidic dihedral angles for each trimer.

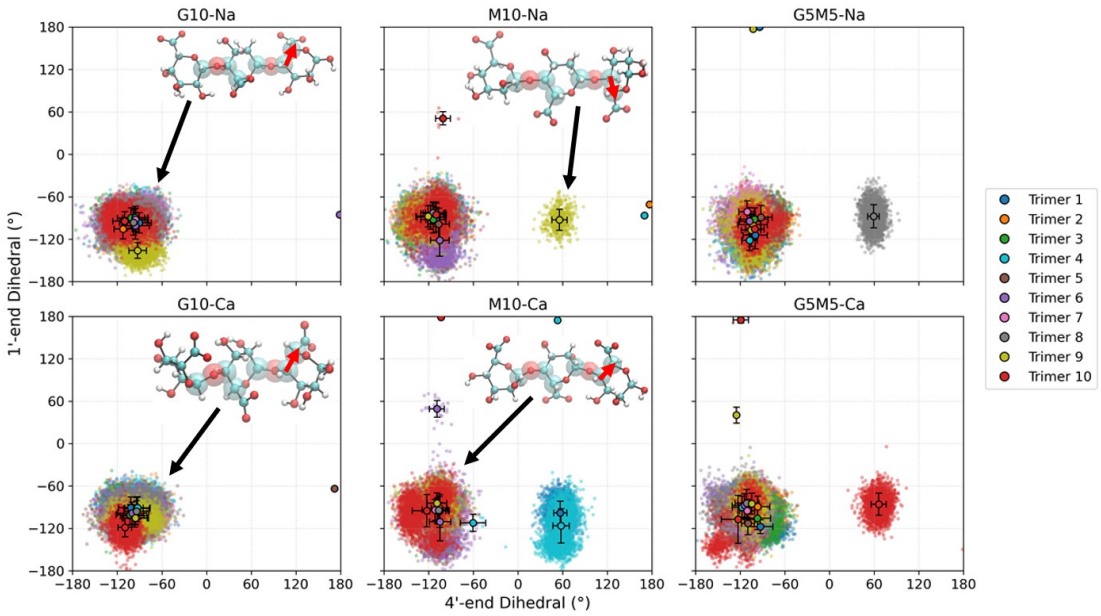


Figure 4.3: 1'-end to middle (1'-end) C1-O4-C4-C5 dihedral angles plotted against the same dihedral in the 4'-end to middle (4'-end) monomers for each trimer in all systems. In systems G5M5-Na and G5M5-Ca, trimers 1-5 are G and trimers 6-10 are M. Examples of the most common structures are given for systems G10-Na, M10-Na, G10-Ca, and M10-Ca.

G trimers were only found in a conformation which saw adjacent carboxylate groups point in opposite directions, as shown in the example trimer conformation in plot G10-

Na of Figure 4.3. Trimer 6 in system G10-Na and trimer 5 in system G10-Ca both accessed the bottom right quadrant of the plots once, corresponding to an up-down-down carboxylate direction configuration as shown in the inset of the dihedral plot of system G10-Na in Figure 4.3 - since this was rare for the G trimers it can be concluded that the up-down-up configuration of the carboxylate group direction corresponds to a low energy conformation. This configuration corresponded to dihedral angles in the range of -180 to 0° where negative and positive values denote whether the C5 atom points out of or into the page. This disagrees with the work by Perić *et al.* [150] who found that alginate chains with lengths of 8 or 9 monomers had conformational variability with time, as the trimers in the present study presented with primarily one conformation. This could of course be due to the short chain lengths used in these systems, and the presence of less bonded monomers to influence each other. This work does agree with another finding by Perić *et al.* that there is conformational variability along the chain shown in the carboxylate groups of each monomer pointing in opposite directions.

M trimers existed mainly in the same up-down-up configuration as the G trimers but also, less commonly, accessed an up-down-down configuration at around a 4'-end angle of 60°, the G trimers did not. This could suggest that the corresponding conformation (shown in plot M10-Na in Figure 4.3) of this region is lower in energy and therefore more accessible for the M trimers than G trimers. It is more likely, however, an effect of reduced sampling. The two M trimers in Na^+ systems which accessed this bottom right quadrant began the simulation in that conformation as shown in Appendix G Figure G.3. These M trimers did, however, also sample the bottom left quadrant, as shown in Appendix G Figure G.3, which shows that the more common up, down, up conformation does not correspond to such a low energy minimum that the trimers cannot leave this configuration once they are in it. The G trimers sampled the bottom right of the plots at minimally during the simulation which suggests that the bottom left region corresponds to a lower energy minimum for the G trimers than the M trimers and the bottom right region corresponds to a lower energy minimum for the M trimers than the G trimers. This may suggest that while the G chains have more flexibility

in how they bend around the glycosidic bonds, as shown in the R_e and glycosidic bond angle data, the M trimers could have more flexibility in the twist around the glycosidic bond, highlighted by the inclusion of the C5 atom. This possible flexibility presenting in M trimers could be responsible for the decreased persistence lengths and therefore increased flexibility seen in M chains compared with G chains in previous work. Longer simulations would be required to confirm this to increase the sampling of M trimer conformations. Using altered starting configurations and comparing the resulting trimer conformations would also make this finding more robust. Mixing the trimers in system G5M5-Na does not seem to have any effect on the glycosidic dihedral angles of the individual trimers.

There was little to no change between the Na^+ and Ca^{2+} systems in the trimer bonding structures. The example structures in the system G10-Na and G10-Ca plots of Figure 4.3 show the difference between trimers in similar glycosidic dihedral configurations but with different R_e values. As was the case in the Na^+ systems, the G trimers all fell into the bottom left quadrant of the plots while a minority of M trimers also accessed the bottom right quadrant. It can be seen from the plots presented in Appendix G Figure G.3 that the trimers which were, for the majority of the simulation, in the conformation corresponding to the bottom right quadrant of the plots all started in this conformation or in the top right quadrant, post system energy minimisation. Trimers 10 in system G5M5-Ca and 1 in M10-Ca never accessed the bottom left quadrant. This, as discussed before, is likely due to the length of the simulation, but could indicate an increased flexibility in twisting around the glycosidic bond in M trimers compared with G trimers.

It is possible in longer chains that the possible M flexibility to twist around the glycosidic bond becomes more prominent than the G flexibility to bend, especially when the polymers are aggregating with each other and become more packed, and this is what shows as M chain flexibility compared to G chain stiffness in the longer chains of previous work[148].

4.3.2 Water and ion distributions around the trimers

The RDFs for the oxygen atoms along the alginate trimers to the water hydrogen atoms in all systems are shown in Figure 4.4. An area of hydration is clear at a distance of 1.9 Å from the carboxylate oxygen atoms, followed by a second water peak at 3.2 Å. The hydration around the hydroxyl, and glycosidic/ring oxygen types is less pronounced than the carboxylate although occurs at similar distances. Hydration shells therefore exist around the alginate trimers at these distances.

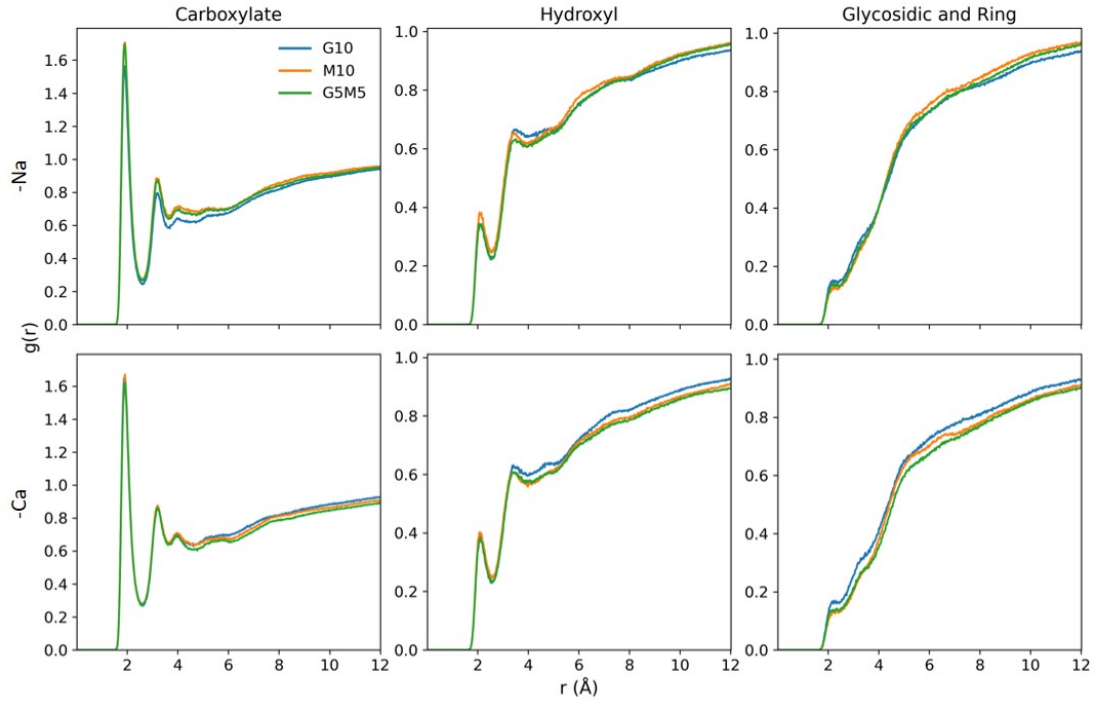


Figure 4.4: Oxygen group (carboxylate, hydroxyl, glycosidic and ring) to water molecule hydrogen atom RDFs for (top) Na^+, and (bottom) Ca^{2+} systems.

Minimal differences between the systems indicate that while water may mediate alginate-ion interactions as shown by the hydration shells present, it is not significantly impacted by the alginate monomer type or ion environment. This contradicts the work by Li *et al.* (2021)[152], who studied a single 30-monomer chain in 0.3-1 M Na^+ ion concentration or 0.15-1 M Ca^{2+} ion concentration, and saw water exclusion from proximity to alginate in Ca^{2+} ion systems compared with Na^+ ion systems. The

chains were linked at the periodic boundary to produce an infinite chain and were analysed using simulated annealing. The difference in water behaviour is likely due to the ion concentration being 3-23 times greater than in this work which means effects caused by Ca^{2+} ions are more prominent. The simulated annealing methodology used by Li *et al.* could also be moving the system to energy locations that are not accessible in MD.

The alginate-Na^+ and alginate-Ca^{2+} ion RDFs are shown in Figure 4.5. It is clear from the peak heights that, as with the water RDFs, the carboxylate groups are the primary points of interaction with Na^+ and Ca^{2+} for the trimers which is expected due to their accessibility and relatively high charge density. When compared with the water RDFs presented previously, it can be seen that the trimers have alternating water and ion shells at 1.9 Å (water), 2.6-2.7 Å (Na^+/Ca^{2+}), 3.2 Å (water), and 4.5 Å (Na^+/Ca^{2+}) from the oxygen atoms. The glycosidic/ring oxygen groups have a similar peak height for their second ion shell but the distance from the oxygen groups on both the glycosidic/ring and hydroxyl oxygen groups, and the width of these peaks is increased compared with the carboxylate groups second ion shell. This is likely due to the higher charge density and accessibility of the carboxylate groups and their higher proximity on average to the glycosidic/ring oxygens compared with the hydroxyl oxygens. This proximity allows the glycosidic/ring oxygens to "see" more of the ions which are coordinated by the carboxylate oxygens and corresponds to the second peak in the hydroxyl and glycosidic and ring group RDFs.

As shown in Figure 4.5, there is a more pronounced difference between the carboxylate-Na^+ ion RDFs between the systems with different trimer types (10G, 10M, 5G5M) than in the carboxylate-Ca^{2+} ion RDFs. The G trimers in system G10-Na saw more alginate-Na^+ interactions than the M trimers in system M10-Na for all oxygen groups. This shows that the Na^+ ions were more attracted to the G trimers than the M trimers which could be due to a slightly decreased net charge of the G trimers carboxylate groups (-2.48 per trimer) compared with the M trimers (-2.40) which would increase the electrostatic attraction between the G trimer carboxylate groups and the Na^+ ions.

The comparison of system G5M5-Na's oxygen group interactions compared with

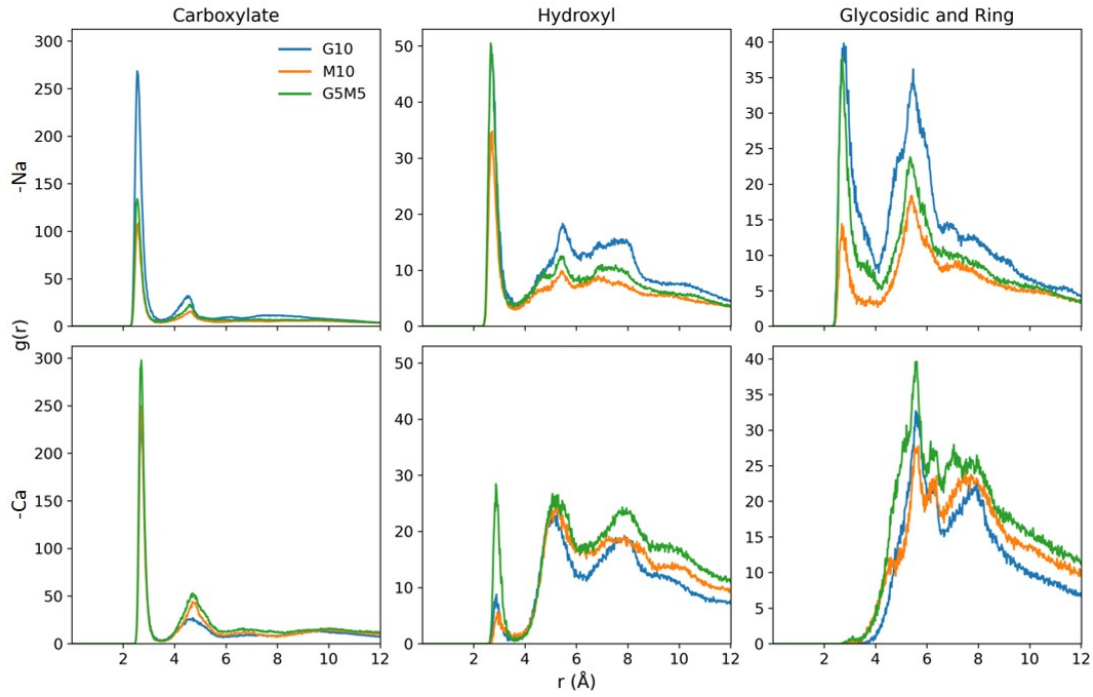


Figure 4.5: (Top) Na^+-alginate oxygen groups RDFs for systems G10-Na, M10-Na and G5M5-Na, and (bottom) Ca^{2+}-alginate oxygen groups RDFs for systems G10-Ca, M10-Ca and G5M5-Ca.

that of the pure systems shows that, in the carboxylate groups, the trimers seem to mix in a way which corresponds to a mixture of the pure systems, but in the other oxygen groups, the interaction is much closer to that of a pure G system. It is possible that the reduced R_e values seen for both G and M trimers in system M10-Na compared with the pure Na^+ systems, allows for more hydroxyl and glycosidic and ring oxygen to ion interactions.

The comparison between the M and G systems is contrary to what was seen in the study by Hecht and Srebnik (2017)[149] which concluded that 30-monomer M chains saw more interaction with Na^+ ions than G. However, these systems had both Na^+ and Ca^{2+} ions in the system which provided ion competition. Li *et al*[146] saw a lower first Na^+ ion shell peak for M than G 30-monomer chains with either Na^+ or Ca^{2+} ions in the systems, although the second peak was higher in the M system. This could suggest that alginate-Na^+ ion interaction is highly dependent on the length and number of alginate chains, and presence of other ions in the system; these trends could also be an

effect of the differing computational methods used as discussed previously.

The alginate-ion RDFs' peak positions were unchanged in the Ca^{2+} systems, showing that the alginate-water-ion shell behaviour is independent of the alginate and ion environment. The peak heights in the alginate-ion RDFs did change when moving from Na^{+} to Ca^{2+}. The carboxylate oxygens, the most accessible and charge dense along the trimers, saw their 2.7 Å peak height increase by 110 and 213 % for system changes of G10-Na to G10-Ca and M10-Na to M10-Ca, respectively. There was a corresponding reduction or loss in the first RDF peak for the hydroxyl and glycosidic and ring oxygen groups which shows an increase in ion specificity to the carboxylate ions. This agrees with the work by Li *et al.* (2021)[152] which showed more ion-alginate interaction specificity for carboxylate oxygens in Ca^{2+} systems.

For the Ca^{2+} systems, mixing the trimers in system G5M5-Ca resulted in an increase in alginate-ion interactions from the pure systems. This suggests that the trimers are aggregating more and providing a larger area of negative charge which will attract the cations. To assess the validity of this theory, the trimer clustering was investigated.

4.3.3 Trimer clustering

The progression of trimer clustering throughout the simulation and a snapshot of the clustering in the final time step of each system can be seen in Figure 4.6. As simulation time proceeds, the number and size of clusters increases in systems with Ca^{2+} ions, and it is likely that eventually a single cluster of 10 trimers would form in all Ca^{2+} ion systems. This strong tendency to cluster is likely due to the cross-linking effects of the divalent Ca^{2+} ions which can bind trimers together through electrostatic interactions. For the Na^{+}-ion systems, alginate trimers formed smaller clusters, compared to the Ca^{2+}-ion systems. This is consistent with previous simulations that found that Ca^{2+}-ion systems has a smaller solvent accessible surface area, indicating higher degree of aggregation compared to Na^{+}-ion systems [182].

Differences in clustering between G-rich, M-rich and mixed systems was less clear. For Ca^{2+}-ion systems, the degree of clustering was lowest for the G10-Ca, which is unexpected and contradicts the well known cross-linking "egg-box" structures formed

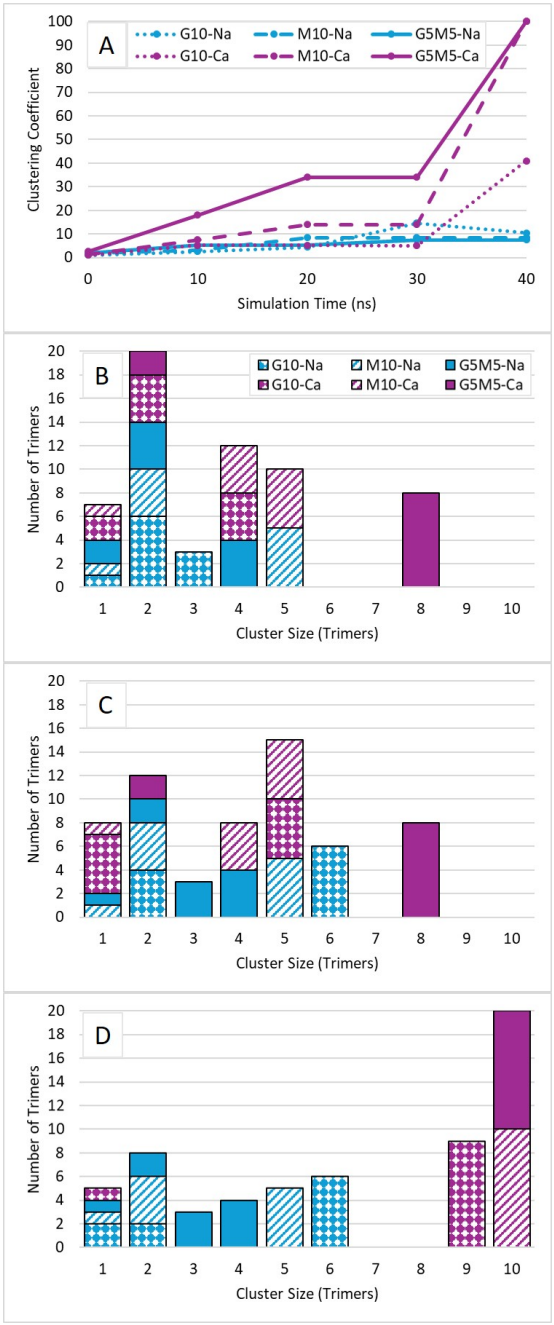


Figure 4.6: (A) Degree of clustering as described in Equation 4.1 versus simulation time. Number of trimers involved in 1-10 trimer clusters for each system (B) 20 ns into the simulation, (C) 30 ns into the simulation, (D) at the end of the simulation (40 ns).

by G chains. However, we note that trimers may be too short to form these structures, which is expected for the 8-monomer minimum chain length required to form a

stable "egg-box" structure[46]. We also note that the G5M5-Casystem aggregated the fastest, which agrees with previous simulations that found that M trimers promoted G-G interactions [148, 149].

4.3.4 Species Mobility

The MSDs for alginate trimers, ions and water are presented in Figure 4.7. In all systems and for all species, the exponent $\alpha \approx 1$, indicating that the systems reached timescales for normal diffusion.

The MSDs for water show almost no variation between the different systems which reflects the dilute nature of these systems and relatively weak interactions between the water and alginate trimers. The diffusion coefficients of water, included in Appendix G Figure G.4, values were in the range of 4.3-4.4 $\times 10^{-9}$ m^2s^{-1}. This is slightly less than the bulk TIP3P water diffusion of 5-6 $\times 10^{-9}$ m^2s^{-1}[208], as expected as the alginate and ions are likely to hinder water diffusion.

The alginate trimers are the largest and most highly charged species and have the slowest diffusion, as expected. The log-log plot for alginate in Figure 4.7 shows that the sodium systems have slightly higher MSDs, and correspondingly the diffusion coefficients presented in Appendix G Figure G.4 are slightly lower for the calcium systems, consistent with the increased aggregation in these systems.

The ions, while the smallest species in each system, are charged and involved in interactions with the alginate molecules, as discussed in the ion-alginate RDFs; their MSDs are therefore higher than that of alginate but lower than that of the water molecules. The Na^+ ions have higher MSDs in all systems than Ca^{2+} ions which is explained by the increased alginate-ion interactions in the Ca^{2+} systems as shown in the alginate-ion RDFs. The increased interactions between alginate and Ca^{2+} ions hinders the Ca^{2+} ion diffusion, resulting in lower MSDs.

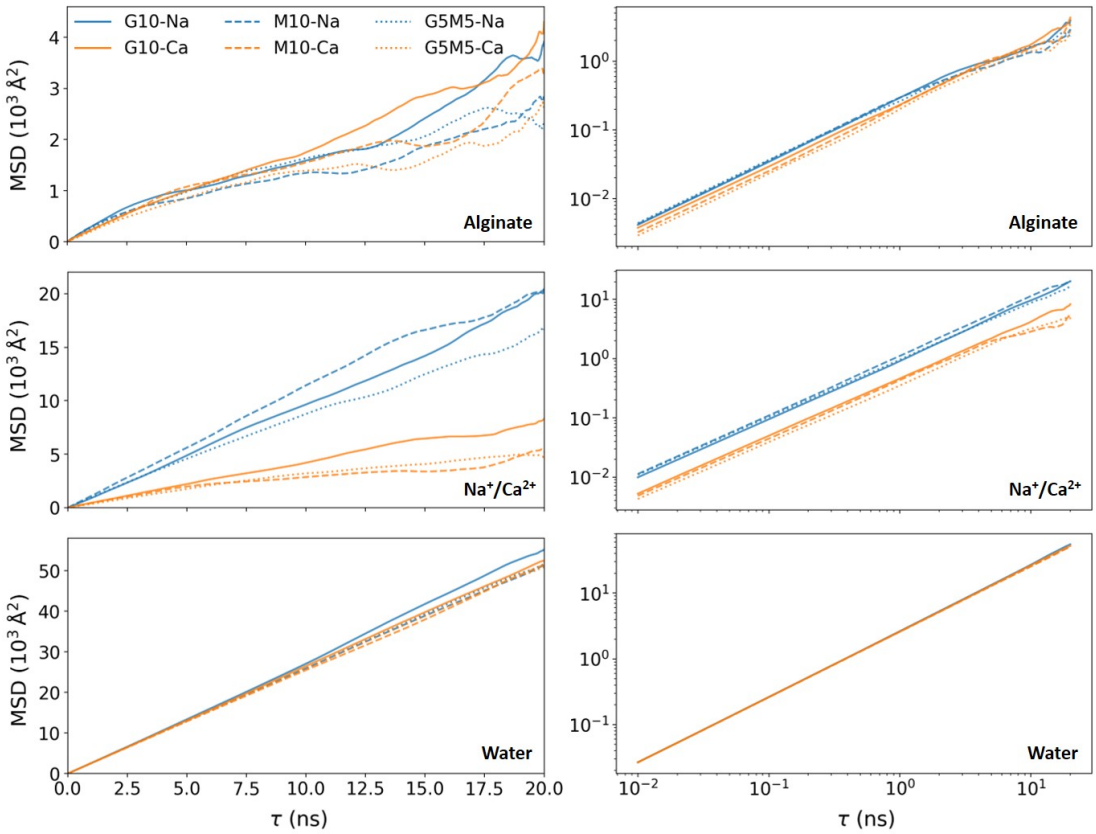


Figure 4.7: MSDs for all species in all systems shown as both normal (left) and log-log (right) plots. The key and axes labels are the same for all graphs.

4.4 Conclusions

Molecular dynamics simulations were used to investigate the structure and dynamics of alginate trimers in aqueous solution with a concentration of 0.016 g alginate/g water. The systems studied had 10 trimers, either all G, all M, or half G and half M, and either 30 Na^+ ions or 15 Ca^{2+} ions. The trimer conformations, ion-alginate and water-alginate interactions, alginate clustering, and species mobility for each system were analysed and compared to give insight into the differences between G, M, and mixed systems, and systems with Na^+ or Ca^{2+} ions.

In both Ca^{2+} and Na^+ systems, G trimers consistently had lower R_e values than M trimers, corresponding to less flexible bending of the glycosidic angle in M trimers. The Ca^{2+}-ion systems all had lower trimer R_e values compared to Na^+-ion systems, particularly for the more flexible G trimers, where the negatively charged monomers interact strongly with the Ca^{2+}-ions leading to bending of the trimers.

The water structure around the alginate trimers showed that water most strongly associated with the exposed trimer carboxylate oxygen atoms with well defined hydration shells at distance of 1.9 Å and 3.2 Å. A weak shell was observed around the hydroxyl groups and there was almost no association around the central glycosidic groups. Hydration shells showed no significant dependence on the ion environment or monomer type. The Na^+ and Ca^{2+} ions also interacted most strongly with the accessible and charge-dense carboxylate groups with ion shells at 2.6-2.7 Å. The Ca^{2+} ions generally had a stronger association than Na^+, particularly for the hydroxyl groups. However, the Na^+ ion association with the glycosidic group at 2.5 Å whereas Ca^{2+} ions showed almost no interaction with the glycosidic groups as the interaction is concentrated on the carboxylate groups. The Na^+ systems showed a greater variation with monomer type than the Ca^{2+} systems, with the Na^+-ions more strongly associated with groups on the G monomers, and least strongly with the M monomers.

In all systems the trimers were observed to aggregate, with aggregation happening more quickly in the Ca^{2+} systems, where the G5M5-Ca and M10-Ca systems formed a single cluster of all ten trimers within the simulation time. Surprisingly, the presence

of G monomers did not appear to correlate to a higher degree of clustering, although this behaviour may change in longer chain systems. Mean square displacements of alginate and ions showed that Na^+ systems exhibited slightly higher mobility than Ca^{2+} systems, consistent with the higher aggregation in Ca^{2+} systems.

The detailed gelling mechanisms of sodium alginate and calcium alginate are still largely unknown. This study of alginate trimers and ions in a dilute aqueous environment, found faster aggregation in M-containing Ca^{2+} systems. This shows that the role of M monomers in alginate gelation should be further explored for longer chain systems and M-G copolymers.

While this study allowed conclusions to be made about the behaviour of M and G chains in pure and mixed systems with Na^+ and Ca^{2+} ions, there were limitations. The use of trimers allows the analysis of alginate building blocks but is hard to link directly to long-chain and experimental alginate systems. The behaviour of short-chain polymers, as discussed in this study, can be very different from longer chains. This is due to the small size of the molecules which gives them different movement and interaction behaviour as well as the increased effects of the end groups on the chains. Usually the end groups of a polymer are a very small part of the polymer and have negligible effects on the polymer behaviour. In this study the end groups make up two-thirds of the system and therefore contribute a lot to the alginate behaviour. The simulations were also run for a relatively short time for polymer systems which are usually run for 100 ns or more to achieve full relaxation and equilibrium. The short timescales were used in this study due to limitations in simulation speed but do also match or exceed times used in previous work[150, 151]. The fastest simulation speed that could be reached with these systems was 4 ns per day. The decision was therefore made to move to GROMACS, an MD software which is more efficient for dilute systems. The methodology and tests done to move to GROMACS are discussed in Chapter 6.

To continue this investigation, experimental work was used to study similar factors as were presented in this chapter on alginate gel properties. This experimental work was used to get a broad multi-factor description of alginate systems that could be used to guide further, focused, simulation work.

Chapter 5

A design of experiments methodology to investigate the effects of curing conditions on calcium alginate strength

5.1 Introduction

While computational simulation can give us a very detailed look at an alginate system, it is harder to link to practical applications of alginate. Most available alginate research studies experimental systems. A range of factors - including G:M ratios[142, 209], sodium alginate concentration[109, 210], Ca^{2+} ion concentration [210, 211], pH[210], temperature [210, 212, 213], gelation time [212], and post-curing sodium chloride soaking [214] - have been shown to influence the mechanical properties of calcium alginate gels[209, 215, 216]. These effects have been studied using methods such as compression[109, 210, 214, 216], rupture[109, 213, 216], shear stress relaxation[214], and wet tensile strength [215]. Other studies have looked at, for example, alginate composite gels for increasing strength and entrapment properties [105], the differences between pH- and Ca^{2+}-induced gelling [39], and the impact of using various concentrations of ethanol in water as a sodium alginate solvent [217].

While these studies provide valuable insight into individual factors affecting gel strength, a systematic, multivariable analysis of their combined and interacting effects is lacking. This study contributes a structured, multivariable analysis for understanding and optimising calcium alginate gel strength, by systematically quantifying how

formulation factors influence gel mechanics, thus advancing beyond the single-variable approaches common in prior research. In this study a full factorial 2-level design of experiments (DOE) methodology was employed to examine how alginate G:M ratio, sodium alginate concentration, Ca^{2+} and Na^{+} ion concentrations, and pH influence the mechanical strength of calcium alginate gels. Excess sodium was included as a factor due to its role in alginate extraction and potential presence in commercial alginates[218–220], and pH was included for its relevance to food and biomedical applications where alginate is used as a gelling agent for a variety of solvents. A subset of gels was also used to assess the effect of dialysis membrane use during curing, in an attempt to slow Ca^{2+} ion diffusion into the sodium alginate solution, and produce consistently uniform gels. After gel formation, mechanical work during compression was measured, and statistical analysis was used to identify significant main effects and interactions.

5.2 Methodology

5.2.1 Materials

Preliminary experiments for preparing calcium alginate gels used Manugel GMB (FMC Corporation) and Manucol® LKX (International Flavors & Fragrances, Inc.) sodium alginate powders. The DOE experiments for calcium alginate gel strength testing used two sodium alginate powders supplied by CEAMSA, with G:M ratios of 0.79 and 1.08, and GG, GM, MG, and MM bonding ratios as shown in Table 5.1. The viscosities and molecular weights of these sodium alginate powders is unknown. Calcium chloride hexahydrate (98%), sodium chloride (99.5%), 8M sodium hydroxide, and 10% hydrochloric acid were purchased from APC Pure, Fisher Bioreagents, Honeywell, and Mistral, respectively. These reagents were used throughout the preliminary and DOE experiments.

5.2.2 Gel Preparation

Before the main DOE experiments were conducted, a series of small-scale (20-60 ml) preliminary tests were carried out to optimise the factor levels and methodologies used

Table 5.1: GG, GM, MG, and MM bonding pattern ratios for the 0.79 and 1.08 G:M ratio sodium alginate powders used in the DOE methodology provided by CEAMSA.

G:M Ratio	Ratios of Bonding Type			
	GG	GM	MG	MM
0.79	0.29	0.15	0.15	0.41
1.08	0.20	0.32	0.32	0.17

in the DOE experiments. These tests focused on the tactile and visual assessment of gels formed under varying conditions and did not include mechanical testing as this was unavailable for the preliminary testing. Various methods of introducing the calcium-containing curing solution to the sodium alginate solution were explored in an effort to consistently produce uniform gels without the addition of extra reagents. The most effective approach used visking/dialysis membrane tubing to slow Ca^{2+} diffusion into the sodium alginate solution, resulting in gels with minimal defects. However, using tubing for all required samples would have required excessive resources and bench space. The method was therefore adapted: flat sheets of dialysis membrane were laid on top of the alginate solution, and the curing solution was carefully poured over them. This adapted gel preparation method was broadly consistent with that used in the DOE experiments, as detailed below.

The pH of deionised water was adjusted to 5, 7, or 9 as required, using dropwise addition of either 0.05 M (for pH 5 or 9) or 0.005 M (for pH 7) NaOH or HCl. These pH-adjusted solutions were used to prepare low or high G:M ratio sodium alginate solutions at 1, 1.5, or 2 %w/w concentration using an overhead mixer. Solutions were mixed until visually fully dissolved (typically < 30 minutes), with mixer speed adjusted to maintain a vortex. Sodium alginate solutions were sealed and left at room temperature overnight to naturally de-aerate, then stored under refrigeration until use.

Curing solutions were prepared by dissolving the required amounts of NaCl and $CaCl_2$ hexahydrate into deionised water to yield 100 g of curing solution per gel at the required concentrations. Flat dialysis membranes (MWCO 3.5 kDa, Medicell Membranes Ltd) were cut to size, soaked for 30 minutes, rinsed, and re-soaked in deionised water until use. For each gel, 600 g of sodium alginate solution was poured into a

plastic container, a membrane was placed on top of the sodium alginate solution so that the whole surface of the sodium alginate was covered, and 100 g of curing solution was carefully poured onto the membrane. The containers were sealed and left to cure for one week. A metal cutter was then used to extract 4 samples (diameter of 35 mm and height of 15 mm) from each gel which were stored in 60 mL containers at room temperature until testing. All sodium alginate solutions were cured as soon after preparation as possible, and calcium alginate gels were tested within three days of post-curing sampling to minimise degradation and microbial growth.

During the first two experimental replicates, it was observed that the membranes occasionally sank into the sodium alginate solutions, causing the gel to form in two layers, as shown in Figure 5.1. In some gels, these layers had markedly different gel strengths. Gels affected by this issue were replicated a third time, alongside six additional gels selected for their low variance and representative gel strength values (low, medium, and high) based on replicate 1 and 2 results. These additional gels were included to assess whether membrane presence influenced gel strength, and membrane presence was therefore added as a sixth factor in the DOE.

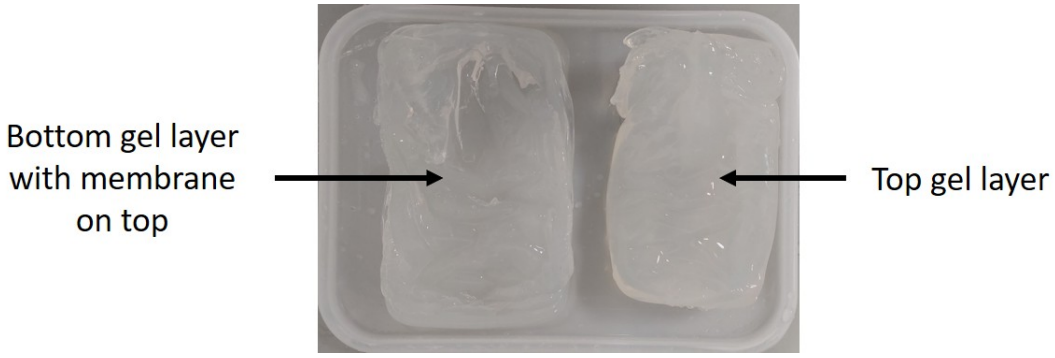


Figure 5.1: An example of the gel layer separation which occurred during curing when the membrane sank into the sodium alginate solution. The two layers formed were often tactilely very different.

5.2.3 Selection of factors and levels, and DOE methodology

The factors and levels selected through preliminary testing are shown in Table 5.2. Factors were classified as either qualitative (discrete, with defined low and high levels)

Table 5.2: Factors and levels used in DOE, decided through tactile and visual assessments in preliminary testing. Membrane presence was added after completion of the first two replicate runs as explained above.

Factor	**Factor Type**	**Low Level**	**Centre-Point**	**High Level**
Sodium alginate G:M Ratio	Qualitative	Low (0.79)	N/A	High (1.08)
Sodium alginate concentration (%w/w)	Quantitative	1	1.5	2
Ca^{2+}: alginate molar ratio	Quantitative	1:1	1.5:1	2:1
Na^{+}: alginate excess molar ratio	Quantitative	0:1	1.5:1	3:1
pH	Quantitative	5	7	9
Membrane presence in gel preparation	Qualitative	No	N/A	Yes

or quantitative (continuous, with defined low, centre-point, and high levels). The G:M ratio of sodium alginate was treated as a qualitative factor due to the presence of other unquantified characteristics - such as molecular weight and viscosity - that could also influence gelation, and because no centre-point material was available. The G:M ratio values for these sodium alginate powders are included in Table 5.2.

Codes were used to identify gel samples; each code is made up of a letter followed by 4 numbers, representing the value of each factor used in the creation of the gel. Gel L1105 was created using low sodium alginate G:M ratio, 1 %w/w sodium alginate concentration, 1:1 Ca^{2+}:alginate molar ratio, 0:1 Na^{+}:alginate excess molar ratio, and pH 5. Gel H2239 was created using high sodium alginate G:M ratio, 2 %w/w sodium alginate concentration, 2:1 Ca^{2+}:alginate molar ratio, 3:1 Na^{+}:alginate excess molar ratio, and pH 9. Codes LM and HM refer to centre-point formulations using low and high G:M ratio sodium alginate, respectively, with 1.5 %w/w sodium alginate concentration, 1.5:1 Ca^{2+}:alginate molar ratio, 1.5:1 Na^{+}:alginate excess molar ratio, and pH 7. Since

membrane presence was tested on a subset of gels and was not applied across the entire DOE, it was not included in the coding.

The quantitative factor ranges were determined based on preliminary experiments that produced gels with favourable properties. Specifically, selected ranges yielded gels that (i) exhibited minimal deformities such as air bubbles or doming, (ii) retained structural integrity when cut without breaking or collapsing, and (iii) showed distinguishable differences in tactile and visual assessments, ensuring that measured variations in gel strength would be meaningful. Additionally, the lower bound of the pH range was set above the pK_a of alginate in order to minimize the formation of protonated carboxylic acid groups, which can interfere with calcium crosslinking and hinder gel formation.

Minitab Statistical Software 22[195] was used to create a full factorial 2^5 DOE and to randomise the run order of the experiments. There were 32 base runs from the 2^5 design (encompassing all gel codes except LM and HM), plus a centre-point for each level of the sodium alginate G:M ratio (gel codes LM and HM), resulting in 34 runs for each replicate. Two replicas were performed and four samples were extracted from each gel replica, resulting in 272 samples being tested in the texture analyser. After identifying the membrane sinking problem, 17 of the original gels were prepared a third time, without membranes, and four additional samples were extracted from each, resulting in a further 68 samples being tested. In total, 340 samples were tested. Each combination of factor levels was represented by either 8 or 12 samples, depending on whether the formulation was replicated a third time without membrane use.

5.2.4 Texture analysis of calcium alginate gels

Calcium alginate gel strengths were measured using a CT3 Brookfield Texture Analyser at the James Hutton Institute, Dundee. The probe used was flat and circular with a diameter of 34 mm. The texture analyser was set to "Compression" with a target distance of 4 mm. One cycle (probe compresses, and returns to its starting point once) was performed on each gel sample with 10 load data outputs every second. The test and return speeds were set to 0.5 mm s^{-1}. Calcium alginate gels were tested in a randomised run order created using Minitab, although samples from the same gel were

tested consecutively.

It was originally planned that gels would be tested in their 60 ml storage containers but the gels' resistance to deformation around the probe and subsequent consistent exceeding of the equipment's 25 kg load maximum, causing the analyser to stop, meant that the gels were instead taken out of their storage containers and tested on a flat concrete surface as shown in Figure 5.2. In this state, with no container, the gels were free to deform in any direction outwards and around the probe.

Figure 5.2: An example of the pre-compression sample set-up in the texture analyser. The gel was placed on a clean, dry concrete plate and the probe was manoeuvred electronically to sit flat on top of the gel sample before the compression began.

A typical compression load vs time graph is shown in Figure 5.3. The compression began with a sharp increase in the load experienced by the probe followed by a slower load increase, potentially indicating gel relaxation, followed by another sharper increase as the extent of gel compression increased. After the target distance of 4 mm (at 8 s) was reached, the probe started to return to its initial position, which corresponded to a sharp drop in load experienced by the probe and gel decompression. The sharp decrease in load showed that the gels were not very elastic as they did not push against the probe as it ascended. Many samples had an area of negative load near the end of the decompression phase which is explained by the gels not returning to their full height at the same speed as the probe.

In some datasets, there was an anomaly where the last data point of the compression stage showed a decrease in load even though probe descent distance had increased from the last step (Figure 5.4). Since none of the gels visibly significantly deformed or broke at this stage of the analysis, outside of the expected compression, and it was only ever

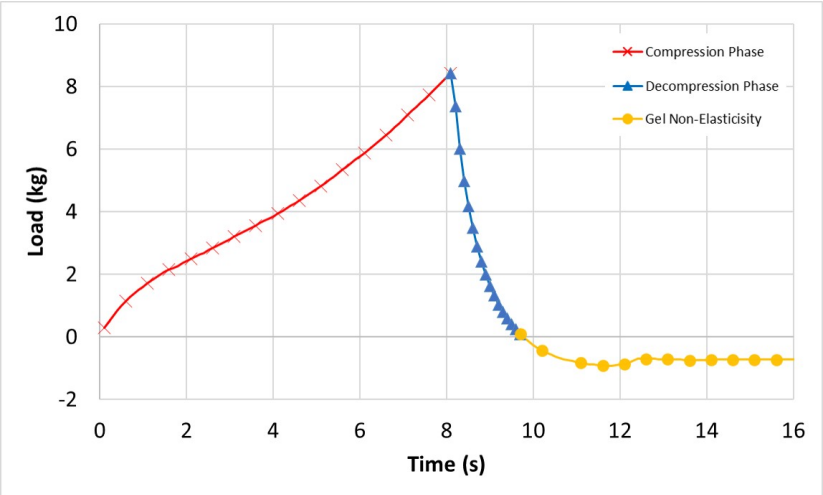


Figure 5.3: A typical example of the raw data obtained from the texture analyser where the load experienced by the probe is plotted against the time passed since the start of compression. The graph has three sections: compression of the gel in the first 8 seconds of analysis (red crosses), decompression as the probe retracts (blue diamonds), and below-zero load due to the gel not returning to its full height (yellow triangles).

the very last data point in the compression phase, it was assumed that this was due to an effect of the equipment and these points were excluded from the analysis.

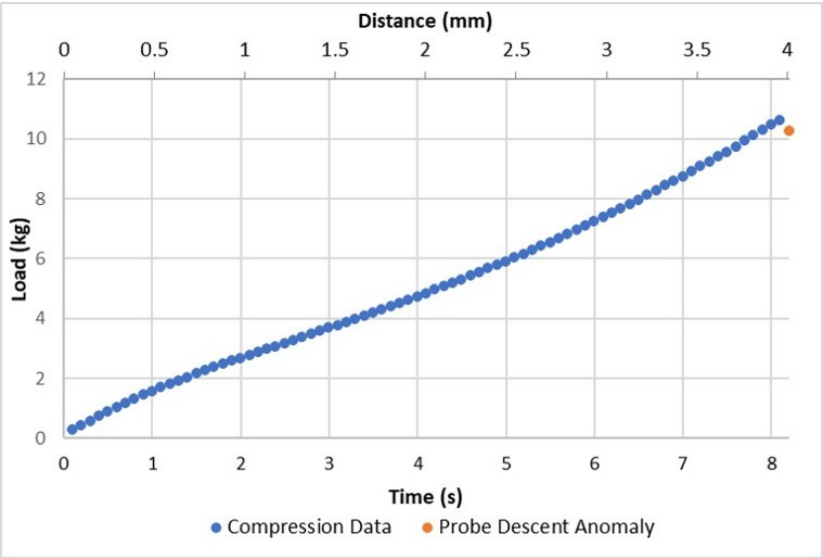


Figure 5.4: An example of the anomaly which was sometimes seen in the data sets produced by the texture analyser where the last data point in the compression data saw a decrease in load while the probe descent distance was still increasing.

From these load vs time graphs, the maximum load and the area under the curve were extracted. This area is proportional to the work done during compression since the load vs time plot is analogous to a load vs distance plot; area under the curve of the load vs time graphs is therefore labelled work (W) and has units kg s. The results for maximum load and work were very similar so only the work results are discussed here.

In this study we first present a comparison of the low and high levels for each factor, followed by a statistical analysis of the significance of these results.

5.2.5 Statistical analysis

Minitab software was used to perform a factorial regression on the experimental data with the aim of determining which factors (as listed in Table 5.2) and factor interactions had the most impact on the response data. A general linear model was fitted, with all statistical testing being conducted at a 95 % confidence level, using least squares regression to estimate the coefficients of effect on response of each factor and factor interaction. The full methodology for this analysis is presented in Section 2.3.

5.3 Results

We will first compare the effect of each individual factor by presenting the average values of all gels created at low and high levels of each value, respectively. These comparisons look at the work experienced by the gel samples only to allow an assessment of the gels' strength and what the strength values could indicate about the gels' micro-properties with reference to the simulated systems presented in: (i) Chapter 4, and (ii) literature. We will then discuss the statistical analysis of this data performed using DOE methodology to determine the significance of the effect of each factor and factor compound.

5.3.1 Effects of individual factors on the calcium alginate gel strength

In this section, each factor is analysed independently to assess the effects on gel strength data of moving from the factor's low to high level. We first present an overview of the average work values seen at the low and high levels of each factor followed by a more in-depth look at the data for each factor. Centre-point data was included in the DOE for use in statistical analysis and will be discussed further with those results. In the comparison of low and high level for the qualitative sodium alginate G:M ratio factor, centre-point data points were included to enable a comparison across all levels. Centre-point data was excluded from the comparison of low and high levels of quantitative factors since the compounded effects of all quantitative factors being changed to their centre-point cannot be compared in this way. Only the gels which were replicated a third time to test the effects of membrane presence are included in the membrane presence low and high level factor comparison.

Average work at low and high levels for all factors

Figure 5.5 shows the average work for each factor's low and high levels with standard deviation included. It is clear from the data in this graph that sodium alginate G:M ratio, sodium alginate concentration, Ca^{2+}:alginate molar ratio, and Na^{+}:alginate excess molar ratio had the most effect on work, with sodium alginate concentration being especially relevant. These factors saw a much greater change in work than pH or membrane presence between their low and high levels.

Membrane presence in the gel preparation method

Figure 5.6 shows the effect of presence and absence of dialysis membrane during preparation of calcium alginate gels. The graph contains two sets of data: (left) the replicated gels which showed no signs of layering, gave relatively low variability data sets in the first two replications, and were included as validity tests, and (right) the gels which were replicated a third time due to layering around the sunken membrane and producing high variability data sets in the first two replications. There is no clear pattern which

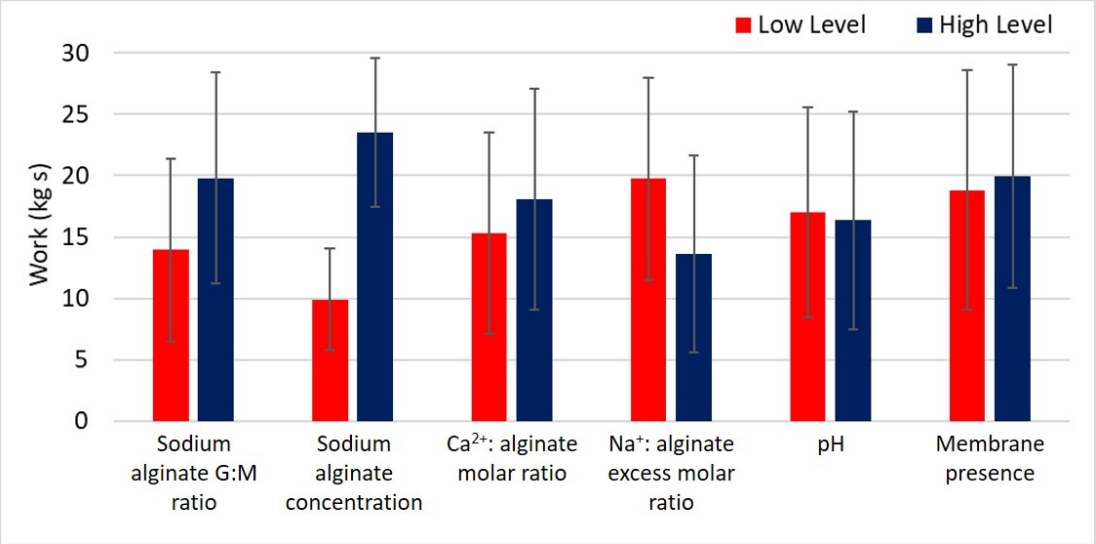


Figure 5.5: A comparison of the average work at the low and high levels of each factor with standard deviation included. The low level of "membrane presence" is absence and the high level is presence.

would suggest membrane presence/absence has a consistent effect on the resulting gel strength; as can be seen in Figure 5.5, the average work performed on the gels changed by 5.9 % from 18.8 ± 9.8 kg s to 19.9 ± 9.1 kg s.

The presence of membranes was not expected to induce any effect on the work of the gel samples, it was added into the methodology in an attempt to make uniform gels by slowing Ca^{2+} diffusion, rather than to affect the strength of the final gels. The washing and rinsing process performed on the membranes prior to their use was imposed to reduce or negate any influence of impurities within the membranes, adding to the expected lack of effect.

Since there is no discernible effect from including dialysis membranes in the gel creation process, the third replication of the experiment, in which no membranes were used, was included in the original two replicas' dataset, in which membranes were used throughout, and the data was processed as one set. The following sections compare data from all three replicas, with and without membranes.

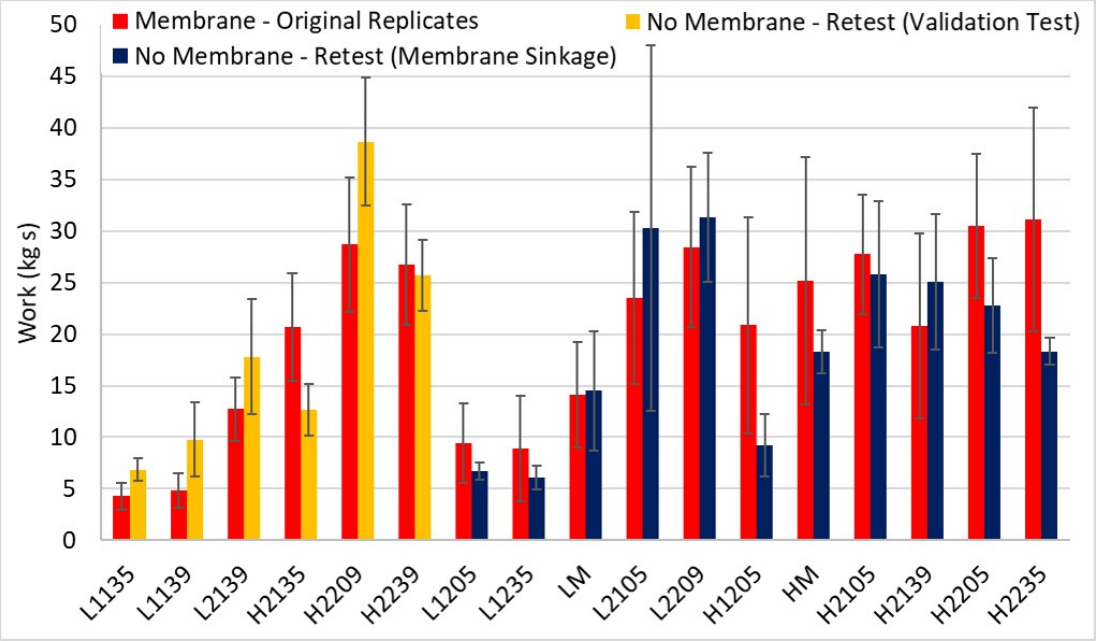


Figure 5.6: A comparison of presence and absence of membrane in the gel creation process on work produced by gel sample compression. A subset of gels (left) were retested as a validation test and the rest (right) were retested due to membrane sinkage in replicas 1 and 2.

Sodium alginate G:M Ratio

Moving from low G:M ratio sodium alginate to high G:M ratio saw an increase of 42.0 % from an average of 13.9 ± 7.4 kg s to 19.8 ± 8.6 kg s, respectively. This is shown clearly in Figure 5.5 and in more detail in Figure 5.7. As can be seen in Figure 5.7, there were no obvious trends in this data that any other gel conditions contributed to this effect.

Low G:M ratio sodium alginate is known to produce weaker gels due to fewer "egg-box" forming GG bonding[209]. The details in Table 5.1 show that the low G:M ratio sodium alginate powder had a higher proportion of GG bonding than the high G:M ratio although this could refer to short GG chains which are not long enough to form stable "egg-box" structures as discussed by Bowman *et al.*[46]. More information would be needed about the molecular weight of these sodium alginate powders and the length of the bonding sections shown in Table 5.1 to form a conclusion. This could also link to the findings of the trimer study presented in Chapter 4 which saw increased

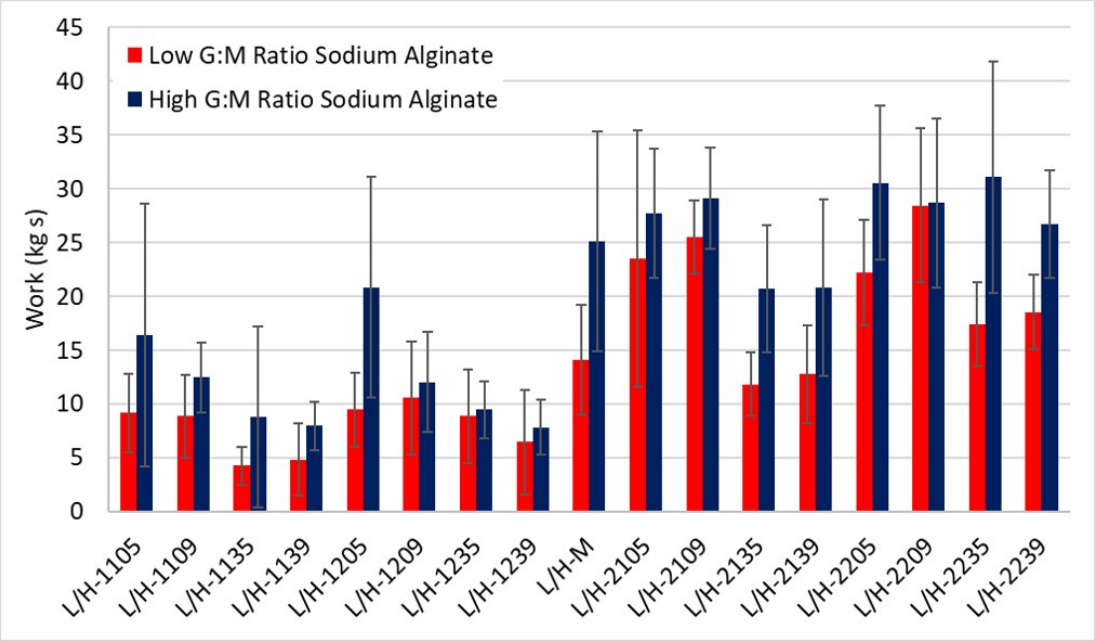


Figure 5.7: A comparison of low and high G:M ratio sodium alginate on work produced by gel sample compression.

alginate-alginate interactions between G and M trimers compared with G-G and M-M, suggesting the monomer types encourage more crosslinking in each other. These mixed interactions could be contributing more than expected when relying on the "egg-box" model of cation binding. It can be said from these results, however, that increased G content - whether in long or short blocks within the alginate chains, and crosslinking with other G or M blocks - results in an increased calcium alginate gel strength.

It is also possible that the low G:M ratio sodium alginate powder had a lower molecular weight (shorter alginate chains) than the high G:M ratio sodium alginate powder, although this is unknown. A lower molecular weight would mean that the calcium alginate gel was relying more on alginate-alginate and alginate-Ca^{2+} intermolecular interactions rather than stronger intramolecular alginate bonding to create the gel matrix. This would also be expected to contribute to a weaker gel.

Sodium alginate concentration

Increasing the sodium alginate concentration significantly and universally increased the gel strength as seen in Figures 5.5 and 5.8. Moving from 1 %w/w to 2 %w/w sodium alginate concentration increased work by 137.1 % from 9.9 ± 4.1 kg s to 23.5 ± 6.1 kg s. This effect was expected as it is reported multiple times in literature [39, 109, 209, 210, 213, 214]. Increasing the sodium alginate concentration increases the number of alginate chains that are present in the network. When Ca^{2+} ions are present, as in all gels presented in this study, increasing the number of alginate chains in the system facilitates more intermolecular chain-Ca^{2+}-chain interactions which create more crosslinks in the alginate network and produce a stronger gel. No trends are present in this data which would indicate that another factor is affecting the scale of the gel strength increase from 1 to 2 %w/w sodium alginate concentration.

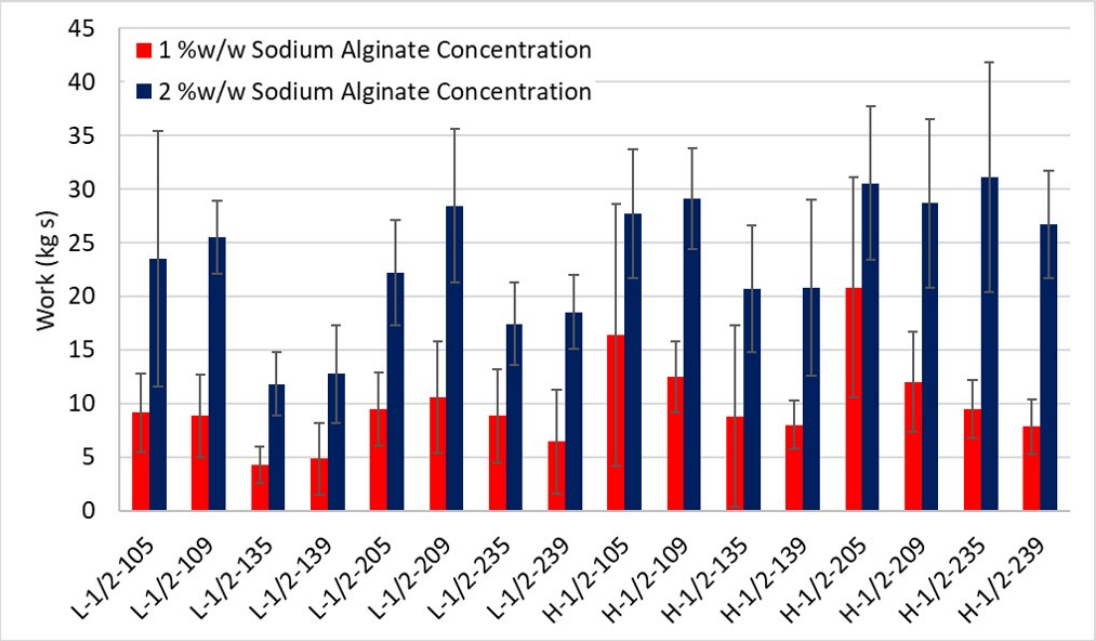


Figure 5.8: A comparison of 1 and 2 %w/w sodium alginate concentration on work produced by gel sample compression.

Ca^{2+}:alginate monomer molar ratio

The effect of Ca^{2+}:alginate monomer molar ratio on gel strength is shown in Figures 5.5 and 5.9. Figure 5.5 shows that moving from 1:1 to 2:1 Ca^{2+}:alginate molar ratio increased the average work by 18.1 % from 15.3 ± 8.2 kg s to 18.1 ± 9.0 kg s. However it is clear from Figure 5.9 that this increase in work was not universal; gels L2105, H1109, H1139, and H2109 had a slightly higher work that their 2:1 Ca^{2+}:alginate molar ratio counterparts.

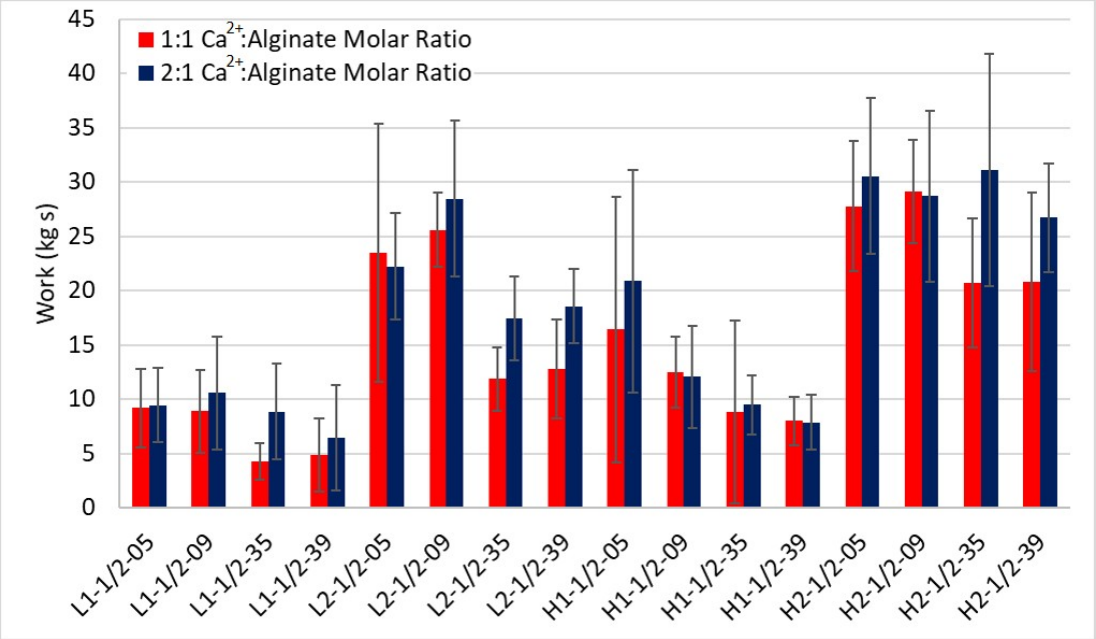


Figure 5.9: A comparison of 1:1 and 2:1 Ca^{2+}:alginate molar ratio on work produced by gel sample compression.

Due to its crosslinking effects, it is expected that an increase in Ca^{2+} ion concentration within the calcium alginate system would produce a substantial increase in the gel strength. Previous work showed that increasing the Ca^{2+}:alginate monomer molar ratio from 0:1 to 2.5:1 increased the calcium alginate gel strength logarithmically[142, 211, 221–223]. However, it has been noted that a 0.5:1 Ca^{2+}:alginate monomer molar ratio in the "egg-box" model corresponds to a fully crosslinked system[141, 142, 144]. The small increase in gel strength with increasing Ca^{2+}concentration in the current study is consistent with the high Ca^{2+}:alginate monomer molar ratios in the range from 1:1

to 2:1, where the number of crosslinks is already beyond saturation. This accounts for the relatively small increase in gel strength in this factor (18.1 %) when compared with the same value for sodium alginate concentration (137.1 %), even though both factors are essential in creating crosslinks within the system.

Na^+: alginate monomer excess molar ratio

Figure 5.10 shows the effect of Na^+:alginate monomer excess molar ratio on individual gel work. This ratio does not include the 1:1 Na^+:alginate ratio included in sodium alginate, it only reflects the Na^+ which was added to the gel through the curing solution. Increasing Na^+ concentration in the curing solution resulted in a pronounced decrease in gel strength in most gels with only one (H22-0/3-5) seeing a small increase in gel strength when Na^+ molar excess increased. The gel strengths changed by -30.8 % from 19.7 ± 8.3 kg s to 13.7 ± 8.0 kg s as shown in Figure 5.5.

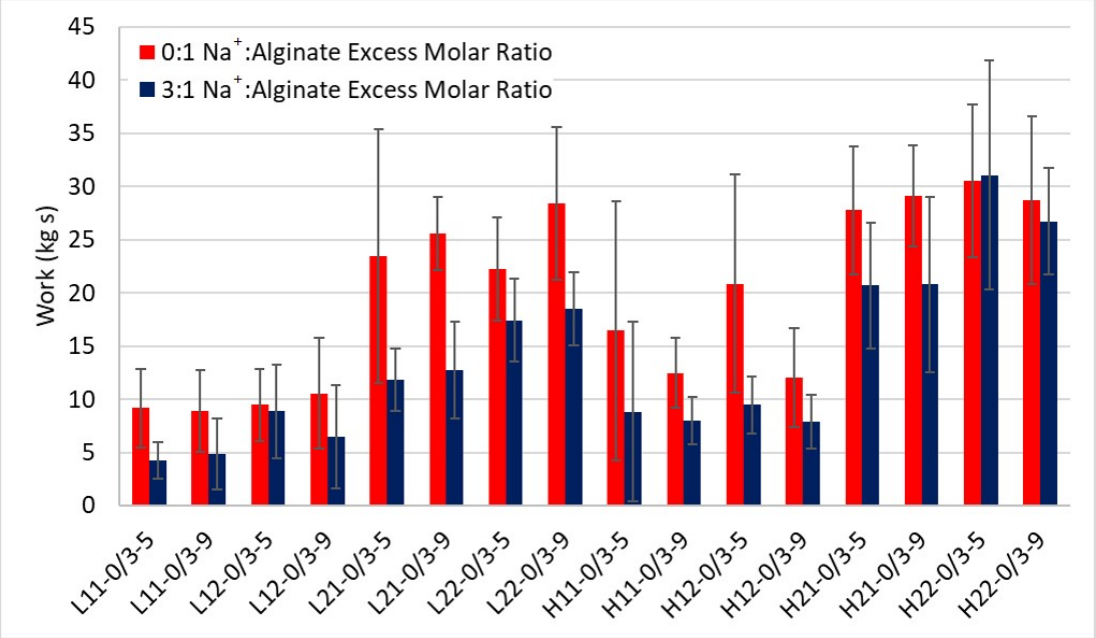


Figure 5.10: A comparison of 0:1 and 3:1 Na^+:alginate excess molar ratio on work produced by gel sample compression.

Gels which had at least two of (i) high G:M ratio sodium alginate, (ii) high sodium alginate concentration (2 %w/w), or (iii) high Ca^{2+}:alginate molar ratio (2:1 Ca^{2+}:alginate)

had increased gel strength, as shown in the previous factor analyses, and were less likely than gels at other factor levels to experience a significant gel strength decrease when Na^+:alginate excess molar ratio was increased from 0:1 to 3:1.

The work done by LeRoux *et al.* [214] shows a similar trend, in which calcium alginate gels soaked in NaCl solutions resulted in a decrease in calcium alginate gel strength. It is likely that the Na^+ ions in the alginate matrix compete with Ca^{2+} ions, thus reducing the formation of crosslinks, hindering stronger alginate-cation interactions, and creating a weaker gel, as was also the conclusion of LeRoux *et al.* in their work.

pH of the alginate solution solvent

The pH of the solution used to dissolve the sodium alginate powders had no obvious effect on resulting gel strength as seen in Figures 5.5 and 5.11. The average work changed by -3.92 % from 17.0 ± 8.8 kg s to 16.4 ± 8.9 kg s when moving from pH 5 to pH 9. Figure 5.11 shows that this change was relatively random across the samples and it is therefore concluded that pH had an insignificant effect on work of the samples in the pH range studied.

This is believed to be due to the buffering effects of sodium alginate. While all solutions started at the planned pHs of 5, 7 (not included in this comparison due to centre-point incompatibility with this comparison as explained previously), or 9, once the sodium alginate was dissolved and the solution had de-aerated, the pH was recorded again and was always within a range of 7-7.5 with no obvious trends. This suggests that the alginate is acting as a buffer in the solution and the extent of deprotonation along the alginate chains, centred on the carboxylic groups, is changing to prevent the pH of the system moving from the 7-7.5 range. This suggests that as long as the pH is above the pK_a of about 3.4-4.4[40, 41], it should not effect the gel created.

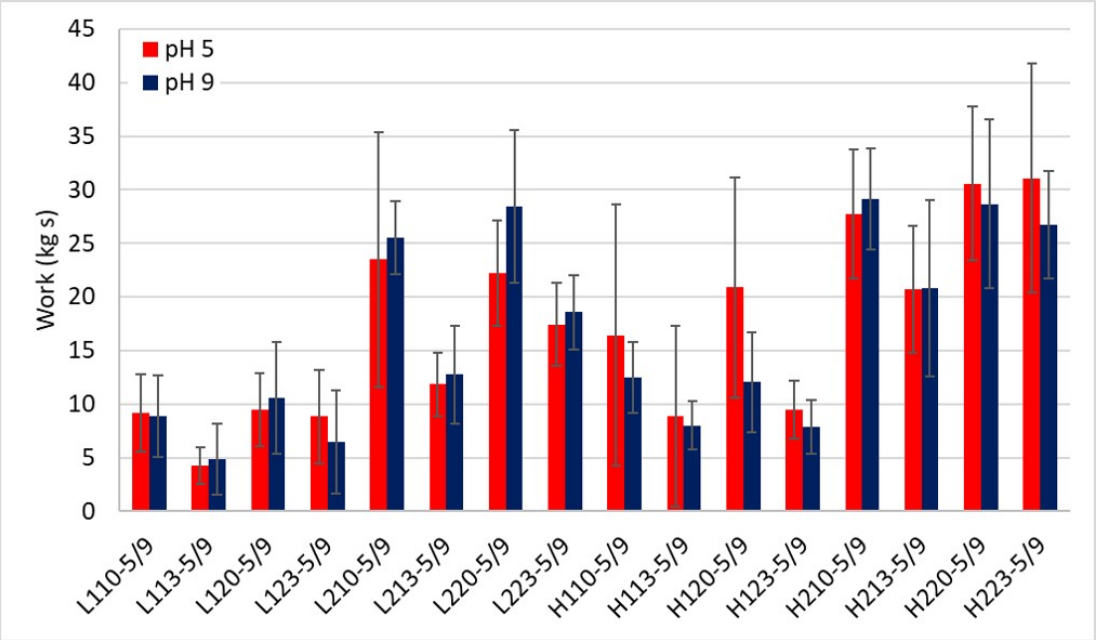


Figure 5.11: A comparison of the sodium alginate solutions solvents' pH of 5 and 9 on work produced by gel sample compression.

5.3.2 Statistical analysis: factorial regression

Stepwise factorial regression resulted in a general linear model (GLM) fitted to the data presented in the previous section, with only statistically significant factors and factor interactions included. The original data presented with non-ideal residuals and so a Box-Cox transformation was performed using Minitab's optimal transformation coefficient (λ) calculation. The methodology for this transformation is discussed in Section 2.3.3. The analyses presented in this section are of this transformed data.

The significant effects on the work proxy response variable were, in descending order of significance, the sodium alginate concentration, Na^+: alginate excess molar ratio, sodium alginate G:M ratio, Ca^{2+}:alginate molar ratio, and Na^+:alginate excess molar ratio times Ca^{2+}:alginate molar ratio; the most effective way to alter gel strength is by changing the sodium alginate concentration. Factors were primarily independent with only a weak interaction present between the ions in the curing solution. The extent of the effect each factor or factor interaction had on the work proxy response variable can be seen in the Pareto chart presented in Figure 5.12.

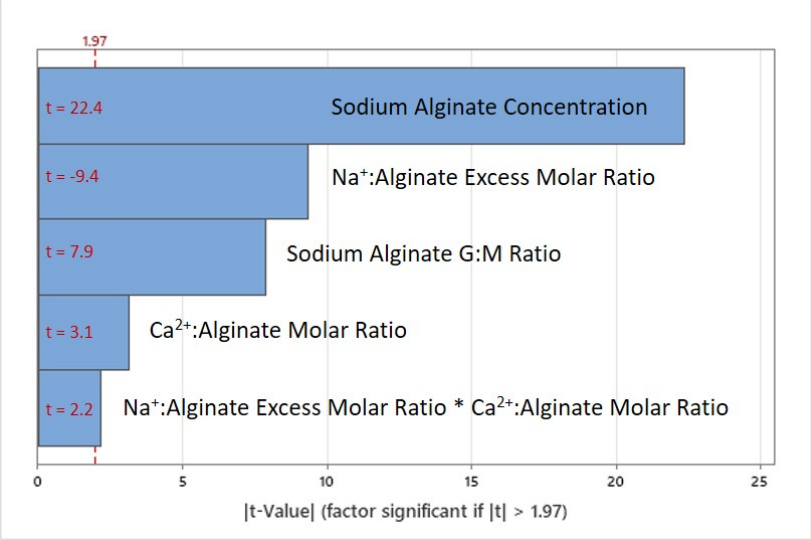


Figure 5.12: Pareto chart of effects of experimental factors on calcium alginate gel compression work, where a factor was considered significant at a |t-value| of 1.97 or greater.

The t-values are a statistical value based on the size of the effect the factor had on

the response variable and allow direct comparison between each factor or compounded factors; a t-value of 1.97 or above (as seen in Figure 5.12) indicates a factor which had a statistically significant effect on the gel strength at a 95 % confidence level.

It can be seen from the negative t-value included that when Na^+:alginate excess molar ratio was increased the gel strength decreased, all other significant factors had a proportional effect on the gel strength. The presence of the compounded effect of Ca^{2+}:alginate molar ratio times Na^+:alginate excess molar ratio shows that there may be an ion competition effect within the alginate network which plays a part in the resulting gel properties, furthering this hypothesis which was presented in the previous section. Further study is required to fully confirm this effect and understand the nature of the competition which is taking place, although it is logical that the two cations would compete for areas of negative charge along the alginate chains. Comparing this ion competition factor to the factors presented in the previous section, and acknowledging its small t-value, shows that its effect on the gel strength of calcium alginate gel was small. Although statistically significant, this compounded factor likely has limited practical impact.

A significant centre-point factor was included in the GLM, suggesting that there is curvature within the model which cannot be modelled by the linear model. A Response Surface Model (RSM) design would allow a fuller description of this curvature.

The model had predictive R^2 values of about 70 % which indicates that this model would not predict previously untested values accurately. This is due to the large variability seen in all gels, though specifically those which split around the membrane. This does not, however, take away from the important conclusions which can be drawn from the individual factor and statistical analysis performed on this data and the foundation this work provides for future analysis.

5.4 Conclusions

Calcium alginate gels were made using different sodium alginate and curing solution conditions and were tested with texture analysis. Samples were compressed over a set

period of time, and a proxy for work done on the gel was extracted. This work provides quantitative analysis of how calcium alginate gels can be adapted through only their curing conditions to further the practical understanding of calcium alginate gelling. A full 2^5 factorial design of experiments was used to systematically evaluate the effects of sodium alginate G:M ratio, sodium alginate concentration, Ca^{2+}:alginate molar ratio and Na^+:alginate excess molar ratio introduced through a sodium alginate curing salt solution, and pH of the sodium alginate solution aqueous solvent on the resulting calcium alginate gel strength. An additional factor, membrane use in slowing the curing solutions' diffusion into the sodium alginate solution during the gel creation process, was introduced mid-study in response to an observed gel splitting issue and was appropriately included in the design. Centre-points were included for qualitative factors enabling the identification of model curvature. A general linear model was fitted to the data, and its robustness was analysed.

Sodium alginate concentration had the strongest effect on gel strength - increasing the sodium alginate concentration significantly increased work done on the gel through increasing the amount of available alginate for crosslinking within the system. Na^+:alginate excess molar ratio had a strong negative effect on gel strength with increases in Na^+ concentration in the curing solution decreasing gel strength due its ability to disrupt alginate-Ca^{2+}-alginate interactions and therefore weaken crosslinking. Sodium alginate G:M ratio and Ca^{2+}:alginate molar ratio also had significant, although weaker, effects on the gel strength by providing more crosslinking opportunities. The effect of Ca^{2+}:alginate molar ratio was limited due to the high range (1:1 and 2:2) used; once all available alginate cation coordination sites were saturated, excess Ca^{2+} had a much reduced effect. pH and membrane presence did not produce a significant effect in gel strength: sodium alginate acted as a buffer, so all solutions, pH 5, 7, or 9, were altered to pH 7-7.5 before curing took place; membrane presence introduced little to no impurities due to the soaking process used to prepare it and simply slowed the introduction of the Ca^{2+} ions in the sodium alginate solution.

The inclusion of a significant centre-point effect in the fitted GLM and a significant lack-of-fit test indicated non-linearity that the GLM could not fully capture. Despite

this, a second-order model was not fitted due to dataset size and design constraints; a response surface design would be needed for this. The final model had a predictive R^2 value of about 70 % indicating low to moderate predictive ability. However, the model was still able to identify clear, significant factor effects and provided a reasonable approximation of system behaviour. Future studies should use a more complex DOE and increased replication to better capture curvature and improve model adequacy.

This work has provided an overview of the most important factors which affect the resulting strength of calcium alginate gels and can be used as a starting point for more focussed studies, such as a Response Surface Model (RSM) DOE which would allow full modelling of the curvature within the model, or a second full factorial DOE following the same methodology as this but removing pH and membrane presence, and increasing the replicas to allow for datasets closer to 20-50 for each gel to be included.

Calcium alginate gelling properties are often treated qualitatively in formulation work, with reliance on well-used methods rather than understanding of the underlying mechanisms. This work lays groundwork for optimisation of calcium alginate gels to be achieved through understanding of the molecular interactions involved, allowing customisation of the calcium alginate gels via their formulation. The findings of this study have the potential to impact the food and materials industries by enabling the customisation of food texture and material strength.

A practical outcome of this work resulted from the identification of Na^+ ion concentration as factor which can significantly affect calcium alginate gel strength. The researchers at OCEANIUM® had been attempting to extract a high gel strength alginate from their seaweed feedstocks and this work proposed that Na^+ left in the alginate after the extraction process could be affecting this property. Work on this by OCEANIUM® is ongoing. It was therefore decided to investigate the effect of excess Na^+ ion concentration further in a more focused, computational, investigation. This work is presented in Chapter 7 and is preceded by an explanation and discussion of transferring the computational model from the LAMMPS software package to GROMACS in Chapter 6.

Chapter 6

Transferring from LAMMPS to GROMACS for more efficient simulations

As discussed in Chapter 4, the alginate trimer study highlighted limitations in the efficiency of LAMMPS as a simulation software for systems of this kind. With the ARCHIE-WeSt supercomputer and 40 CPU tasks (the maximum available), the fastest simulation speed that could be achieved was 4 ns/day. It was therefore decided to move to another simulator and GROMACS was chosen due to its high efficiency, especially for systems which are dominated by non-bonded interactions, such as the dilute systems presented in Chapter 4[189, 224, 225]. This chapter discusses the benefits of using GROMACS specifically for dilute polymer systems.

In this chapter, we first present methodologies proposed to increase the efficiency of dilute alginate simulations in LAMMPS and discuss why these methodologies would not give the scale of efficiency increases that was achieved in transferring to GROMACS. We then detail how the force field described in Chapters 2-4 was adapted for GROMACS and the major differences in the chosen algorithms used in LAMMPS and GROMACS. Finally, a comparison of a few select parameters between like systems run in both LAMMPS and GROMACS is made to conclude whether the results from LAMMPS and GROMACS simulations of alginate systems can be compared directly without extensive qualification regarding differences in the underlying codes.

Following this work, it was believed that the LAMMPS and GROMACS simulation results were similar enough to continue with GROMACS simulations, and to compare

the results of systems simulated in the two software packages. It was later found, following the completion of this Chapter and Chapter 7, that the differing ranges found in bonded parameter accessibility between LAMMPS and GROMACS was indicative of different force constants being used in the two simulators. Analysis of these parameters highlighted two small mistakes in the translation of AMBER parameters into GROMACS units which could have produced significant differences in behaviour in the systems: (i) a comma where there should have been a decimal place created a force constant of 585,760 kJ mol^{-1} rad^{-1} instead of the intended 585.76 kJ mol^{-1} rad^{-1} in the angle between atom types Cg-Cg-Os, and (ii) a misplaced factor of 2 included in the improper force constant doubled the correct C-O2-O2-Cg improper force constant from 43.932 to 87.864 kJ mol^{-1}. Future work should use the corrected values, presented in Appendix I, to replicate the simulations presented in both this Chapter - to ensure correctness in the force field - and then Chapter 7.

6.1 Proposed methodologies for increasing efficiency of dilute alginate simulations in LAMMPS

We first present the task timing breakdown output from LAMMPS for the systems presented in Chapter 4 to understand where the main efficiency bottlenecks are in these systems. We then propose different methods that could potentially increase the efficiency of dilute alginate systems simulated in LAMMPS.

6.1.1 Task timing breakdown output for dilute alginate systems simulated in LAMMPS

Within the output from a LAMMPS simulation is a breakdown of the task timing, as shown in Figure 6.1. The example in Figure 6.1 is taken directly from the output of system G10-Na presented in Chapter 4, and is representative of all systems presented in Chapter 4. It is clear that there are two main efficiency bottlenecks in the system: "Pair" (short-range non-bonded interactions) and "Kspace" (long-range electrostatics).

```
MPI task timing breakdown:
Section |  min time  |  avg time  |  max time  |%varavg| %total
---------------------------------------------------------------
Pair    | 2.885e+05  | 2.9896e+05 | 3.1034e+05 |1025.4 | 69.55
Bond    | 81.517     | 195.4      | 544.73     | 633.4 |  0.05
Kspace  | 51757      | 62810      | 72815      |2214.7 | 14.61
Neigh   | 26706      | 26744      | 26773      |  11.4 |  6.22
Comm    | 7542.4     | 11121      | 15612      |2950.8 |  2.59
Output  | 155.4      | 155.42     | 155.82     |   0.5 |  0.04
Modify  | 21610      | 27561      | 32588      |2495.9 |  6.41
Other   |            | 2276       |            |       |  0.53
```

Figure 6.1: Task timing breakdown from output of system G10-Na presented in Chapter 4.

Traditionally, MD was performed on Central Processing Units (CPUs) that show strengths in stability and running in serial but are slower when there are a large number of calculations such as force calculations at every time step. In recent years, simulation software packages are being updated to run certain processes on Graphics Processing Units (GPUs) which are optimised for parallel tasks and are very efficient at repetitive tasks, such as force calculations. CPUs are still used for tasks like communication between parallel cores while GPUs perform the majority of the calculations required; this combined use of CPUs and GPUs can produce speed increases of 100-800 % when used correctly[153, 189]. To reduce the time that LAMMPS spends on non-bonded interactions, the largest bottleneck presented in Figure 6.1, the main step to take would be to run the simulation with GPU usage. At the time of writing, LAMMPS was not fully compiled to work with GPUs on the ARCHIE-WeSt supercomputer, so "Pair" could not be significantly reduced.

It may be possible to reduce, or at least balance, both the short- and long-range non-bonded interaction calculation times by altering the Coloumb cut-off distance. By testing values between 0.9-1.2 nm, which are common values for this parameter, there may have been some increase in overall simulation efficiency[163, 226]. To further optimise the time spent in the long-range electrostatic calculation times, the accuracy of the Particle-Particle Particle-Mesh (PPPM) algorithm could have been adjusted to test for time saving. This method maps the long-range electrostatics onto a grid and solves them in Fourier space. Again, this would be a balance, this time between the quality of the output and the speed of the calculations. Neither of these optimisations

would produce the large simulation efficiencies expected of GPU integration.

6.1.2 Water bond and angle constraints

Most of the non-bonded interaction calculations, which are presenting as a bottleneck in the LAMMPS simulations presented in Chapter 4, result from the large number (18,851) of water molecules. During these simulations, the LAMMPS command *fix shake* was applied to the C-O bonds and C-O-C angle in each water molecule. This constraint command allows the movement of each water atom as expected from the bonded and non-bonded forces, and then attempts to return the C-O bond lengths and C-O-C bond angle back to their equilibrium values. This command was used with an accuracy tolerance of 0.01 % and a maximum number of iterations of 20, both recommended values for this command[153]. A large amount of computational energy and time, therefore, was spent iteratively returning these values to their equilibrium values. An alternative to the *fix shake* command is *fix rigid*[153]: instead of returning each constrained parameter to its equilibrium value in each time step, *fix rigid* treats each constrained molecule as a fixed structure. Fully rigid water molecules can massively increase simulation speeds as they remove the requirement of bond and angle vibration and energy calculations. This algorithm still requires force and torque calculations to be performed on each molecule but the individual atom movements, and subsequent attempts to return these atoms to their equilibrium values, as needed in *fix shake*, are removed.

6.1.3 Simulation box shape

Since the large number of water molecules and their corresponding calculations were deemed a major factor in the relatively low efficiency of LAMMPS simulations, it would be ideal to reduce this number. To maintain the properties of the system while reducing water molecule numbers, the minimum periodic image distance must be maintained[156]. By choosing a non-cubic simulation box such as a rhombic dodecahedron or truncated octahedron, the minimum periodic image distance observed in a cubic system can be maintained while the "extra" water molecules in the corners of the

cube are removed[153, 156, 189]. These non-cubic box shapes can remove up to 30 % of the solvent molecules in the system in this way[189, 227].

6.1.4 Integrator and time step

As discussed in Section 2.2, there are many different algorithms available for implementation in solving the equations of motion, including multiple time step methods. A multiple time step integrator allows different time steps to be applied for the calculation of different interaction forces[153]. The "fastest" interactions, including bonded interactions, are calculated at the smallest time step. Intermediate interactions, calculated at an slightly larger time step than the "fast" interactions, include short-range non-bonded interactions. "Slow" forces include long-range electrostatics[153]. This reduces the number of calculations performed on interactions which are unlikely to change in the time step provided, increasing overall simulation efficiency. A multiple time step integrator can be implemented using the LAMMPS command *run_style respa* which applies the reversible reference system propagator algorithms (rRESPA) proposed by Tuckerman *et al.*[228].

6.1.5 Outcomes of LAMMPS optimisation

Overall, the main alteration to the LAMMPS simulations which would increase simulation efficiency would be implementation of GPUs, especially for the non-bonded interaction calculations. These calculations were the biggest bottleneck in the simulations presented in Chapter 4 and, aside from some minor optimisations in the accuracy or changes to the system, this bottleneck could only be significantly reduced or removed by implementation of GPUs. Since GPUs were not available with LAMMPS for this project, the decision was made to move to a simulator software package which was fully compiled with GPUs on the ARCHIE-WeSt supercomputer.

GROMACS is a popular MD software package that was designed for biological macro-molecules in aqueous and membrane environments and is currently optimised for use with GPUs on ARCHIE-WeSt[154]. There are many differences between the implementations of LAMMPS and GROMACS. The following sections discuss how the

established force field and algorithms used in Chapter 4 were adapted to better suit GROMACS.

6.2 Adapting system force field from AMBER/LAMMPS to GROMACS

One of the LAMMPS software package's greatest strengths is its customisability, which allows all information needed for a simulation to be stored in one input file. GROMACS instead requires the information for the force field to be stored in a force field directory from which it pulls information as required. GROMACS has force fields built in, including datasets from AMBER, CHARMM[173], GROMOS[229], and OPLS[176], but this does not include the AMBER GLYCAM force field used in this work. The first step was therefore to compile a force field directory that could be applied to alginate systems with the TIP3P water model and the ion parameters used in Chapter 4.

6.2.1 Force field potentials and parameters

There are some differences between the units or descriptions of bonded and non-bonded parameters between AMBER and GROMACS which are solved by simple conversion: (i) AMBER uses kcal and Å while GROMACS uses kJ and nm, (ii) AMBER includes a factor of 0.5 in it's dihedral potential barrier term which GROMACS does not, and (iii) AMBER LJ atom sizes are given as R_{min} and GROMACS are given as σ where $R_{min} = \sigma \times 2^{1/6}$. Another difference is that LAMMPS does not have an option to treat improper dihedrals as described in the AMBER force field which uses the potential equation shown in Equation 6.1. Instead, the *improper_style cvff* LAMMPS command was used which uses the equation presented in Equation 6.2. Since the phase shift (ϕ_s) is 180° for the C-C-O-O improper included in alginate (this is the only improper included in the structure) as discussed in Chapter 2, a value of $d = -1$ is used in the LAMMPS parameters to match the AMBER parameters.

$$E = K\left[1 + \cos(n\phi - \phi_s)\right] \tag{6.1}$$

$$E = K\left[1 + d\cos(n\phi)\right] \tag{6.2}$$

A force field directory was compiled and contains files which give descriptions of the types, masses, charges, and bonds of all atoms, and bonded and non-bonded parameters used in the system making it applicable to many alginate systems. Files are included which describe an alginate molecule, a Na^+, Ca^{2+}, or Cl^- ion, and a TIP3P water molecule. These all had the same parameters as used in Chapter 4 and presented in Chapter 2.2.2. Use of the key *-cs spc216.gro* in the command *gmx solvate* calls the file "*spc216.gro*" which is included in GROMACS, and is also required for the addition of the TIP3P water model into these systems.

6.2.2 Differences in algorithms used in LAMMPS and GROMACS

In this section we will discuss the biggest algorithm choice differences between LAMMPS and GROMACS, specific for the systems presented in this work. All of the algorithms discussed here are discussed more generally in Section 2.2.

Electrostatics

For calculation of electrostatic (Coulomb) interactions, the particle-particle particle-mesh (PPPM) methodology is used in LAMMPS and fast smooth particle-mesh Ewald (SPME) is used in GROMACS. These methodologies are related as they both map the long-range part of the electrostatics onto a grid and solve them in Fourier space. The main differences are that SPME is standardised and optimised for biomolecular systems and GPU use within GROMACS whereas PPPM in LAMMPS is more general, tunable, applicable to a wider range of systems, and not yet fully GPU accelerated[153, 189, 224].

Minimisation

The default minimisation algorithm used in LAMMPS is the Polak-Ribiere conjugate gradient which is recommended as the most effective choice for most systems[153]. Conjugate gradient methodology is often applied when an accurate and well-relaxed configuration is required[153]. It is more common to use steepest descent in GRO-

MACS as it is very robust and stable even when the initial configurations are far from realistic (overlapping structures)[189]. Biomolecular systems do not often require a very accurate configuration, as required in, for example, a crystal simulation, and so a more complicated algorithm such as the conjugate gradient algorithm, which is better at finding the true local minimum, is not required.

Constraints

As discussed above, the *fix shake* command was used in the LAMMPS simulations presented in Chapter 4 to constrain the water molecule atom movements. As with the long-range electrostatic interactions and minimisation algorithms, GROMACS has a more specialised algorithm called *settle*, which is specifically for rigid 3-site water models such as TIP3P[189]. This works in a similar way to the LAMMPS command *fix rigid*, discussed above, by moving each water molecule as an unchanging structure instead of allowing individual atom movements[189].

Thermostats and barostats

The last major difference between the input of the simulations presented in Chapter 4 and the input used for GROMACS simulations is the thermo- and barostat algorithms applied. In the LAMMPS systems, the Nose-Hoover thermo- and barostat were implemented, in the GROMACS systems the V-rescale thermostat and Parrinello-Rahman barostat were applied. The Nose-Hoover algorithm is safe and general but can produce large fluctuations and can require careful equilibration. Although the stochastic V-rescale method introduces random fluctuations in the dynamics of the system, it produces smaller temperature fluctuations than the Nose-Hoover, and therefore works better with smaller systems. The Parrinello-Rahman algorithm is more physically realistic and rigorous than simpler pressure coupling methods. GROMACS forums and documentation generally recommend the V-rescale and Parrinello-Rahman combination for post-minimisation/equilibration runs as they are stable and produce low temperature and pressure fluctuations[189].

Once these choices had been translated as closely as possible from LAMMPS and

then optimised for use in GROMACS, alginate systems were run in both simulators and the output was compared. This allowed the new input and methods used with GROMACS to be checked, and allowed any major differences between the LAMMPS and GROMACS outputs to be identified.

6.3 Comparison of alginate simulations in LAMMPS and GROMACS

To test if the alginate system had been successfully translated from LAMMPS to GROMACS, different properties in various systems were tested and compared between the simulator software packages.

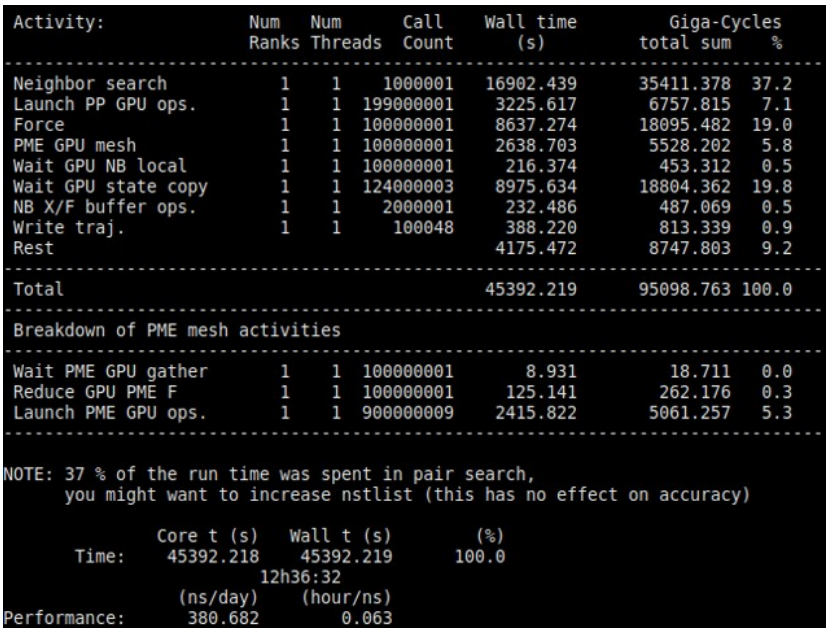
1. Select bonded energies from an alginate molecule were compared to check that the AMBER parameters had been correctly converted into GROMACS form and units.

2. The end-to-end distances of a trimer were compared to make sure that the intramolecular bonds and intermolecular interactions of one trimer were presenting in a similar way in both simulators.

3. Alginate to Na^+ ion RDFs were compared between the simulators to ensure that the non-bonded parameters were correct and that alginate-ion interactions were presenting as expected.

4. Trimer clustering was tested to ensure alginate intermolecular interactions were presenting in a similar way in both LAMMPS and GROMACS.

For comparisons (i)-(iii) above, a single G trimer with three Na^+ ions and no water was simulated for 1 ns. This system was chosen for its simplicity while maintaining a system net charge of zero. For the comparison described in (iv) above, system G10-Na, as described in Chapter 4, which had 10 G trimers, 30 Na^+ ions, and 18,851 water molecules, and was run for 20 ns, was recreated in GROMACS. Both system types were energy minimised following the methods described in Chapters 4 and 7 before being



Activity:	Num Ranks	Num Threads	Call Count	Wall time (s)	Giga-Cycles total sum	%
Neighbor search	1	1	1000001	16902.439	35411.378	37.2
Launch PP GPU ops.	1	1	199000001	3225.617	6757.815	7.1
Force	1	1	100000001	8637.274	18095.482	19.0
PME GPU mesh	1	1	100000001	2638.703	5528.202	5.8
Wait GPU NB local	1	1	100000001	216.374	453.312	0.5
Wait GPU state copy	1	1	124000003	8975.634	18804.362	19.8
NB X/F buffer ops.	1	1	2000001	232.486	487.069	0.5
Write traj.	1	1	100048	388.220	813.339	0.9
Rest				4175.472	8747.803	9.2
Total				45392.219	95098.763	100.0
Breakdown of PME mesh activities						
Wait PME GPU gather	1	1	100000001	8.931	18.711	0.0
Reduce GPU PME F	1	1	100000001	125.141	262.176	0.3
Launch PME GPU ops.	1	1	900000009	2415.822	5061.257	5.3

NOTE: 37 % of the run time was spent in pair search,
you might want to increase nstlist (this has no effect on accuracy)

	Core t (s)	Wall t (s)	(%)
Time:	45392.218	45392.219	100.0
		12h36:32	
	(ns/day)	(hour/ns)	
Performance:	380.682	0.063	

Figure 6.2: Task timing breakdown from output of a single-chain 24-monomer G system with a 1:1 Na^+ ion: alginate monomer molar ratio, presented and discussed in Chapter 7.

run in NPT. The main reason for the change in simulator software was to increase the efficiency of the simulations. Simulation speed for system G10-Na presented in Chapter 4 increased from 4.020 ns/day to 285.050 ns/day. With further optimisation this speed was increased to 380 ns/day for the systems presented in Chapter 7, as shown in Figure 6.2. It can be seen in the GROMACS output shown in Figure 6.2 that further optimisation would be possible via the *nstlist* value.

The theoretical bonded potential and the energy of the bond in question at each value it sampled when run in LAMMPS and GROMACS should match exactly. The angle versus energy plot for a C-C-O angle in Figure 6.3 shows an example of the comparisons done between the potential and outputs from LAMMPS and GROMACS. The example bond, dihedral, and improper energy comparisons performed also matched between simulators and are included in Appendix H. There were no differences in the potentials but there were differences in the ranges of bond lengths, angles, dihedral/improper torsion angles that each system sampled. This is attributed to the short simulation timescale, not allowing each system to fully explore the energy landscape.

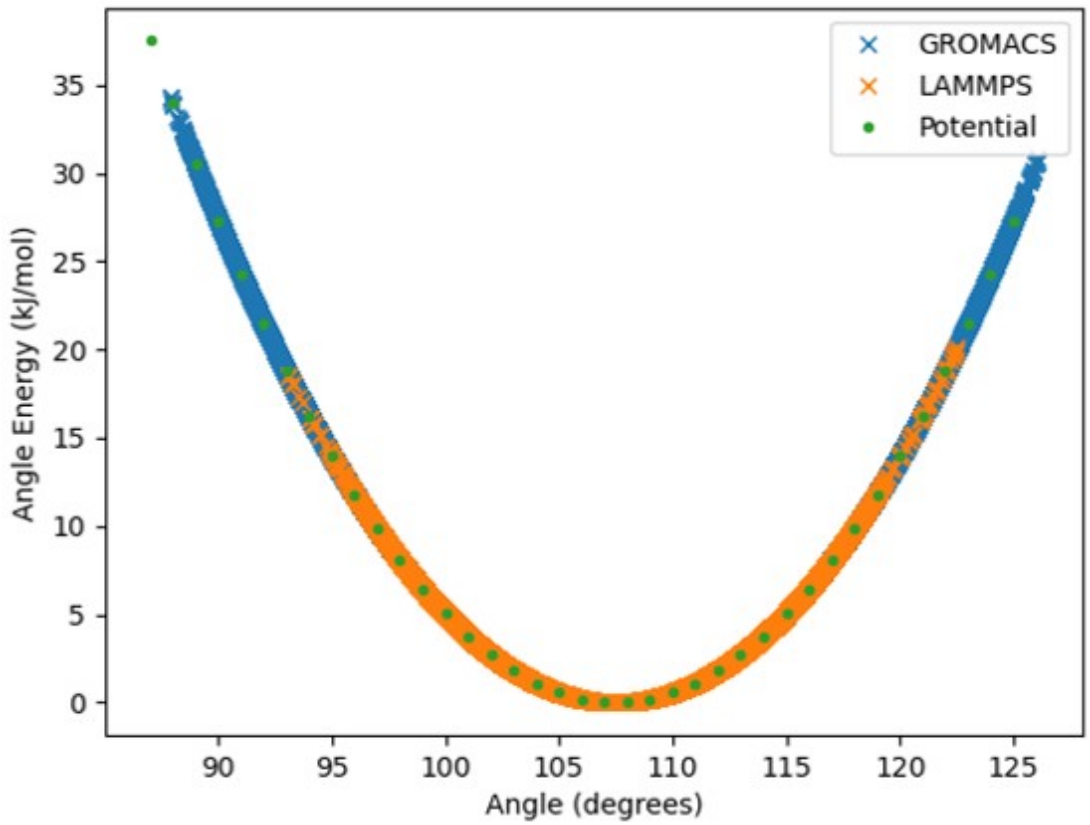


Figure 6.3: The theoretical, LAMMPS, and GROMACS angle potentials for the C-C-O angle in a system with one G trimer and three Na^+ ions.

The trimer simulated in LAMMPS and GROMACS had R_e values of 9.15 ± 1.32 and 7.46 ± 1.45 Å, respectively. These values are within the same range, with the standard deviations overlapping, which shows that the two simulators are simulating alginate in a similar way. The slight difference between the averages and ranges of the R_e is likely due to the low sampling used in the comparison: the systems simulated only one trimer for 1 ns, and each trajectory is expected to be slightly different even in the same simulator.

The RDFs of the alginate trimers to the three Na^+ ions in each system can be seen in Figure 6.4. There is very good agreement between the two simulation software packages, showing that the alginate-ion interactions and Na^+ ion movements are being simulated very similarly between the packages and that the alginate-ion interactions are comparable between simulators. Especially important in this comparison is the distance from the alginate at which the first Na^+ ion RDF peak occurs (as discussed in Chapter 4) and it can be seen in Figure 6.4 that these match almost exactly between the simulators.

Once the intramolecular alginate and intermolecular alginate-ion interaction similarities between LAMMPS and GROMACS had been established, the alginate-alginate interactions were studied to ensure that clustering was also similar between the simulators. The clustering is not expected to be exactly the same, as even different trajectories of the same system in the same simulator would show some differences, but a general consensus shows that the results from each simulator can be directly compared. As can be seen in Figure 6.5, the clustering was very similar between the two simulators with one larger cluster being formed alongside one or two smaller clusters, and at least one lone trimer which did not aggregate with others. To compare the aggregation in each systems quantitatively, the clustering coefficient defined in Section 4.2 was calculated for the last time step of each simulation as presented in Figure 6.5. Both systems had a clustering coefficient of 8.5 showing the same level of clustering between the systems even with differences in the size of clusters formed.

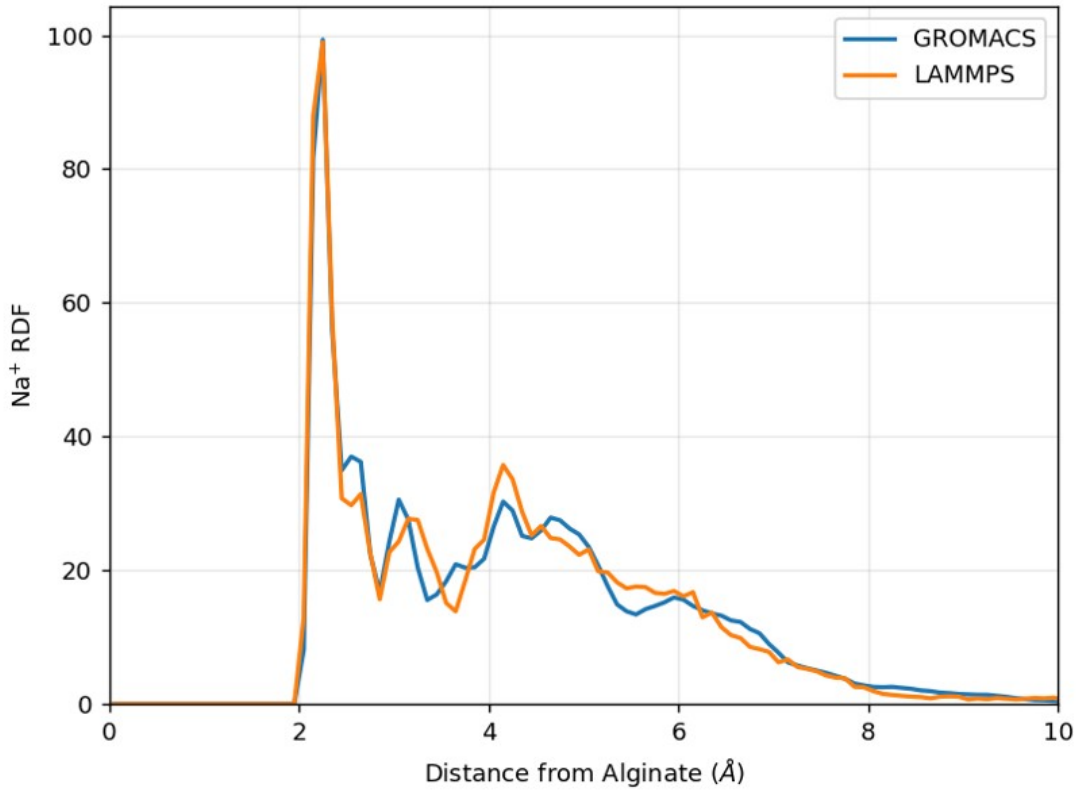


Figure 6.4: Alginate to Na^+ ion RDF for a system with one G trimer and three Na^+ ions simulated in LAMMPS and GROMACS.

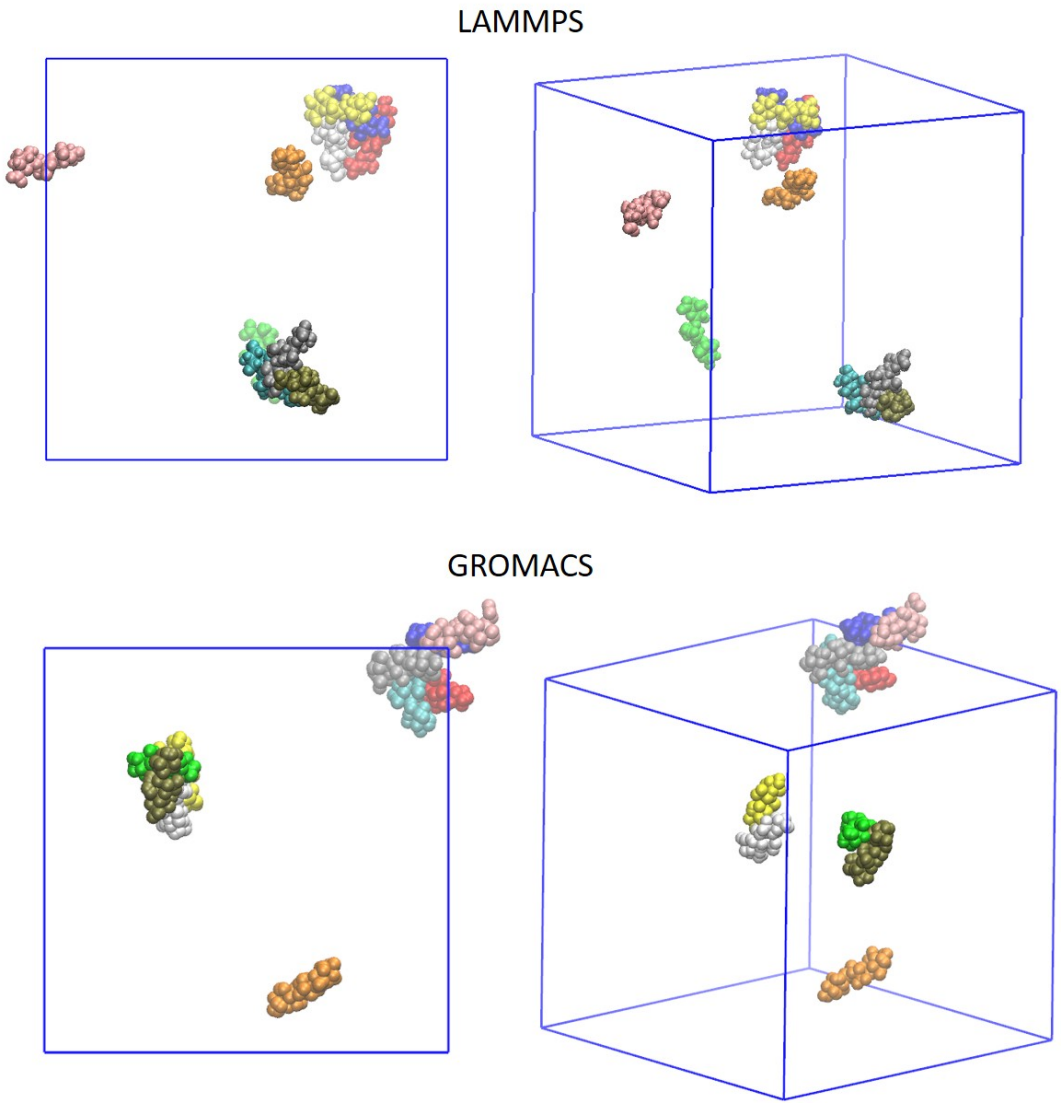


Figure 6.5: Alginate trimer clustering at the final timestep in systems with 10 trimers, 30 Na^+ ions, and 18,851 water molecules run in either LAMMPS (top) or GROMACS (bottom).

6.4 Conclusions from the comparison of LAMMPS and GROMACS alginate systems

The force field and LAMMPS simulation methods established in Chapters 2-4 have been adapted for use in GROMACS. LAMMPS and GROMACS algorithms have been compared, and GROMACS inputs have been optimised for use in simulating alginate systems. The bonded potentials, trimer R_e values, and alginate-Na^+ ion RDFs in a system with one G trimer and 3 Na^+ ions were compared to assess the similarities of species interactions between LAMMPS and GROMACS. Once these similarities were established, system G10-Na, described in Chapter 4, was recreated in GROMACS and run for the same amount of time to establish whether the alginate clustering between the simulators presented any major differences.

These comparisons are in no way extensive with only G and Na^+ systems being compared, no M or Ca^{2+} as in Chapter 4. The simulations were short (1-20 ns), and only a few properties were examined. However, since the properties examined all agreed, it can be concluded that simulations of alginate, ions, and water, in both LAMMPS and GROMACS, set-up as described in this chapter, can be directly compared.

Following this comparison work, alginate simulations were run in GROMACS to expand the work done in Chapter 4 and focus on the interesting Na^+ ion concentration result discussed in Chapter 5. This work is presented in Chapter 7.

Chapter 7

The effect of excess Na^+ ion concentration on alginate structure and cation behaviour in pure and blended alginate systems

7.1 Bonded Parameter Corrections

As discussed at the beginning of Chapter 6, following the completion of the work presented in this Chapter, two errors were found in the compilation of the force field files required to run alginate simulations in GROMACS. The corrected bonded, and non-bonded, parameters are included in Appendix I. It is therefore recommended that the work presented in this Chapter is replicated using the corrected parameters. This Chapter can hopefully be used as a guide for the methodology, analysis, and discussion of system behaviour for simulating dilute ion-alginate systems in GROMACS.

7.2 Introduction

Sodium carbonate is commonly used in the extraction of sodium alginate from seaweed[218–220]. It is important for the ratio of sodium carbonate to alginate to be optimised to prevent excess sodium carbonate being left in the sodium alginate product. The work in Chapter 5 showed that excess Na^+ ions left in a sodium alginate product could significantly decrease the strength of a calcium alginate gel made from the same product. While the experimental methods presented in Chapter 5 were successful in highlight-

ing this effect, they did not offer a way to examine the underlying alginate and ion behaviour changes which occurred between systems with low and high Na^+ ion concentration; molecular dynamics can offer this. This is an under-studied effect which could be vital in customising calcium alginate gel strengths.

While some studies which present alginate simulations do include systems with different Na^+ ion concentrations, none of these investigate the effects of this change in detail, preferring instead to focus on the gel-inducing Ca^{2+} ions or a comparison between Na^+ and Ca^{2+} systems as presented in Chapter 4[146, 148, 149, 152]. These studies, by Li *et al.* and Hecht *et al.* studied G, M, or co-polymer chains in differing alginate, Na^+, and Ca^{2+} concentrations. While a lot of the work looked at chain structure in the form of monomer-monomer angles and persistence length, these values are mostly presented as a function of the alginate or Ca^{2+} concentration or as a comparison between systems with only Na^+ or only Ca^{2+} ions and so a conclusion about the effect of Na^+ ion concentration cannot be made[146, 148, 149, 152]. These studies all use simulated annealing, where the system is heated and cooled in cycles, for simulation times of less than 10 ns, and present the goal of finding optimised alginate structures and cation distributions. While this technique can efficiently find low-energy conformations, it does not capture the equilibrium fluctuations and time-dependent behaviour that MD can provide.

As shown in the study presented in Chapter 4, interesting trends seem to arise when pure systems are compared to blended systems; blended systems contain both M and G residues that are not bonded to each other. The results showed that chain-ion interactions and chain-chain interactions were higher in GM systems with Ca^{2+} ions compared with G or M. To study this further and determine whether it was an effect of short-chain length, a comparison of pure and blended alginate systems is presented here using 24-monomer long alginate chains.

This study is therefore presented as an investigation of the effects on alginate systems of low and high Na^+ ion concentration and of pure and blended alginate chains. Understanding the effects of Na^+ ion concentration highlights ways in which alginate systems, as sodium alginate liquids or calcium alginate gels, can be customised, and

also the ways in which extraction conditions of sodium alginate can affect the final product. A comparison of pure and blended systems is made to extend the interesting findings presented in Chapter 4. To allow as much comparison as possible between Chapters 4, 5, and the work presented here, certain system parameters were chosen to match the previous chapters. These are presented in Table 7.1.

Table 7.1: System parameters in the trimer, experimental, and present study to show comparability.

Chapter	Alginate Conc.	Na^+:Alginate Molar Ratio	Ca^{2+}:Alginate Molar Ratio
4	0.016	1:1	1:2
5	0.01-0.02	1:1-4:1	1:1-2:1
Here	0.016	1:1 and 4:1	1:1

MD simulations were analysed to study the effects on i) polymer conformation and ii) ion and water distribution around the alginate. Each system contained either one chain of G or M, two chains of G or M, or one chain of each of G and M, and a Na^+ ion concentration of either 0.09 M or 0.36 M. By including systems with one G and one M chain, the differences between pure and blended alginate systems could be analysed. The chain structure and bonding conformation, and the ion and water distribution around the chains were studied to present conclusions on the effect of Na^+ ion concentration.

7.3 Methodology

Molecular dynamics (MD) simulations were run using GROMACS[154]. 10 different systems were investigated, each with either 1 or 2 24-monomer long alginate chains and a Na^+ ion concentration of either 0.09 M or 0.36 M. The Na^+ ion concentrations correspond to a 0:1 and 3:1 Na^+ ion:alginate monomer excess molar ratio, respectively, as was used in the experimental work presented in Chapter 5. Also in-keeping with the experimental work, all systems had a Ca^{2+} ion concentration of 0.09 M (corresponding to a 1:1 Ca^{2+} ion:alginate monomer ratio), a Cl^- ion concentration of either 0.18 M or 0.45 M to maintain a system net charge of zero, and a concentration of 0.016 g alginate per g water. These details are summarised, and each system's code is presented, in

Table 7.2. The single-chain systems were each simulated in duplicate to confirm that the observed behaviour was reproducible and not due to simulation-specific artefacts.

Table 7.2: Code and composition of each system where each system has an alginate in water concentration of 0.016 g/g and all alginate chains are 24-monomers long. Single-chain systems were simulated in duplicate.

System Code	Chains		Ions						Number of Water Molecules
	G	M	Number			Conc. (M)			
			Na^+	Ca^{2+}	Cl^-	Na^+	Ca^{2+}	Cl^-	
G-0.09Na	1	0	24	24	48	0.09	0.09	0.18	14634
G-0.36Na	1	0	96	24	120	0.36	0.09	0.45	14634
M-0.09Na	0	1	24	24	48	0.09	0.09	0.18	14634
M-0.36Na	0	1	96	24	120	0.36	0.09	0.45	14634
GG-0.09Na	2	0	48	48	96	0.09	0.09	0.18	29285
GG-0.36Na	2	0	192	48	240	0.36	0.09	0.45	29285
MM-0.09Na	0	2	48	48	96	0.09	0.09	0.18	29285
MM-0.36Na	0	2	192	48	240	0.36	0.09	0.45	29285
GM-0.09Na	1	1	48	48	96	0.09	0.09	0.18	29285
GM-0.36Na	1	1	192	48	240	0.36	0.09	0.45	29285

Alginate was modelled using the GLYCAM-06j force field, the TIP3P model was used to model water, and ion parameters were taken from Li *et al.* (2014 & 2015)[178, 179] as discussed in Chapters 2-4 and 6.

Alginate chains were added either in the centre of (for single-chain systems) or 4 nm apart in the z-direction (for double-chain systems) in a 8x8x8 nm box. 1 nm of space was added around the chains before the system was solvated and ions were added to the desired concentration using the GROMACS *solvate* and *genion* commands. Each G chain and each M chain in all systems had the same starting configurations which are shown in Figure 7.1.

The steepest descent algorithm was used for energy minimisation with a maximum force restraint of 1000 kJ/mol/nm. Alginate heavy atoms were position restrained for a 1 ns NVT followed by 1 ns NPT run with a time step of 1 fs to allow the water and ions to relax. The system was then run for 200 ns NPT with a time step of 2 fs using the Leapfrog algorithm to integrate Newton's equations of motion. The V-rescale and Parrinello-Rahman thermo- and barostats were used throughout this method to maintain a system temperature of 300 K and pressure of 1 atm. Short-range

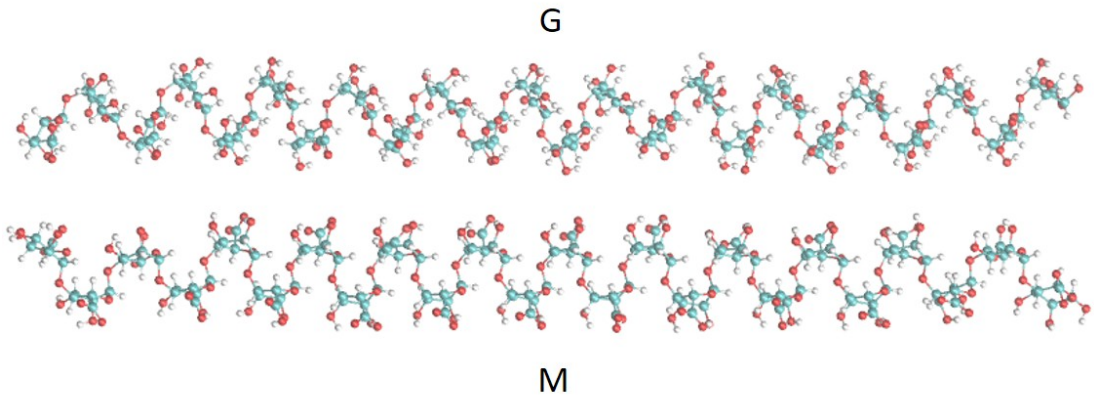


Figure 7.1: The initial configurations of each (top) G and (bottom) M chain used in each simulation.

electrostatics and van der Waals cut-offs of 1 nm were applied throughout with fast smooth particle-mesh Ewald (SPME) applied for long-range electrostatics.

The alginate chains were not relaxed before the system was solvated or ions were included and so care was taken to note when the polymer conformation stabilised before data extraction began. The maximum polymer relaxation seen in the simulations occurred between 50 and 100 ns in an NPT simulation as shown in the end-to-end distance and radius of gyration data in Figure 7.2; the data used in this study was therefore taken from 100 ns onwards. Images from different stages of the simulation of system GG-0.09Na are shown in Figure 7.3 and show an example of the full aggregation and polymer curling which occurred in all systems after approximately 50 ns of simulation time.

The end-to-end distances (R_e) of each chain were calculated as the distance between atoms O4 on the 1'-end and O1 on the 4'-end monomers, where the labelling is shown in Figure 7.4. The R_e values allow analysis of the folding of the chain based on the proximity of the ends of the chain. This data was coupled with the radius of gyration (R_g) which gives details of how far each atom in the chain is, on average, from the centre-of-mass of the chain, which allows the description of the size of the molecule as a whole. The R_g of each chain was calculated using GROMACS command *gmx gyrate*. The chains were also described using the persistence length (L_p) which gives

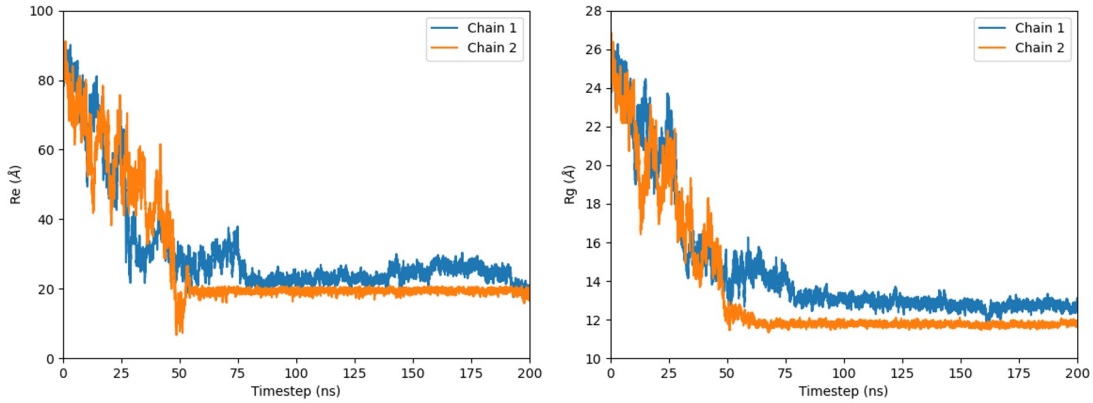


Figure 7.2: R_e and R_g values for system GG-0.09Na over the course of the simulation.

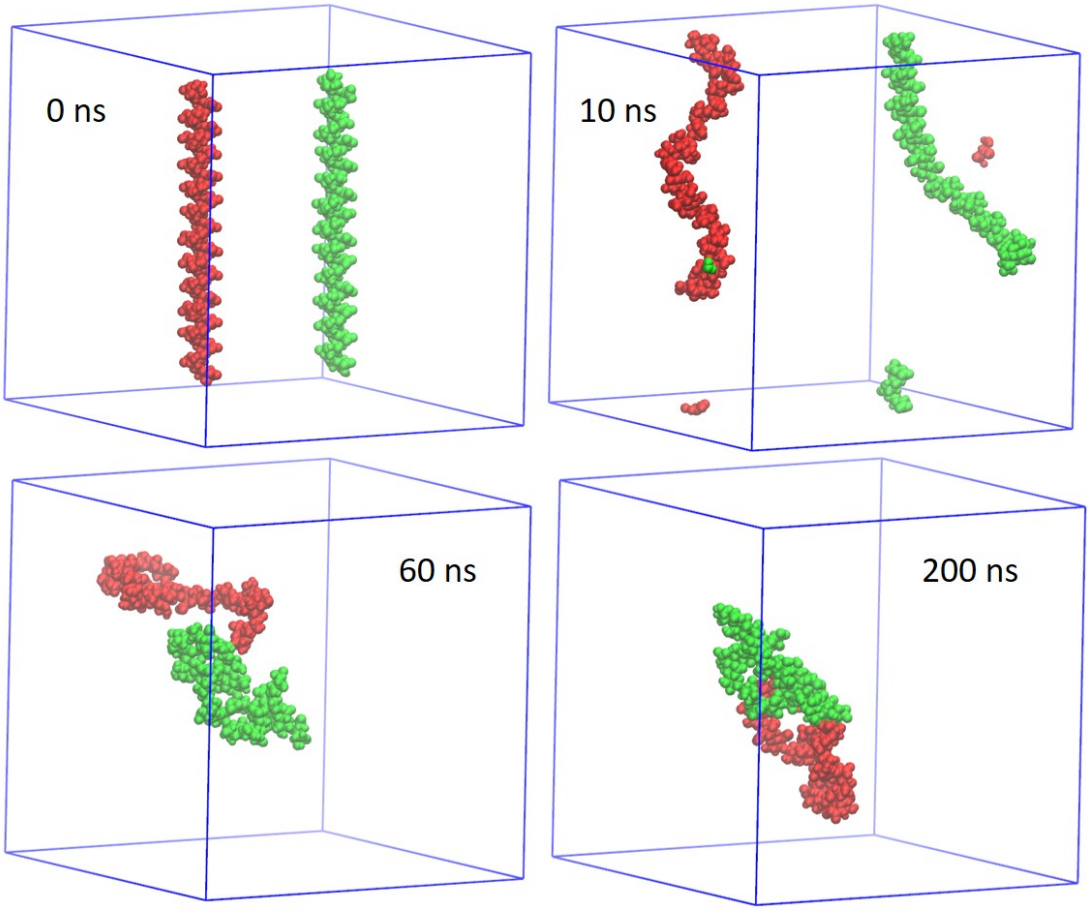


Figure 7.3: Time dependent progress of chain structure in system GG-0.09Na. In the top right box which shows the structures 10 ns into simulation time, parts of the chains have crossed the periodic boundary which is showing as disconnected chains.

details on the distance along the chain at which the chain backbone changes direction, and therefore the stiffness of the chain. The L_p was output using the GROMACS *gmx polystat* command in which the backbone of the chain was described as atoms C1-OG-C4, as labelled in Figure 7.4, along the chain. The solvent-accessible surface area (SASA) was also calculated for each chain or pair of chains to analyse how much of the chain remains accessible to water and ions during the simulations. A probe size of 0.14 nm was implemented with the GROMACS *gmx sasa* command which uses the algorithm detailed by Eisenhaber *et al.*[154, 230]. For double-chain systems, the chain-chain buried surface area (BSA) was also calculated to investigate the extent of the aggregate interface. These values allowed a description of each chain's ability to curl into itself and its stiffness/flexibility, and of the extent of interface which is present between each alginate-alginate pairing.

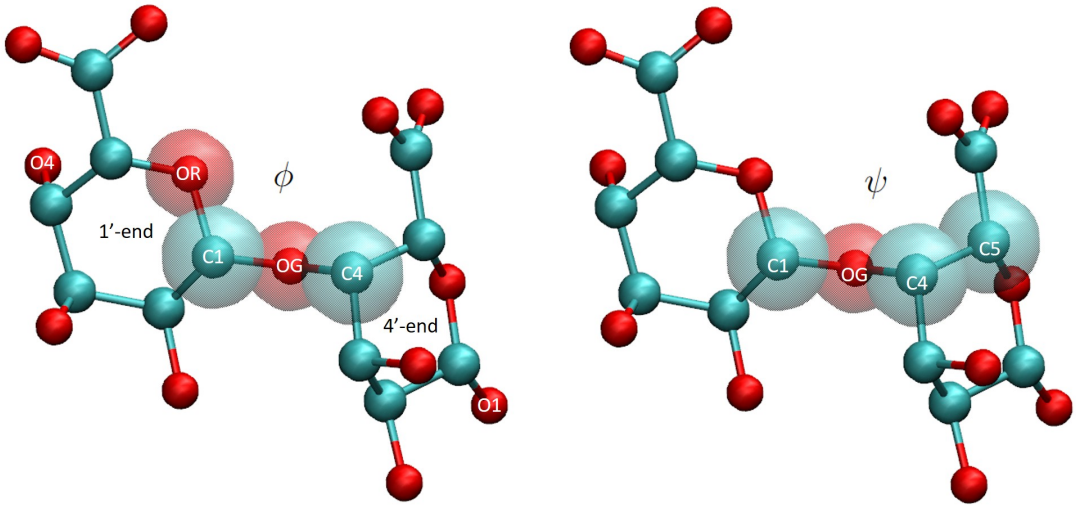


Figure 7.4: An example dimer from simulation GG-0.09Na with atoms used in chain structure analysis and dihedral distributions labelled.

To include details of the internal bonding structures between the monomers of each chain, the angles of the dihedrals highlighted and labelled as ϕ and ψ in Figure 7.4 were output and plotted against each other. This allowed the most commonly occurring monomer-monomer bonding configurations to be identified. A dihedral angle range of -180 to 180°, where negative and positive signs denote the O4 or C5 atom pointing into and out of the screen, and a bin size of 72° were used.

Radial distribution functions (RDFs) were calculated using a bin size of 0.02 Å and a maximum distance of 12 Å. The RDF origin species was taken as all atoms along the alginate chains, and the target species were Ca^{2+} ions, Na^+ ions, Cl^- ions, and water molecules. All-atom to all-atom chain-chain RDFs in the double-chain systems were also calculated to help quantify aggregation, and the proximity of the chains.

System visualisation and output of R_e values, dihedral angles, and RDFs, were completed using VMD software through built-in analysis tools[206].

7.4 Results

We first present the polymer conformation results for the systems at low and high Na^+ ion concentration; this section describes the trends seen in the R_e, R_g, and L_p values. We then discuss the monomer-monomer bonding dihedral distributions, the chain-chain interactions in the double-chain systems, and the alginate-ion and -water interactions. Finally, the differences seen in the double-chain systems of pure (GG/MM) and blended (GM) alginate are presented.

For the majority of the data presented, the values have been averaged over each G or M chain in every system studied. This allows us to perform a comparison between the systems with low and high Na^+ ion concentrations. Where this is not the case, it will be made clear by reference to the specific system(s) being discussed by use of the codes presented in Table 7.2.

7.4.1 Polymer chain size and stiffness

The R_e, R_g, and L_p data for G and M chains in all systems studied, as well as from literature, are shown in Table 7.3. The values from literature are from systems which contained either single[146] or multiple (4-20)[148] 30-monomer chains and Ca^{2+} ion concentrations of 0-1 M.

As seen in Table 7.3, the R_e values from this study are largely unaffected by the Na^+ ion concentration. These values increase from G to M chains in low Na^+ ion concentration but not high. The increase in R_e values from G to M is expected due to the

Table 7.3: R_e, R_g, and L_p results from this study and literature. Units of alginate concentration ("Alg. Conc.") are g alginate/g water.

Chain	Na^+ Conc. (M)	Alg. Conc.	R_e (nm)	R_g (nm)	L_p (nm)	Ref.
G	0.09	0.016	2.3 ± 1.1	1.3 ± 0.3	0.7 ± 0.1	Present
	0.36	0.016	2.5 ± 0.8	1.6 ± 0.5	0.7 ± 0.1	Present
	0-0.003	0.016	11.7		10.4	[148]
	0.3	0.053			7.3 ± 1.0	[146]
	0.5	0.053			6.4 ± 0.6	[146]
	1	0.053			4.9 ± 0.8	[146]
M	0.09	0.016	2.6 ± 0.8	1.3 ± 0.2	1.3 ± 0.4	Present
	0.36	0.016	2.5 ± 0.8	1.2 ± 0.3	1.0 ± 0.1	Present
	0-0.003	0.016	13.8		15.0	[148]
	0.3	0.053			3.8 ± 0.6	[146]
	0.5	0.053			3.3 ± 3	[146]
	1	0.053			8.2 ± 26	[146]

straighter, more ribbon-like monomer bonding exhibited in M chains compared with the more buckled bonding of G chains. The same increase from G to M chains was also seen in the trimer study presented in Chapter 4 in systems with either 0.09 M Na^+ ion concentration and no Ca^{2+} ions, or no Na^+ ions and 0.05 M Ca^{2+} ion concentration. It is suggested from this data that alginate chain R_e is dependent on Na^+ ion concentration only at low values of Na^+ ion concentration, and this effect diminishes once the Na^+ ion concentration is increased to 0.36 M. Analysis of systems with different Na^+ ion concentrations between the values of 0.09 and 0.36 M would be required to confirm this.

The R_e values from literature included in Table 7.3 are significantly higher than those seen in this study. Hecht and Srebnik[148] studied systems with 4-20 30-monomer long alginate chains in implicit water using a simulated annealing methodology. The use of simulated annealing, implemented with a short timescale (< 10 ns) means that it is possible that the curled up structures represented by low R_e values in this study were not reached in the simulation time used. Use of many (4-20) alginate chains is likely to prevent extensive curling of chains due to packing. This would depend on the initial positions of the chains and their distances from each other which is unknown for

the study by Hecht and Srebnik. The use of an implicit water model could also affect the resulting structures, as water plays an important mediation role in alginate-alginate and alginate-ion interactions as shown in the water-ion shells seen in the trimer study presented in Chapter 4. The use of explicit water models is preferred by most alginate simulation studies for this reason, as it allows an in-depth study of the alginate-water and ion-water interactions[150, 152, 182]. Implicit water models are used less often, and seem to be employed to simplify the system[148, 149]. The differences between the results found in this study and those in the study by Hecht and Srebnik show that a direct comparison between the results may not be possible, and highlight the difference in results that can be gained in alginate simulations which utilise different methods.

While the R_e values can give us some details of chain conformation, R_g values give a more inclusive description of the whole chain. The R_g values (i) increased with increasing Na^+ ion concentration for G chains, (ii) decreased with increasing Na^+ ion concentration for M chains, (iii) remained constant moving from G to M chains in low Na^+ ion concentration, and (iv) decreased in moving from G to M chains in high Na^+ ion concentration, as shown in Table 7.3. These trends, and those seen in the R_e data suggest that increasing Na^+ ion concentration produces different effects in G and M chains. In systems with higher Na^+ ion concentration, G chains produce more open structures with more distance between the chain end groups and between each atom and the chain's centre-of-mass. M chains see the opposite behaviour change with increasing Na^+ ion concentration, becoming more closed with lower R_e and R_g values. It can be theorised from these results that different intermolecular interactions are primarily responsible for the chain structures formed by G and M chain types. G chains, well known for their strong interactions with Ca^{2+} ions, create structures based primarily on these strong chain-Ca^{2+} ion interactions. M chains have weaker chain-Ca^{2+} ion interactions and create structures based on both chain-Ca^{2+} ion interactions and monomer-monomer interactions between positively and negatively charged regions of the chain. By increasing Na^+ ion concentration, it is possible that the change in R_e and R_g values indicates a disruption of the chain-Ca^{2+} ion interactions which held the G structure together in the low Na^+ ion concentration system, causing the molecule

to unfurl slightly. For M chains, this increase in Na^+ ion concentration could have more heavily disrupted or screened repulsive forces between negatively charged areas of the chain, compared with the weaker chain-Ca^{2+} ion interactions, causing a slight collapse in the structure compared with the low Na^+ ion concentration system.

The L_p values add to this study by quantifying chain flexibility. No change in L_p values of G chains was seen when increasing Na^+ ion concentration but M chains had a reduction in L_p. L_p increased from G to M chains for both Na^+ ion concentrations. As with the R_e and R_g values, these values suggest different interaction changes between G and M chains. The G chains show no change in flexibility between Na^+ ion concentrations but the M chains become more flexible with increasing Na^+ ion concentration. As with the R_e values, the L_p values from literature are higher than those found in this study. This is thought to be, as described above with the R_e values, due to the different simulation methodologies and the shorter time scales not allowing the chains to curl up as much as they may have done in longer simulations. Li *et al.*[146] used single-chain 30-monomer long chains which were connected over the periodic boundary to utilise an infinite chain model, with TIP3P water, and a simulated annealing methodology. The infinite chain model methodology could prevent the chain fully folding on itself in the way the chains in this study did. The authors of this study state that non-infinite chain model simulations were also run to confirm the reliability of the infinite-chain results and that these simulations were in good agreement with each other but no other information is given, and so it is unclear whether good agreement extends to chain structure.

More systems would need to be simulated and added to this study to provide robustness and confirm the trends seen in Table 7.3 but trends are presenting and theories can be proposed based on these. Changing from 0.09 M to 0.36 M Na^+ ion concentration causes G chains to have a more open structure (increased R_e and R_g values) with no change in the flexibility of the chain (consistent L_p values). This same change in Na^+ ion concentration causes M chains to have a more closed structure (decreased R_e and R_g values) with less flexibility (increased L_p values). It is thought that this is due to the different preferences for intermolecular bonding of G and M chains. G

chains can interact more strongly with Ca^{2+} ions than M chains, which is thought to be due to their buckled monomer bonding forming junction sites. G chains are often described as stiffer and therefore having a higher L_p value than M chains but there have been different conclusions throughout literature[148]. The reportedly high rigidity of poly-G chains, especially in experimental work, is thought to be attributed to the "egg-box" model rather than the poly-G chains themselves[148]. This suggests that G chain rigidity and structure is more dependent on chain-Ca^{2+} ion interactions than chain-chain interactions. The creation of more open G chain structures in systems with high Na^+ ion concentration is therefore thought to be due to the interruption of these chain-Ca^{2+} ion interactions. With less strong chain-Ca^{2+}-chain interactions, caused by an increase in Na^+ ion concentration and the resulting Na^+-Ca^{2+} ion competition, the chains create more open structures. The opposite seems to be true for M chains, which bind less with Ca^{2+} ions. These chains instead see an increased screening effect by the Na^+ ions on the alginate electrostatic repulsive forces causing the chains to somewhat collapse and form tighter structures.

7.4.2 Dihedral distributions

Since the monomer-monomer bonding structure is important in ion coordination, as shown in the proposed "egg-box" model in G chains, it is important to analyse this too. The (ϕ, ψ) plots of all G and M chains studied at 0.09 M and 0.36 M Na^+ ion concentration are shown in Figure 7.5. Examples of the monomer-monomer bonds at highly sampled areas of these plots are labelled and shown in Figure 7.5. The most sampled (ϕ, ψ) combinations are similar in all plots with regions of high sampling around ψ values of -120 to -60° and 90 to 180°. The areas of high sampling on the plot increase from G to M, , as shown in Table 7.4 which presents the percentage of total available (ϕ, ψ) combinations sampled on each plot. This suggests there is a higher flexibility in M monomers to twist around this glycosidic bond. This contradicts the relatively high values of L_p for M chains compared with G chains shown in Table 7.3 which denote lower flexibility in M chains. These results together imply that G chains have an increased flexibility to bend at the glycosidic bond, as seen in their buckled

structure, while M chain flexibility lies in its ability to twist around the glycosidic bond. This agrees with the findings of the trimer study presented in Chapter 4 which presented the same theory. There is also a clear increase in the (ϕ, ψ) sampling when comparing systems of high and low Na^+ ion concentration; showing that both chain types have more freedom to twist around the glycosidic bond in high Na^+ ion concentration. This once again suggests that increasing Na^+ ion concentration disrupts the strongest intermolecular interactions in each chain, allowing more movement.

Table 7.4: Percentage of the (ϕ, ψ) plot that is sampled at least once in G and M systems.

Chain	% of possible (ϕ, ψ) sampling	
	0.09 M	0.36 M
G	70.0	87.3
M	72.2	88.9

Agulhon *et al.* implemented the Hartree-Fock method, likely using a vacuum environment, to find low-energy G and M dimer configurations[231]. Low energy configurations were found at (ϕ, ψ) combinations of approximately (-75, 60), (180, -135), and (-120, -150) for G dimers and (30, -90), (-120, -165), and (-105, 0). These energy minima do not directly correspond to the highly sampled areas shown in Figure 7.5 but this is likely due to the different chain lengths and environments used in the study by Agulhon *et al.* and the present study. The shape of the (ϕ, ψ) plots and the increase in sampling from G to M chains, however, are similar in the study by Agulhon *et al.* and this study, showing a general agreement between the works.

7.4.3 Chain-chain interactions

To help understand how changes in structure-defining properties, as presented in the previous sections, can influence chain-chain interactions in double-chain systems, the buried surface area (BSA) between the chains was calculated. The value of BSA for the GG systems increased by 70.8 % from 6.8 ± 0.7 nm^2 to 11.6 ± 1.4 nm^2, and for the MM systems, a change of -26.4 % from 4.6 ± 2.4 nm^2 to 3.4 ± 0.9 nm^2 was seen when moving from low to high Na^+ ion concentration. By taking these values to be synonymous

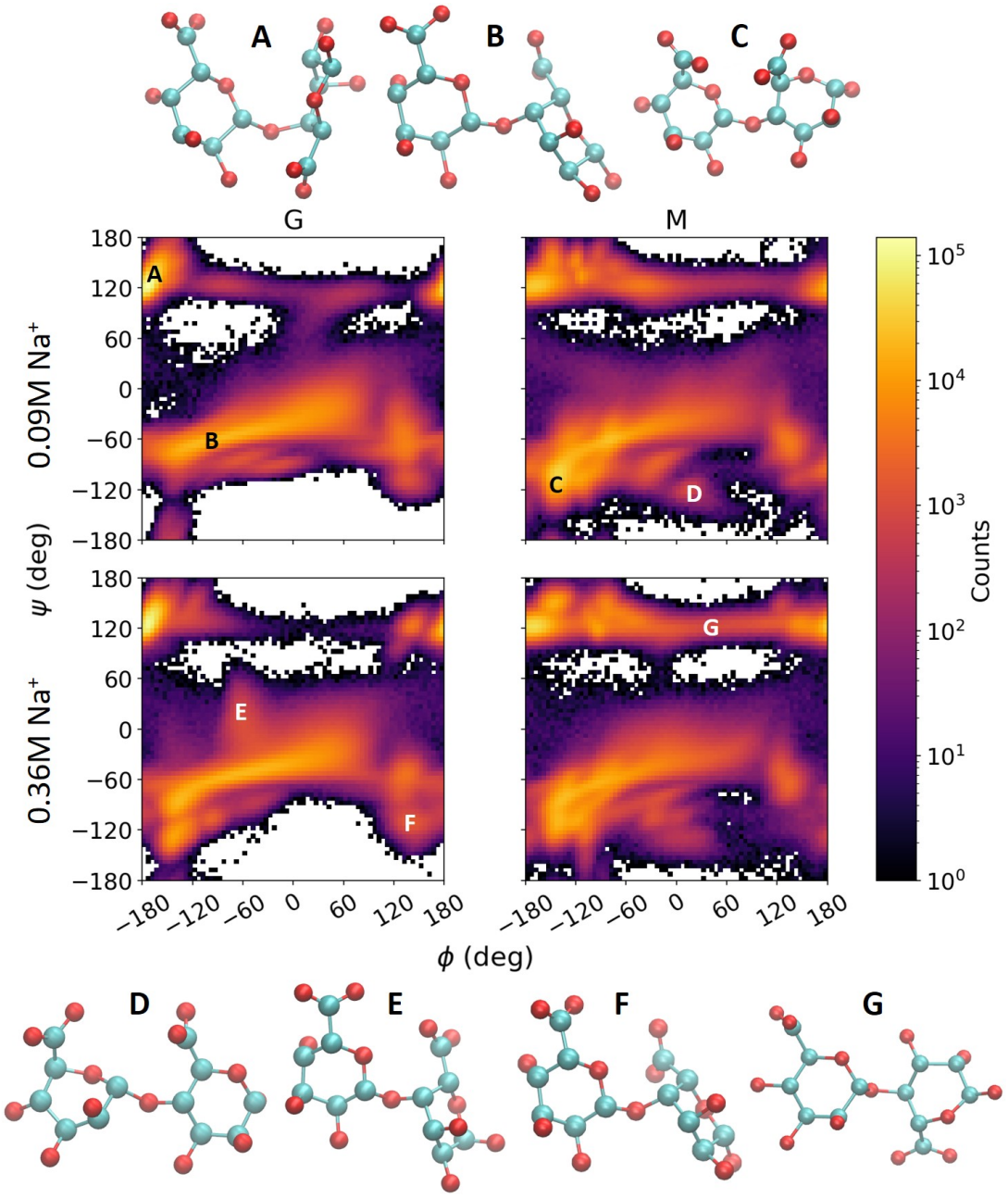


Figure 7.5: Occurrences of conformations defined by the ϕ (C4-O4-C1-OR) and ψ (C5-C4-O4-C1) dihedral angles in the single-chain systems. White space denotes no sampling of that region. Example images of highly sampled areas, labelled A-G, are included.

with the size of the interface between the two chains, we can see that G-G interaction increases with increasing Na^+ ion concentration while M-M interaction decreases with

the same change. This is expected following the more open structures produced in G systems, and more closed structure produced in M systems, with increasing Na^+ ions concentration; a more open structure would be expected to be able to form a larger interface with other molecules. Images of these chain-chain interactions are shown in Figure 7.6. The decrease in M-M interfacial area is visually clear and shows little to no close contact between the chains.

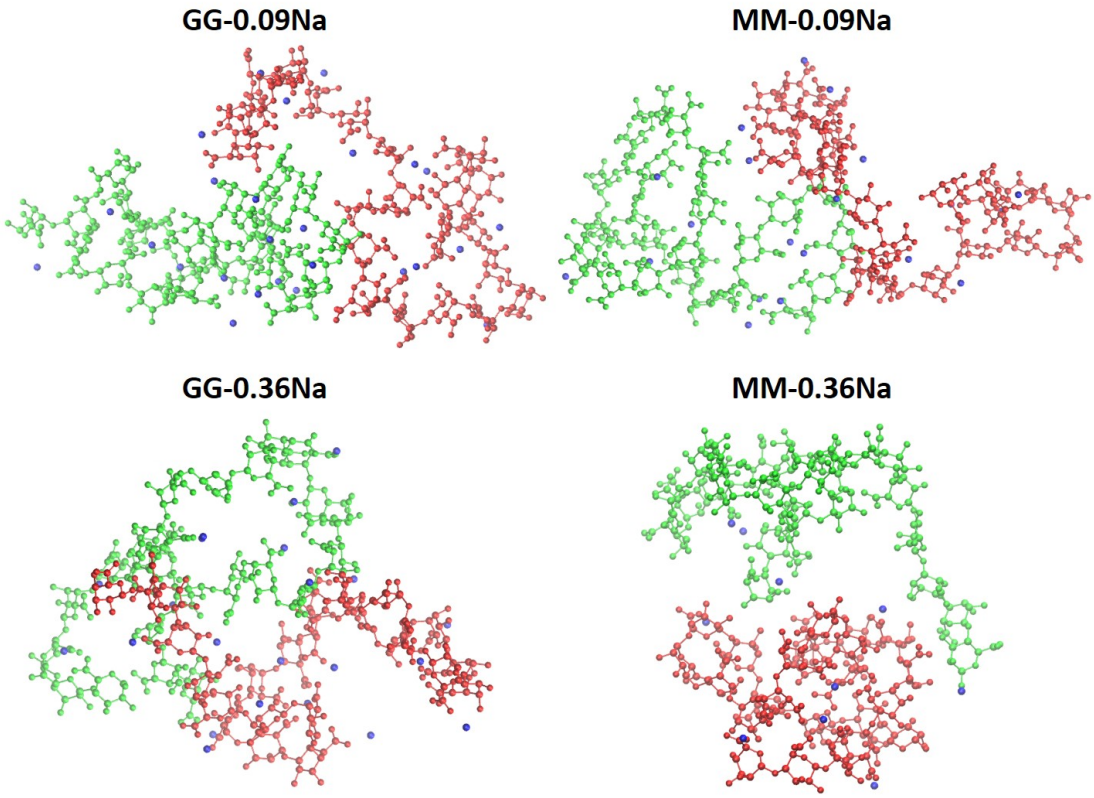


Figure 7.6: Images of alginate chains (green and red) and Ca^{2+} ions (blue) in systems GG-0.09Na, MM-0.09Na, GG-0.36Na, and MM-0.36Na.

To gain a clearer picture of the distance and extent of chain-chain interactions, the chain-chain RDFs were studied; these RDFs are included in Figure 7.7. The RDFs show the same trends as the BSA results with G-G interactions increasing with increasing Na^+ ion concentration, and M-M interactions decreasing overall with the same change. The RDFs show us that the majority of chain-chain interaction occurs at a distance of 1 nm with the interaction distance ranging from 0.14-5 nm. The maximum distance of

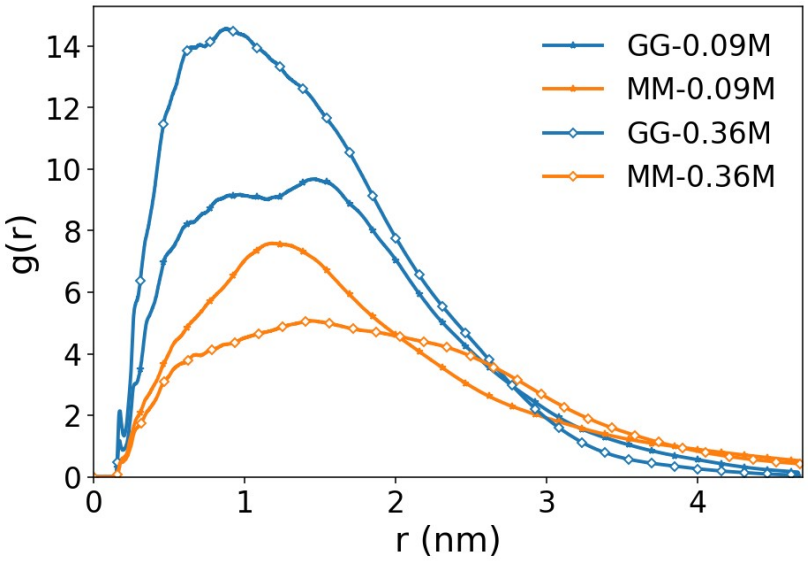


Figure 7.7: Chain to chain RDFs for GG and MM double chain systems.

the RDFs corresponds well with the R_g values presented in Table 7.3. For both GG systems, there is a chain-chain RDF peak which hits its maximum around 0.17 nm, this is greatly reduced and instead presents as a plateau of the RDF in the MM systems. This peak likely corresponds to chain-cation-chain interactions which occur when alginate atoms are close together. The increased height of the G-G interactions at this distance suggests that the chain-cation interactions of G chains are higher than that of M, as theorised in the previous sections.

While the increase in G-G interfacial area and RDFs from low to high Na^+ ion concentrations could imply stronger interactions between the chains, it is not necessarily the case. The chains are closer but there may be less chain-Ca^{2+}-chain intermolecular interactions and so less crosslinking. This effect is shown in Figure 7.6 in which there are less Ca^{2+} ions (blue) around and between the chains of both high Na^+ ion concentration systems compared with the low Na^+ ion concentration systems, but these images show only a snapshot of the simulations. Chain-ion RDFs give us a more inclusive, time-averaged view of the chain-ion interactions and allow us to make more robust hypotheses of strength of the alginate network formed in these systems.

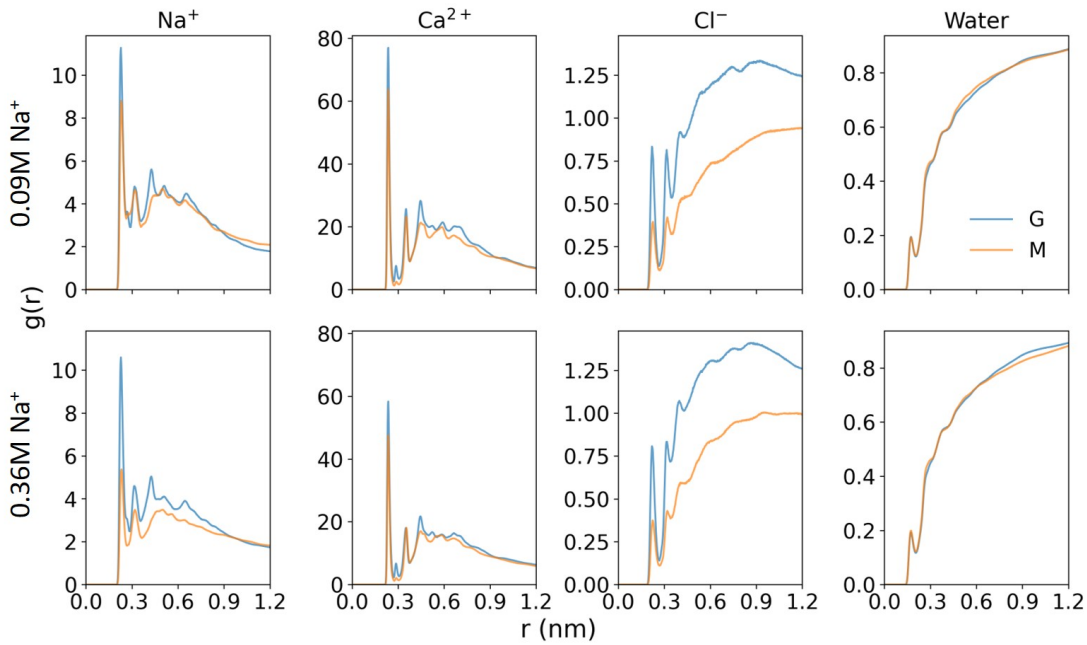


Figure 7.8: RDFs of alginate to Na^+, Ca^{2+}, Cl^-, and water for 0.09 M and 0.36 M Na^+ ion concentration averaged over all G and all M chains studied.

7.4.4 Ion and water radial distribution functions

The alginate-ion and -water RDFs, pooled over all G and M chains studied are shown in Figure 7.8. The RDF peaks show that ion and water shells are present with a water shell at 0.17 nm and an ion shell at 0.22-0.24 nm from the alginate. This is in agreement with the ion and water shells seen in the trimer studied presented in Chapter 4, although the trimer study RDFs presented with much clearer second shells. The lack of clear second shells in this study is likely due to (i) the increased large-distance alginate-ion interactions in this study produced by larger chain lengths compared with trimers and their clusters, and (ii) the use of all chain atoms in the calculation of the RDFs in this study compared with the use of specific oxygen atoms in the trimer study. The shape of the RDFs and the decreased M RDFs compared with G all agree with those presented in previous work[149, 232, 233].

The relative scales of each of the species' RDFs can be explained by the electrostatic attraction between each species and alginate. The relatively high charge of the Ca^{2+} ions presents favourable electrostatic interactions, while the larger water

Table 7.5: The average and maximum Na^+ ion residence times in the first shells over all systems. Shell distances are taken as the end of the first RDF peak shown in Figure 7.8

Chain	Na^+ Residence Time In First Shell (ps)			
	Average		Maximum	
	0.09 M	0.36 M	0.09 M	0.36 M
G	127	176	129598	154400
M	61	86	51662	172188

.

molecules with a net zero charge feel the least electrostatic attraction. The G RDFs were greater for all ion species, showing more interaction between G chains and all ions compared with M chains, independent of the Na^+ ion concentration, as was theorised in the chain-chain RDF discussion above. This is contrary to the results presented in Chapter 4 which presented trends for G or M ion preference that were different for different alginate and ion environments. Since the trimer study investigated systems with either Na^+ or Ca^{2+}, this difference in results with the current study shows the strong effect Ca^{2+} ions have in a Na^+-alginate system and the extent of the ion competition a relatively high Na^+ ion concentration can produce.

The Ca^{2+} ion RDFs reduced for both chain types with an increase in Na^+ ion concentration, indicating increased Na^+-Ca^{2+} ion competition. The Cl^- and water RDFs were largely unaffected, as expected due to the negative or net zero charge on these particles. The alginate-Na^+ RDFs show minimal change in G chains with increasing Na^+ ion concentration, but a significant reduction in the same property for M chains. This could be due to the open G structures allowing more alginate-ion interactions compared with the more closed M structures. Despite these differences in behaviour changes between G and M systems, it can be seen in Table 7.5 that the average and maximum residence time of the Na^+ ions in the first shell around both chain types increased with increasing Na^+ ion concentration. This shows that the Na^+ ions are interacting for longer at any given interaction site. This is likely due to the increased weight Na^+ ions have in the 0.36 M Na^+ ion concentration systems in competition with Ca^{2+} ions compared with in the 0.09 M Na^+ ion concentration systems allowing them to stay in favourable positions longer before a more highly charged Ca^{2+} ion displaces them.

The reduction in Ca^{2+} ion distribution around the G chain with increasing Na^+ ion concentration shows that while G-G interactions increase as shown in Figure 7.7, there is likely to be less crosslinking between these chains. It is likely therefore that, even with the increase in chain-chain interactions shown in the BSA values and chain-chain RDFs presented above, the gel strength would decrease with increasing Na^+ ion concentration in an experimental system. This agrees with the significant decrease in gel strength seen when the same Na^+ ion concentration increase was applied in the experimental work presented in Chapter 5.

7.4.5 Comparison of G-G and M-M with G-M interactions

While the focus of this study was on the behaviour changes in alginate systems when Na^+ ion concentration increased, the G-M interactions seen in systems GM-0.09Na and GM-0.36Na highlighted important differences between pure and blended systems. The main difference which was seen was the increased chain-chain interaction in GM systems compared with chain-chain interactions in GG and MM systems. The BSA for the GM systems were 17.8 ± 2.2 nm^2 and 11.3 ± 0.6 nm^2 for the low and high Na^+ ion concentration systems, respectively. This shows a large increase (161.8 %) from system GG-0.09Na and (287.0 %) from system MM-0.09Na. This interface can be seen in the image of system GM-0.09Na in Figure 7.9 which shows one chain (red) wrapped around the other (green). The interface between these chains was so close, in fact, that six Na^+ ions (also shown in Figure 7.9 in yellow) become trapped between the chains when they came together and did not leave this entrapment within the simulation time. This entrapment is shown in the Na^+ ion first shell residency time of 103.5 ns which occurred in system GM-0.09Na.

While this Na^+ ion trapping behaviour could be a product of the individual trajectory studied, the G-M interactions in system GM-0.36Na were also high as evidenced by the chain-chain RDFs presented in Figure 7.10. This, coupled with the high G-M interaction occurrence relative to G-G and M-M seen in the trimer study presented in Chapter 4, shows that the increased interaction of blended, compared with pure, is likely a genuine property of alginate systems.

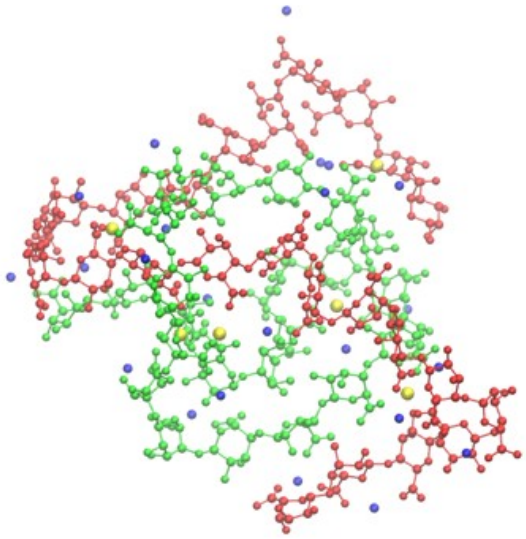


Figure 7.9: Image of the chains, Ca^{2+} (blue), and select Na^+ ions (yellow) in system GM-0.09Na. The Na^+ ions shown became trapped between the two chains.

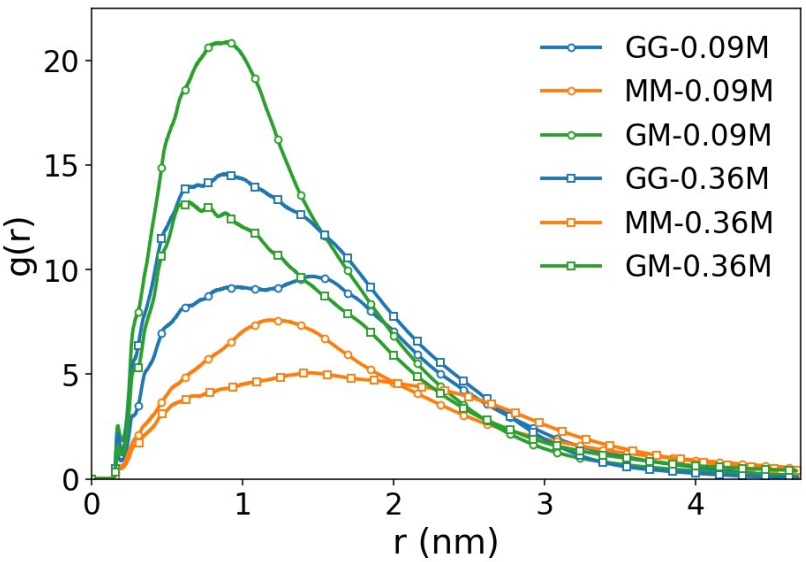


Figure 7.10: The chain-chain RDFs for all double-chain systems.

To confirm that these trends between pure and blended systems are not an effect of the specific trajectories analysed, this study should be expanded to include more systems which have the same compositions but different initial configurations and velocities. It is also suggested that an investigation of the G-G, M-M, and G-M interactions, perhaps through free energy profile investigations, is conducted to help shed light on why G-M blended interactions are presenting as more favourable than G-G or M-M.

7.5 Conclusions

Molecular dynamics systems of alginate chains in dilute aqueous systems with Na^+ and Ca^{2+} salts were analysed to describe the change in behaviour of all species when the Na^+ ion concentration was increased from 0.09 M to 0.36 M. All alginate chains were 24-monomers long, and systems contained either one G and/or one M, two G, or two M chain(s). The systems all had a Ca^{2+} ion concentration of 0.09 M, an alginate in water concentration of 0.016 g/g, and enough Cl^- ions to neutralise the system charge.

From analysis of the chain conformations exhibited in all systems, it was found that an increase in Na^+ ion concentration affects G and M chains differently. G chains form more open structures with high R_e and R_g values while M chains become more closed as shown through their decreased R_e and R_g values. These changes are accompanied by an increase in G-G interactions and a decrease in M-M interactions as shown by the alginate-alginate radial distribution function and buried surface area data. This is thought to be due to the increased chain-Ca^{2+}interactions exhibited by G chains. When Na^+ ion concentration is increased, the strong chain-Ca^{2+}-chain interactions in G systems are disrupted and a more open structure is produced, while in M systems an increase in Na^+ ion concentration causes the chain-chain repulsions to be screened causing a slight collapse of the structure on itself.

From comparison of the L_p values and glycosidic bonding dihedral distributions, it is hypothesised that G chains have more flexibility than M chains in their bending of the glycosidic bond, as shown by their relatively low L_p values. The M chains, however, presented with a higher flexibility than the G chains in their ability to twist around

the glycosidic bond as shown by their relatively highly sampled (ϕ, ψ) plots.

The alginate-ion and -water RDFs showed that water distribution around the alginate chains is largely unaffected by the alginate chain or ion environments. More ion-G interactions occurred than ion-M for all ion species. Less Ca^{2+} ion to chain interactions occurred with increased Na^+ ion concentration, as expected with the increase in ion competition. This shows that although there was an increase in G chain-chain interactions, it is not expected that this would result in a stronger gel because there is less network bridging.

The comparison between the pure and blended systems shows that, as in the trimer studies presented in Chapter 4, G-M interactions were more favourable than G-G or M-M interactions. This confirms that this was not an effect of the short chain length used in the systems presented in Chapter 4 and is an interesting factor which is contrary to the widely held belief that G-G interactions are the most favourable alginate interactions, which could be investigated further.

Trends in the data have been identified and presented in this study but it is recommended that further simulations of the same systems are added to this study to provide robustness. These studies would ideally have the same compositions but have different initial configurations and atom velocities to produce different trajectories. If possible, bigger systems with more, or longer, chains, and a wider range of Na^+ ion concentrations, would also be studied to show how these chains interact and how the properties of bigger alginate systems present, and to allow a wider and more similar comparison to experimental systems to be made. Furthermore, bigger systems run for longer, particularly for more than 1,000 ns, may offer insights into long-chain alginate relaxation, in a way that the (compared to experimental work) short chains and timescales presented here cannot.

Chapter 8

Conclusions & Future Work

The majority of petrochemical plastics are discarded to landfill, where they can take hundreds of years to degrade. Research into replacements using bio-polymers is therefore increasing. Sodium alginate is a naturally occurring polysaccharide that gels in contact with calcium. Alginate is composed of two monomers, α-L-guluronate (G) and β-D-mannuronate (M). The pattern and ratio of these monomers produces different alginate properties.

The aim of this thesis was to investigate the impact of monomer type and ion environment on the behaviour of alginate using computational and experimental investigation methods. First, a force field which describes systems of alginate, water, Na^+, Ca^{2+}, and Cl^- ions was compiled. To complete the force field, the effects of different alginate chain properties on the partial atomic charges were determined using the AM1-BCC charge calculation method, and these charges were optimised for use with any alginate chain. This force field was then used in molecular dynamics simulations of trimers in water with Na^+ or Ca^{2+} ions in which the trimer conformation, water and ion distributions around the trimers, trimer clustering, and species mobility were investigated. Following this, a design of experiments investigation was performed to analyse the significance of the effects of different curing conditions on the strength of calcium alginate gels. To obtain more simulation efficiency, the alginate systems were transferred from LAMMPS to GROMACS, before simulations of 24-monomer long chains were simulated in 0.09 M and 0.36 M Na^+ ion concentration to study the effects of different Na^+ ion concentrations on the behaviour of alginate.

In the molecular dynamics simulations of trimers, each system had 10 G, 10 M, or 5 G and 5 M trimers, either 30 Na^+ or 15 Ca^{2+} ions, and an alginate in water concentration of 0.016 g alginate/g water, allowing the comparison between G and M chains, systems with Na^+ and Ca^{2+} ions, and pure and blended systems. G trimers had more buckled structures shown by their lower end-to-end distances than M trimers, G trimers presented with more flexibility in bending at the glycosidic bond as shown in their end-to-end distance variation, while M timers were more flexible in twisting around this same bond as shown by the dihedral torsion angle distributions. Comparisons of systems with Na^+ or Ca^{2+} ions showed that Ca^{2+} ions induce an increase in trimer bending and increases in both alginate-cation and alginate-alginate interactions. Mixing alginate chains in a blended system with Ca^{2+} ions induced a significant increase in chain-chain clustering when compared with pure G and M systems. While conclusions were made about the behaviour of M and G chains in pure and mixed trimer systems with Na^+ and Ca^{2+} ions, there were limitations in the capacity of these simulations to represent experimental alginate systems due to the short length of the chains, and the low simulation efficiency, which resulted in short simulation timescales, that did not offer the same statistical robustness of longer simulations.

To guide further computational simulations, and gain a broader description of alginate systems, a full factorial design of experiments study was performed. This study presented a systematic and comparative analysis of the most important factors in calcium alginate gel strength. The effects of sodium alginate concentration, G:M ratio, Ca^{2+} ion concentration, Na^+ ion concentration, and pH on calcium alginate gel strength were investigated. Gel samples made with different curing conditions were compression strength tested using a Texture Analyser. It was found that all factors except pH of the alginate aqueous solvent were significant at a confidence level of 95 %. Na^+ ion concentration had a highly significant effect on gel strength, inducing a decrease in gel strength of 30.8 % from a Na^+ ion:alginate monomer molar ratio of 1:1 to 4:1.

To further examine this effect, a second molecular dynamics study was undertaken. To overcome limitations in the previous simulation study brought on by short chain lengths and simulation times, it was decided to transfer the model from the LAMMPS to

GROMACS software package. The full integration of GPUs in the calculation of long-range electrostatics, which GROMACS offers, allowed a simulation efficiency increase from 4 to 380 ns/day.

The second molecular dynamics study investigated the effects of the same Na^+ ion concentrations studied in the experimental study on the behaviour of 24-monomer long G and M alginate chains in dilute systems. Single or double alginate chain systems had 1:1 Ca^{2+} ion:alginate monomer ratio, 1:1 or 4:1 Na^+ ion:alginate monomer ratio, and an alginate in water concentration of 0.016 g/g. Increased Na^+ ion concentration resulted in G chains forming more closed structures (with lower end-to-end distance and radius of gyration values) and presenting with a larger chain-chain interfacial area. M chains reacted in the opposite way, forming more open structures with a decrease in chain-chain interfacial area. Comparisons between G-G/M-M and G-M interactions showed that, as was seen in the trimer study, G-M interactions were more favourable than G-G or M-M interactions. Following the completion of this work, two errors were discovered in the bonded parameters used to simulate alginate. It was outwith the scope of this work to replicate the simulations with the corrected parameters; the corrected parameters are included within this document to aid further work.

Following this work, the force field, including partial atomic charges, and alginate structure creation python script can be used to simulate alginate systems in both LAMMPS and GROMACS simulation software packages. It is recommended, as alginate simulation studies become more prevalent, that the available force fields are tested against experimental results to analyse their validity, and the advantages and disadvantages of each.

The aims and objectives of this thesis have, therefore, been achieved. However, further work would provide robustness to this work. While vast improvements were made in the possible statistical robustness of the molecular dynamics studies by changing from the LAMMPS to GROMACS simulation software package, this could be further improved by increasing the alginate chain length and the size of the systems (number of alginate chains) studied to better mimic experimental studies, and including more Na^+ ion concentrations to more widely explore the effect Na^+ ion concentration has on

alginate system behaviour.

Further work for the experimental work presented in this thesis would ideally be a response surface model design of experiments. This would allow the curvature, identified in the design of experiments presented in this thesis, to be fully modelled and described. It is also recommended that the mass and volume of the gels before and after curing are noted. This would allow the degree of syneresis in each gel to be calculated and a quantitative analysis of the alginate network constriction during gelling to be performed. It is also recommended that mixtures of pure G and M alginate are used in addition to differing G:M ratio alginate as part of the further experimental work to allow the comparison of the effects of blended alginate with co-polymer alginate in alignment with the results from both simulation studies presented in this thesis.

Following the completion of the simulations presented in Chapter 7 with the corrected bonded parameters, it is recommended that the interactions between alginate chains (G-G, M-M, and G-M) are studied in more depth computationally to understand better why G-M interactions presented as more favourable than G-G or M-M interactions. This could be carried out using umbrella sampling to calculate the free energy profiles of pulling two alginate chains apart. By creating different environments between the two chains (*e.g.* vacuum, water, ions, water and ions) and applying the same pulling technique to each system, the strengths and coordination of the interactions between different alginate chains could be quantified. This could also be used to investigate the "egg-box" model further; this work could use different lengths of G chains, different configurations, different positions of cations, and other variations, to compare between systems. A trial run of umbrella sampling used to pull apart alginate chains in a water and ion environment was performed and is presented in Appendix J to guide this work.

Other properties whose analysis would allow a better description of alginate behaviour to be made include the effects of different ions other than sodium and calcium and the alteration of alginate hydroxyl groups into other functional side groups. This research has been presented previously but is currently limited, especially in computational studies.

Bibliography

[1] Matthew B. Stewart, Stephen R. Gray, Todor Vasiljevic, and John D. Orbell. The role of poly-m and poly-gm sequences in the metal-mediated assembly of alginate gels. *Carbohydrate Polymers*, 112:486–493, 11 2014.

[2] Richard C. Thompson, Charles J. Moore, Frederick S.Votn Saal, and Shanna H. Swan. Plastics, the environment and human health: current consensus and future trends. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 364:2153–2166, 7 2009.

[3] Plastics Europe. Plastics – the fast facts 2024 • plastics europe, 2024.

[4] National Academies. Recycled plastics in infrastructure: Current practices, understanding, and opportunities - appendix j, 7 2023.

[5] Carlos F. Jasso-Gastinel, J. F.A. Soltero-Martínez, and E. Mendizábal. Introduction: Modifiable characteristics and applications. *Modification of Polymer Properties*, pages 1–21, 1 2017.

[6] Robert O Ebewele, Boca Raton, and New York. *POLYMER SCIENCE AND TECHNOLOGY*. 2000.

[7] Rudolph D Deanin. The chemistry of plastics. *Journal of Chemical Education*, 64:45–47, 1987.

[8] Lea Marie Heidbreder, Isabella Bablok, Stefan Drews, and Claudia Menzel. Tackling the plastic problem: A review on perceptions, behaviors, and interventions. *Science of The Total Environment*, 668:1077–1093, 6 2019.

[9] Mehmet Seckin Aday and Ugur Yener. Understanding the buying behaviour of young consumers regarding packaging attributes and labels. *International Journal of Consumer Studies*, 38:385–393, 7 2014.

[10] Catherine Phillips. Alternative food distribution and plastic devices: Performances, valuations, and experimentations. *Journal of Rural Studies*, 44:208–216, 4 2016.

[11] Laura Parker. The world's plastic pollution crisis, explained — national geographic, 2025.

[12] Our plastic pollution campaigns - surfers against sewage.

[13] Plastic pollution - greenpeace uk.

[14] Plastic pollution.

[15] Plastics — european environment agency's home page.

[16] Public awareness - plastic smart cities.

[17] Imogen E. Napper, Bede F.R. Davies, Heather Clifford, Sandra Elvin, Heather J. Koldewey, Paul A. Mayewski, Kimberley R. Miner, Mariusz Potocki, Aurora C. Elmore, Ananta P. Gajurel, and Richard C. Thompson. Reaching new heights in plastic pollution—preliminary findings of microplastics on mount everest. *One Earth*, 3:621–630, 11 2020.

[18] A. J. Jamieson, L. S.R. Brooks, W. D.K. Reid, S. B. Piertney, B. E. Narayanaswamy, and T. D. Linley. Microplastics and synthetic particles ingested by deep-sea amphipods in six of the deepest marine ecosystems on earth. *Royal Society Open Science*, 6, 2 2019.

[19] Paul T Anastas and John C Warner. Green chemistry: Theory and practice. 5 2000.

[20] Parikshit Sarda, Jay C. Hanan, Joseph G. Lawrence, and Masoud Allahkarami. Sustainability performance of polyethylene terephthalate, clarifying challenges and opportunities. *Journal of Polymer Science*, 60:7–31, 1 2022.

[21] Sedat Kumartasli and Ozan Avinc. Important step in sustainability: Polyethylene terephthalate recycling and the recent developments. pages 1–19, 2020.

[22] Zouhair Ait-Touchente, Maya Khellaf, Guy Raffin, Noureddine Lebaz, and Abdelhamid Elaissari. Recent advances in polyvinyl chloride (pvc) recycling. *Polymers for Advanced Technologies*, 35:e6228, 1 2024.

[23] Valentina Siracusa and Ignazio Blanco. Bio-polyethylene (bio-pe), bio-polypropylene (bio-pp) and bio-poly(ethylene terephthalate) (bio-pet): Recent developments in bio-based polymers analogous to petroleum-derived ones for packaging and engineering applications. *Polymers 2020, Vol. 12, Page 1641*, 12:1641, 7 2020.

[24] H. P.S. Abdul Khalil, E. W.N. Chong, F. A.T. Owolabi, M. Asniza, Y. Y. Tye, S. Rizal, M. R. Nurul Fazita, M. K. Mohamad Haafiz, Z. Nurmiati, and M. T. Paridah. Enhancement of basic properties of polysaccharide-based composites with organic and inorganic fillers: A review, 3 2019.

[25] Muhammad Khusairy Bin Bakri, Md Rezaur Rahman, and Faisal Islam Chowdhury. Sources of cellulose. *Fundamentals and Recent Advances in Nanocomposites Based on Polymers and Nanocellulose*, pages 1–18, 1 2022.

[26] Nida Khan, K. Sudhakar, and R. Mamat. Biodegradable plastics from marine biomass: A solution to marine plastic pollution. *Journal of Hazardous Materials Advances*, 17:100559, 2 2025.

[27] Tugce Senturk Parreidt, Kajetan Müller, and Markus Schmid. Alginate-based edible films and coatings for food packaging applications. *Foods*, 7, 10 2018.

[28] Harvey Tyler-Walters and Joanna Jones. Tangle or cuvie (laminaria hyperborea), 2008.

[29] Jacqueline Hill, Nicola White, and Dagmar Stengel. Knotted wrack (ascophyllum nodosum), 2008.

[30] Ivan Donati, Sergio Paoletti, I Donati, and S Paoletti. Material properties of alginates, 2009.

[31] Olav Smidsrød and Arne Haug. Precipitation of acidic polysaccharides by salts in ethanol–water mixtures. *Journal of Polymer Science Part C: Polymer Symposia*, 16:1587–1598, 1 1967.

[32] Md Mostafizur Rahman, Md Abdus Shahid, Md Tanvir Hossain, Md Sohan Sheikh, Md Sunjidur Rahman, Nasir Uddin, Abdur Rahim, Ruhul Amin Khan, and Imam Hossain. Sources, extractions, and applications of alginate: a review. *Discover Applied Sciences 2024 6:8*, 6:1–31, 8 2024.

[33] Iain D. Hay, Zahid Ur Rehman, M. Fata Moradali, Yajie Wang, and Bernd H.A. Rehm. Microbial alginate production, modification and its applications, 11 2013.

[34] Lanzeng Zhang, Xue Li, Xiyue Zhang, Yingjie Li, and Lushan Wang. Bacterial alginate metabolism: an important pathway for bioconversion of brown algae. *Biotechnology for Biofuels 2021 14:1*, 14:1–18, 7 2021.

[35] Richard Fischl, Kresten Bertelsen, Fanny Gaillard, Susana Coelho, Gurvan Michel, Markus Klinger, Catherine Boyen, Mirjam Czjzek, and Cécile Hervé. The cell-wall active mannuronan c5-epimerases in the model brown alga ectocarpus: From gene context to recombinant protein. *Glycobiology*, 26:973–983, 9 2016.

[36] Anne Tøndervik, Olav A. Aarstad, Randi Aune, Susan Maleki, Philip D. Rye, Arne Dessen, Gudmund Skjåk-Bræk, and Håvard Sletta. Exploiting mannuronan c-5 epimerases in commercial alginate production. *Marine Drugs 2020, Vol. 18, Page 565*, 18:565, 11 2020.

[37] Jan Muschiol, Morten Schiøtt, Maria Dalgaard Mikkelsen, and Anne S. Meyer.

Mannuronan c5-epimerases for tailored seaweed alginate modification. *Algal Research*, 90:104126, 8 2025.

[38] Trond Helgerud, Olav Gåserød, Therese Fjæreide, Peder O. Andersen, and Christian Klein Larsen. Alginates. *Food Stabilisers, Thickeners and Gelling Agents*, pages 50–72, 10 2009.

[39] Kurt Ingar Draget, Gudmund Skjåk-Bræk, and Bjørn Torger Stokke. Similarities and differences between alginic acid gels and ionically crosslinked alginate gels. *Food Hydrocolloids*, 20:170–175, 3 2006.

[40] Su Hung Ching, Nidhi Bansal, and Bhesh Bhandari. Alginate gel particles–a review of production techniques and physical properties. *Critical Reviews in Food Science and Nutrition*, 57:1133–1152, 4 2017.

[41] Øyvind Skaugrud, Anniken Hagen, Berit Borgersen, Michael Dornish, and Yvind Skaugrud Anniken Hagen. Biomedical and pharmaceutical applications of alginate and chitosan. *Biotechnology and Genetic Engineering Reviews*, 16:23–40, 1999.

[42] Chuhuan Hu, Wei Lu, Analucia Mata, Katsuyoshi Nishinari, and Yapeng Fang. Ions-induced gelation of alginate: Mechanisms and applications, 4 2021.

[43] Ýrr A. Mørch, Ivan Donati, Berit L. Strand, and Gudmund Skjåk-Bræk. Effect of ca2+, ba2+, and sr2+ on alginate microbeads. *Biomacromolecules*, 7:1471–1480, 5 2006.

[44] Siddhesh N. Pawar and Kevin J. Edgar. Alginate derivatization: A review of chemistry, properties and applications. *Biomaterials*, 33:3279–3305, 4 2012.

[45] Gregor T. Grant, Edwin R. Morris, David A. Rees, Peter J.C. Smith, and David Thom. Biological interactions between polysaccharides and divalent cations: The egg-box model. *FEBS Letters*, 32:195–198, 1973.

[46] Kate A. Bowman, Olav Andreas Aarstad, Marcela Nakamura, Bjørn Torger Stokke, Gudmund Skjåk-Bræk, and Andrew N. Round. Single molecule inves-

tigation of the onset and minimum size of the calcium-mediated junction zone in alginate. *Carbohydrate Polymers*, 148:52–60, 9 2016.

[47] Narin Paiboon, Suvimol Surassmo, Uracha Rungsardthong Ruktanonchai, Michael Kappl, and Apinan Soottitantawat. Internal gelation of alginate microparticle prepared by emulsification and microfluidic method: Effect of ca-edta as a calcium source. *Food Hydrocolloids*, 141:108712, 8 2023.

[48] Ushasree Mrudulakumari Vasudevan, Ok Kyung Lee, and Eun Yeol Lee. Alginate derived functional oligosaccharides: Recent developments, barriers, and future outlooks. *Carbohydrate Polymers*, 267:118158, 9 2021.

[49] Anastasia E. Kapetanakou, Sofia Nestora, Vasiliki Evageliou, and Panagiotis N. Skandamis. Sodium alginate–cinnamon essential oil coated apples and pears: Variability of aspergillus carbonarius growth and ochratoxin a production. *Food Research International*, 119:876–885, 5 2019.

[50] Nazia Tabassum and Mohammad Ali Khan. Modified atmosphere packaging of fresh-cut papaya using alginate based edible coating: Quality evaluation and shelf life study. *Scientia Horticulturae*, 259:108853, 1 2020.

[51] Rosa M. Raybaudi-Massilia, María A. Rojas-Graü, Jonathan Mosqueda-Melgar, and Olga Martín-Belloso. Comparative study on essential oils incorporated into an alginate-based edible coating to assure the safety and quality of fresh-cut fuji apples. *Journal of Food Protection*, 71:1150–1161, 6 2008.

[52] Adriana C. Guerreiro, Custódia M.L. Gago, Maria L. Faleiro, Maria G.C. Miguel, and Maria D.C. Antunes. The effect of alginate-based edible coatings enriched with essential oils constituents on arbutus unedo l. fresh fruit storage. *Postharvest Biology and Technology*, 100:226–233, 2 2015.

[53] Yue Zhang, Qiumin Ma, Faith Critzer, P. Michael Davidson, and Qixin Zhong. Effect of alginate coatings with cinnamon bark oil and soybean oil on quality and microbiological safety of cantaloupe. *International Journal of Food Microbiology*, 215:25–30, 12 2015.

[54] Nima Azarakhsh, Azizah Osman, Hasanah Mohd Ghazali, Chin Ping Tan, and Noranizan Mohd Adzahan. Lemongrass essential oil incorporated into alginate-based edible coating for shelf-life extension and quality retention of fresh-cut pineapple. *Postharvest Biology and Technology*, 88:1–7, 2 2014.

[55] Rosa M. Raybaudi-Massilia, Jonathan Mosqueda-Melgar, and Olga Martín-Belloso. Edible alginate-based coating as carrier of antimicrobials to improve shelf-life and safety of fresh-cut melon. *International Journal of Food Microbiology*, 121:313–327, 2 2008.

[56] Yasser Shahbazi and Nassim Shavisi. Effects of sodium alginate coating containing mentha spicata essential oil and cellulose nanoparticles on extending the shelf life of raw silver carp (hypophthalmichthys molitrix) fillets. *Food Science and Biotechnology*, 28:433–440, 4 2019.

[57] Behnaz Bazargani-Gilani. Activating sodium alginate-based edible coating using a dietary supplement for increasing the shelf life of rainbow trout fillet during refrigerated storage (4 ± 1 °c). *Journal of Food Safety*, 38:e12395, 2 2018.

[58] Mehdi Alboofetileh, Masoud Rezaei, Hedayat Hosseini, and Mehdi Abdollahi. Efficacy of activated alginate-based nanocomposite films to control listeria monocytogenes and spoilage flora in rainbow trout slice. *Journal of Food Science and Technology*, 53:521–530, 1 2016.

[59] Nastaran Jalali, Peiman Ariiai, and Esmaeil Fattahi. Effect of alginate/carboxyl methyl cellulose composite coating incorporated with clove essential oil on the quality of silver carp fillet and escherichia coli o157:h7 inhibition during refrigerated storage. *Journal of Food Science and Technology*, 53:757–765, 1 2016.

[60] Seyede Marzieh Kazemi and Masoud Rezaei. Antimicrobial effectiveness of gelatin-alginate film containing oregano essential oil for fish preservation. *Journal of Food Safety*, 35:482–490, 11 2015.

[61] Fei Lu, Yuting Ding, Xingqian Ye, and Donghong Liu. Cinnamon and nisin in

alginate–calcium coating maintain quality of fresh northern snakehead fish fillets. *LWT - Food Science and Technology*, 43:1331–1335, 11 2010.

[62] Fateme Sharifi, Saeid Khanzadi, Mohammad Hashemi, and Mohammad Azizzadeh. Control of listeria monocytogenes and escherichia coli o157:h7 inoculated on fish fillets using alginate coating containing lactoperoxidase system and zataria multiflora boiss essential oil. *Journal of Aquatic Food Product Technology*, 26:1014–1021, 10 2017.

[63] Dilara Konuk Takma and Figen Korel. Active packaging films as a carrier of black cumin essential oil: Development and effect on quality and shelf-life of chicken breast meat. *Food Packaging and Shelf Life*, 19:210–217, 3 2019.

[64] Ana Carolina Pelaes Vital, Ana Guerrero, Jessica De Oliveira Monteschio, Maribel Velandia Valero, Camila Barbosa Carvalho, Benício Alves De Abreu Filho, Grasiele Scaramal Madrona, and Ivanor Nunes Do Prado. Effect of edible and active coating (with rosemary and oregano essential oils) on beef characteristics and consumer acceptability. *PLOS ONE*, 11:e0160535, 8 2016.

[65] Mojtaba Raeisi, Alijan Tabaraei, Mohammad Hashemi, and Nasser Behnampour. Effect of sodium alginate coating incorporated with nisin, cinnamomum zeylanicum, and rosemary essential oils on microbial quality of chicken meat and fate of listeria monocytogenes during refrigeration. *International Journal of Food Microbiology*, 238:139–145, 12 2016.

[66] Mounia Oussalah, Stéphane Caillet, Stéphane Salmiéri, Linda Saucier, and Monique Lacroix. Antimicrobial effects of alginate-based film containing essential oils for the preservation of whole beef muscle. *Journal of Food Protection*, 69:2364–2369, 10 2006.

[67] Marwa Ragab Abdallah, Mai Atef Mohamed, Hussein Mohamed, and Mohamed Talaat Emara. Application of alginate and gelatin-based edible coating materials as alternatives to traditional coating for improving the quality of pastirma. *Food Science and Biotechnology*, 27:1589–1597, 12 2018.

[68] Zaneta Król, Dominika Kulig, Krzysztof Marycz, Anna Zimoch-Korzycka, and Andrzej Jarmoluk. The effects of using sodium alginate hydrosols treated with direct electric current as coatings for sausages. *Polymers 2017, Vol. 9, Page 602*, 9:602, 11 2017.

[69] Mounia Oussalah, Stéphane Caillet, Stéphane Salmiéri, Linda Saucier, and Monique Lacroix. Antimicrobial effects of alginate-based films containing essential oils on listeria monocytogenes and salmonella typhimurium present in bologna and ham. *Journal of Food Protection*, 70:901–908, 4 2007.

[70] Ghada Ksouda, Sabrine Sellimi, Franck Merlier, Aude Falcimaigne-cordin, Brigitte Thomasset, Moncef Nasri, and Mohamed Hajji. Composition, antibacterial and antioxidant activities of pimpinella saxifraga essential oil and application to cheese preservation as coating additive. *Food Chemistry*, 288:47–56, 8 2019.

[71] V. Chiabrando and G. Giacalone. Quality evaluation of blueberries coated with chitosan and sodium alginate during postharvest storage. *International Food Research Journal*, 24:1553–1561, 2017.

[72] Ramin Heydari, Shahmir Bavandi, and Seyed Roholla Javadian. Effect of sodium alginate coating enriched with horsemint (mentha longifolia) essential oil on the quality of bighead carp fillets during storage at 4°c. *Food Science Nutrition*, 3:188–194, 5 2015.

[73] Roxana Gheorghita Puscaselu, Andrei Lobiuc, Mihai Dimian, and Mihai Covasa. Alginate: From food industry to biomedical applications and management of metabolic disorders. *Polymers 2020, Vol. 12, Page 2417*, 12:2417, 10 2020.

[74] Carla Vilela, Mia Kurek, Zvi Hayouka, Bettina Röcker, Selçuk Yildirim, Maria Dulce C. Antunes, Julie Nilsen-Nygaard, Marit Kvalvåg Pettersen, and Carmen S.R. Freire. A concise guide to active agents for active food packaging. *Trends in Food Science Technology*, 80:212–222, 10 2018.

[75] Dheebika Natrajan, Sharmila Srinivasan, K. Sundar, and Aswathy Ravindran.

Formulation of essential oil-loaded chitosan–alginate nanocapsules. *Journal of Food and Drug Analysis*, 23:560–568, 9 2015.

[76] Nanik Purwanti, Azmi Syahrian Zehn, Eka Dian Pusfitasari, Nauman Khalid, Erfin Yundra Febrianto, Sutrisno Suro Mardjan, Andreas, and Isao Kobayashi. Emulsion stability of clove oil in chitosan and sodium alginate matrix. *International Journal of Food Properties*, 21:566–581, 1 2018.

[77] Mariia Kokina, Mark Shamtsyan, Cecilia Georgescu, Monica Mironescu, and Viktor Nedovic. Essential oil/alginate microcapsules; obtaining and applying. *Immunopathologia Persa*, 5, 1 2019.

[78] Rubén O. Bustos C, Francesca V. Alberti R, and Silvia B. Matiacevich. Edible antimicrobial films based on microencapsulated lemongrass oil. *Journal of Food Science and Technology*, 53:832–839, 1 2016.

[79] Cheng Yin, Chongxing Huang, Jun Wang, Ying Liu, Peng Lu, and Lijie Huang. Effect of chitosan- and alginate-based coatings enriched with cinnamon essential oil microcapsules to improve the postharvest quality of mangoes. *Materials 2019, Vol. 12, Page 2039*, 12:2039, 6 2019.

[80] Szymon Mania, Robert Tylingo, and Anna Michalowska. The drop-in-drop encapsulation in chitosan and sodium alginate as a method of prolonging the quality of linseed oil. *Polymers 2018, Vol. 10, Page 1355*, 10:1355, 12 2018.

[81] Jasdeep Singh, Kamaljit Kaur, and Parminder Kumar. Optimizing microencapsulation of -tocopherol with pectin and sodium alginate. *Journal of Food Science and Technology*, 55:3625–3631, 9 2018.

[82] Kairam Narsaiah, Shyam N. Jha, Robin A. Wilson, Harshad M. Mandge, and Musuvadi R. Manikantan. Optimizing microencapsulation of nisin with sodium alginate and guar gum. *Journal of Food Science and Technology*, 51:4054–4059, 12 2014.

[83] Andrea C.K. Bierhalz, Mariana A. Da Silva, Hermínio C. De Sousa, Mara E.M. Braga, and Theo G. Kieckbusch. Influence of natamycin loading methods on the physical characteristics of alginate active films. *The Journal of Supercritical Fluids*, 76:74–82, 4 2013.

[84] Hoda S. El-Sayed, Heba H. Salama, Kawther El-Shafei, and Nesrine M. Hegazi. Micro-encapsulation of eugenia supra-auxillaris phenolics rich fraction for its possible use as a natural food preservative. *Egyptian Journal of Chemistry*, 61:85–91, 2 2018.

[85] Siti Noraida Abd Rahim, Alawi Sulaiman, Fazlena Hamzah, Ku Halim Ku Hamid, Miradatul Najwa Muhd Rodhi, Mohibbah Musa, and Nurul Aini Edama. Enzymes encapsulation within calcium alginate-clay beads: Characterization and application for cassava slurry saccharification. *Procedia Engineering*, 68:411–417, 1 2013.

[86] Steva Levi, Vladislav Rac, Verica Manojlovi, Vesna Raki, Branko Bugarski, Teresa Flock, Katarzyna Ewa Krzyczmonik, and Viktor Nedovi. Limonene encapsulation in alginate/poly (vinyl alcohol). *Procedia Food Science*, 1:1816–1820, 1 2011.

[87] Henriette M.C. Azeredo, Kelvi W.E. Miranda, Hálisson L. Ribeiro, Morsyleide F. Rosa, and Diego M. Nascimento. Nanoreinforced alginate–acerola puree coatings on acerola fruits. *Journal of Food Engineering*, 113:505–510, 12 2012.

[88] Natalia A. Ballesteros, Marta Alonso, Sylvia Rodríguez Saint-Jean, and Sara I. Perez-Prieto. An oral dna vaccine against infectious haematopoietic necrosis virus (ihnv) encapsulated in alginate microspheres induces dose-dependent immune responses and significant protection in rainbow trout (oncorrhynchus mykiss). *Fish Shellfish Immunology*, 45:877–888, 8 2015.

[89] Sajjad Mohammadi, Seeram Ramakrishna, Sophie Laurent, Mohammad Ali Shokrgozar, Dariush Semnani, Davoud Sadeghi, Shahin Bonakdar, and Mohsen

Akbari. Fabrication of nanofibrous pva/alginate-sulfate substrates for growth factor delivery. *Journal of Biomedical Materials Research - Part A*, 107:403–413, 2 2019.

[90] Qian Li, Chen Guang Liu, Zhen Hua Huang, and Fang Fang Xue. Preparation and characterization of nanoparticles based on hydrophobic alginate derivative as carriers for sustained release of vitamin d3. *Journal of Agricultural and Food Chemistry*, 59:1962–1967, 3 2011.

[91] J ; Xu, Z ; Jiang, F ; Hou, K ; Zhu, C ; Xu, C ; Wang, H Wang, Jiquan Zhang, Yuying Sun, Jiamin Xu, Zhenlin Jiang, Fang Hou, Keyu Zhu, Chenxue Xu, Chaosheng Wang, and Huaping Wang. Preparation and mechanism of bio-based sodium alginate fibers with flame retardant and antibacterial properties. *Polymers 2023, Vol. 15, Page 154*, 15:154, 12 2022.

[92] Alina Hadi, Anjum Nawab, Feroz Alam, and Kishwar Zehra. Sustainable alginate/aloe vera composite biodegradable films reinforced with carboxymethyl cellulose and hydroxypropyl methylcellulose. *Polymer Composites*, 43:3471–3480, 6 2022.

[93] You Luo, Haiqing Liu, Shanzhong Yang, Jiarui Zeng, and Zhenqiang Wu. Sodium alginate-based green packaging films functionalized by guava leaf extracts and their bioactivities. *Materials 2019, Vol. 12, Page 2923*, 12:2923, 9 2019.

[94] Hyeongjin Lee, Yongbok Kim, Suhon Kim, and Geunhyung Kim. Mineralized biomimetic collagen/alginate/silica composite scaffolds fabricated by a low-temperature bio-plotting process for hard tissue regeneration: fabrication, characterisation and in vitro cellular activities. *Journal of Materials Chemistry B*, 2:5785–5798, 8 2014.

[95] Barbara Jadach, Weronika Świetlik, and Anna Froelich. Sodium alginate as a pharmaceutical excipient: Novel applications of a well-known polymer. *Journal of Pharmaceutical Sciences*, 111:1250–1261, 5 2022.

[96] Surendrasingh Y. Sonaye, Elif G. Ertugral, Chandrasekhar R. Kothapalli, and Prabaha Sikder. Extrusion 3d (bio)printing of alginate-gelatin-based composite scaffolds for skeletal muscle tissue engineering. *Materials 2022, Vol. 15, Page 7945*, 15:7945, 11 2022.

[97] Raha Ahmad Raus, Wan Mohd Fazli Wan Nawawi, and Ricca Rahman Nasaruddin. Alginate and alginate composites for biomedical applications. *Asian Journal of Pharmaceutical Sciences*, 16:280–306, 5 2021.

[98] A Ngel Serrano-Aroca, María Ferrandis-Montesinos, and Ruibing Wang. Antiviral properties of alginate-based biomaterials: Promising antiviral agents against sars-cov-2. 2012.

[99] Aida Syafiqah Abdul Rahman, Ahmad Noor Syimir Fizal, Nor Afifah Khalil, Ahmad Naim Ahmad Yahaya, Md Sohrab Hossain, and Muzafar Zulkifli. Fabrication and characterization of magnetic cellulose–chitosan–alginate composite hydrogel bead bio-sorbent. *Polymers*, 15:2494, 6 2023.

[100] Tao Chen, Haochen Liu, Jie Gao, Guowen Hu, Yuan Zhao, Xiuqin Tang, and Xiaobing Han. Efficient removal of methylene blue by bio-based sodium alginate/lignin composite hydrogel beads. *Polymers 2022, Vol. 14, Page 2917*, 14:2917, 7 2022.

[101] Claudia Heuer, Nora Brachhold, and Christos G. Aneziris. Evaluation of critical parameters for the fabrication of metal hollow spheres using an alginate-based sol–gel process. *Advanced Engineering Materials*, 26:2301145, 2 2024.

[102] Alejandro Hurtado, Alaa A.A. Aljabali, Vijay Mishra, Murtaza M. Tambuwala, and Ángel Serrano-Aroca. Alginate: Enhancement strategies for advanced applications. *International Journal of Molecular Sciences 2022, Vol. 23, Page 4486*, 23:4486, 4 2022.

[103] Ismael Santana, Manuel Felix, and Carlos Bengoechea. Seaweed as basis of ecosustainable plastic materials: Focus on alginate. *Polymers 2024, Vol. 16, Page 1662*, 16:1662, 6 2024.

[104] Zahra Eslami, Saïd Elkoun, Mathieu Robert, and Kokou Adjallé. A review of the effect of plasticizers on the physical and mechanical properties of alginate-based films. *Molecules 2023, Vol. 28, Page 6637*, 28:6637, 9 2023.

[105] Tezar Ramdhan, Su Hung Ching, Sangeeta Prakash, and Bhesh Bhandari. Physical and mechanical properties of alginate based composite gels. *Trends in Food Science and Technology*, 106:150–159, 12 2020.

[106] Leticia Montes, Mauro Gisbert, and Ramón Moreira. Water sorption isotherms of different sodium alginates: Thermodynamic evaluation and influence of mannuronate-guluronate copolymers. *Journal of Food Processing and Preservation*, 46:e17179, 12 2022.

[107] Jinhong Yang, Na Sun, Xuelai Xie, Zhangyu Feng, Na Liu, Kai Wang, and Min Lin. Structure characterization and mechanical properties of acidity-induced helix of alginate and fibers. *Materials*, 18:2619, 6 2025.

[108] An Yumin, Dong Liguo, Yang Yi, and Jia Yongna. Mechanical properties of an interpenetrating network poly(vinyl alcohol)/alginate hydrogel with hierarchical fibrous structures. *RSC Advances*, 12:11632–11639, 4 2022.

[109] Marco Mancini, Mauro Moresi, and Roberto Rancini. Mechanical properties of alginate gels: empirical characterisation. *Journal of Food Engineering*, 39:369–378, 1999.

[110] Ji Sheng Yang, Ying Jian Xie, and Wen He. Research progress on chemical modification of alginate: A review. *Carbohydrate Polymers*, 84:33–39, 2 2011.

[111] P. Laurienzo, M. Malinconico, A. Motta, and A. Vicinanza. Synthesis and characterization of a novel alginate–poly(ethylene glycol) graft copolymer. *Carbohydrate Polymers*, 62:274–282, 12 2005.

[112] S. Safi, M. Morshed, S. A. Hosseini Ravandi, and M. Ghiaci. Study of electrospinning of sodium alginate, blended solutions of sodium alginate/poly(viny1

alcoho1) and sodium alginate/poly(ethylene oxide). *Journal of Applied Polymer Science*, 104:3245–3255, 6 2007.

[113] Ramesh M. Gohil. Synergistic blends of natural polymers, pectin and sodium alginate. *Journal of Applied Polymer Science*, 120:2324–2336, 5 2011.

[114] Renjini M. Nair, B. Bindhu, and Reena V L. A polymer blend from gum arabic and sodium alginate - preparation and characterization. *Journal of Polymer Research*, 27:1–7, 6 2020.

[115] Carl D. Saquing, Christina Tang, Brinda Monian, Christopher A. Bonino, Joshua L. Manasco, Eben Alsberg, and Saad A. Khan. Alginate-polyethylene oxide blend nanofibers and the role of the carrier polymer in electrospinning. *Industrial and Engineering Chemistry Research*, 52:8692–8704, 7 2013.

[116] Siddaramaiah, T. M.Mruthyunjaya Swamy, B. Ramaraj, and Joong Hee Lee. Sodium alginate and its blends with starch: Thermal and morphological properties. *Journal of Applied Polymer Science*, 109:4075–4081, 9 2008.

[117] Guadalupe I. Olivas and Gustavo V. Barbosa-Cánovas. Alginate–calcium films: Water vapor permeability and mechanical properties as affected by plasticizer and relative humidity. *LWT - Food Science and Technology*, 41:359–366, 3 2008.

[118] Heather M. Kenny, Clare M. Reynolds, Marco Garcia-Vaquero, and Emma L. Feeney. Keeping an eye on alginate: innovations and opportunities for sustainable production and diverse applications. *Carbohydrate Polymers*, 366:123902, 10 2025.

[119] Viruja Ummat, Saravana Periaswamy Sivagnanam, Saranya Rameshkumar, Mukesh Pednekar, Stephen Fitzpatrick, Dilip K. Rai, Ramesh Babu Padamati, Colm O'Donnell, and Brijesh Kumar Tiwari. Sequential extraction of fucoidan, laminarin, mannitol, alginate and protein from brown macroalgae ascophyllum nodosum and fucus vesiculosus. *International Journal of Biological Macromolecules*, 256:128195, 1 2024.

[120] Magdalena Beata Łabowska, Izabela Michalak, and Jerzy Detyna. Methods of extraction, physicochemical properties of alginates and their applications in biomedical field - a review. *Open Chemistry*, 17:738–762, 1 2019.

[121] Si Yuan Lu, Tao Zhou, Iqra Shabbir, Jaehwan Choi, Young Heui Kim, Myeongsam Park, Jude Juventus Aweya, Karsoon Tan, Saiyi Zhong, and Kit Leong Cheong. Marine algal polysaccharides: Multifunctional bioactive ingredients for cosmetic formulations. *Carbohydrate Polymers*, 353:123276, 4 2025.

[122] Aleksandra Nesic, Maria Valeria De Bonis, Giovanni Dal Poggetto, Gianpaolo Ruocco, and Gabriella Santagata. Microwave assisted extraction of raw alginate as a sustainable and cost-effective method to treat beach-accumulated sargassum algae. *Polymers 2023, Vol. 15, Page 2979*, 15:2979, 7 2023.

[123] J. Queffelec, N. Flórez-Fernández, H. Domínguez, and M. D. Torres. Microwave hydrothermal processing of undaria pinnatifida for bioactive peptides. *Bioresource Technology*, 342:125882, 12 2021.

[124] Ana Malvis Romero, Felix Brozio, Sinah Kammler, Christin Burkhardt, Leon Baruth, Martin Kaltschmitt, Garabed Antranikian, and Andreas Liese. Enzyme-assisted extraction of alginate from beach wrack fucus vesiculosus. *Chemie-Ingenieur-Technik*, 95:549–556, 4 2023.

[125] Sinemobong O. Essien, Brent Young, and Saeid Baroutian. Recent advances in subcritical water and supercritical carbon dioxide extraction of bioactive compounds from plant materials. *Trends in Food Science Technology*, 97:156–169, 3 2020.

[126] Md Sadek Ali, Monjurul Haq, Sin Won Park, Ji Min Han, Jang Woo Kim, Min Seo Choi, Sang Min Lee, Jin Seok Park, Man Seog Chun, Hee Jeong Lee, and Byung Soo Chun. Recent advances in recovering bioactive compounds from macroalgae and microalgae using subcritical water extraction: Prospective compounds and biological activities. *Food Chemistry*, 469:142602, 3 2025.

[127] Thamani Freedom Gondo, Madeleine Jönsson, Eva Nordberg Karlsson, Margareta Sandahl, and Charlotta Turner. Extractability, selectivity, and comprehensiveness in supercritical fluid extraction of seaweed using ternary mixtures of carbon dioxide, ethanol, and water. *Journal of Chromatography A*, 1706, 9 2023.

[128] Wimar Reynaga-Navarro, Lucas Bozzo, René H. Wijffels, Michel Eppink, and Antoinette Kazbar. Characterisation of alginate extracted from saccharina latissima with deep eutectic solvents. *Food Hydrocolloids*, 171:111862, 3 2026.

[129] Olav Andreas Aarstad, Anne Tøndervik, Håvard Sletta, and Gudmund Skjåk-Bræk. Alginate sequencing: An analysis of block distribution in alginates using specific alginate degrading enzymes. *Biomacromolecules*, 13:106–116, 1 2012.

[130] Shao Fu, Ankur Thacker, Diana M. Sperger, Riccardo L. Boni, Sachin Velankar, Eric J. Munson, and Lawrence H. Block. Rheological evaluation of inter-grade and inter-batch variability of sodium alginate. *AAPS PharmSciTech*, 11:1662–1674, 11 2010.

[131] Fengwei Xie, Chengcheng Gao, and Luc Avérous. Alginate-based materials: Enhancing properties through multiphase formulation design and processing innovation. *Materials Science and Engineering: R: Reports*, 159:100799, 6 2024.

[132] Shabnam Sabetkish, Peter Currie, and Laurence Meagher. Recent trends in 3d bioprinting technology for skeletal muscle regeneration. *Acta Biomaterialia*, 181:46–66, 6 2024.

[133] Chen Chen, Xingfei Li, Cheng Lu, Xing Zhou, Long Chen, Chao Qiu, Zhengyu Jin, and Jie Long. Advances in alginate lyases and the potential application of enzymatic prepared alginate oligosaccharides: A mini review. *International Journal of Biological Macromolecules*, 260:129506, 3 2024.

[134] Meiling Song, Lin Chen, Chen Dong, Minghui Tang, Yuan Wei, Depeng Lv, Quancai Li, and Zhen Chen. Alginate oligosaccharide and gut microbiota: Exploring the key to health. *Nutrients*, 17:1977, 6 2025.

[135] Susan K. Tam, Julie Dusseault, Stefania Polizu, Martin Ménard, Jean Pierre Hallé, and L'Hocine Yahia. Impact of residual contamination on the biofunctional properties of purified alginates used for cell encapsulation. *Biomaterials*, 27:1296–1305, 3 2006.

[136] Martina Urbanova, Miroslava Pavelkova, Jiri Czernek, Katerina Kubova, Jakub Vyslouzil, Alena Pechova, Dobromila Molinkova, Jan Vyslouzil, David Vetchy, and Jiri Brus. Interaction pathways and structure-chemical transformations of alginate gels in physiological environments. *Biomacromolecules*, 20:4158–4170, 11 2019.

[137] Enrique Mancha Sánchez, J. Carlos Gómez-Blanco, Esther López Nieto, Javier G. Casado, Antonio Macías-García, María A. Díaz Díez, Juan Pablo Carrasco-Amador, Diego Torrejón Martín, Francisco Miguel Sánchez-Margallo, and J. Blas Pagador. Hydrogels for bioprinting: A systematic review of hydrogels synthesis, bioprinting parameters, and bioprinted structures behavior. *Frontiers in Bioengineering and Biotechnology*, 8:776, 8 2020.

[138] Juhi Chakraborty, Nilotpal Majumder, Aarushi Sharma, Sukanya Prasad, and Sourabh Ghosh. 3d bioprinted silk-reinforced alginate-gellan gum constructs for cartilage regeneration. *Bioprinting*, 28:e00232, 12 2022.

[139] Liangbin Li, Yabeng Fang, Rob Vreeker, Ingrid Appelqvist, and Eduardo Mendes. Reexamining the egg-box model in calcium - alginate gels with x-ray diffraction. *Biomacromolecules*, 8:464–468, 2 2007.

[140] Yi Wang, Yiguo Zhao, Jun He, Cuixia Sun, Wei Lu, Yin Zhang, and Yapeng Fang. Doubling growth of egg-box structure during calcium-mediated molecular assembly of alginate. *Journal of Colloid and Interface Science*, 634:747–756, 3 2023.

[141] I. Braccini and S. Pérez. Molecular basis of ca2+-induced gelation in alginates and pectins: The egg-box model revisited. *Biomacromolecules*, 2:1089–1096, 2001.

[142] Pawel Sikorski, Frode Mo, Gudmund Skjåk-Bræk, and Bjørn T. Stokke. Evidence for egg-box-compatible interactions in calcium - alginate gels from fiber x-ray diffraction. *Biomacromolecules*, 8:2098–2103, 7 2007.

[143] Anastasiya O. Makarova, Svetlana R. Derkach, Tahar Khair, Mariia A. Kazantseva, Yuriy F. Zuev, and Olga S. Zueva. Ion-induced polysaccharide gelation: Peculiarities of alginate egg-box association with different divalent cations. *Polymers*, 15, 3 2023.

[144] Wojciech Plazinski and Anita Plazinska. Molecular dynamics study of the interactions between phenolic compounds and alginate/alginic acid chains. *New Journal of Chemistry*, 35:1607–1614, 2011.

[145] Lianqi Cao, Wei Lu, Analucia Mata, Katsuyoshi Nishinari, and Yapeng Fang. Egg-box model-based gelation of alginate and pectin: A review, 8 2020.

[146] Zezhong John Li, Simcha Srebnik, and Orlando J. Rojas. Competing effects of hydration and cation complexation in single-chain alginate. *Biomacromolecules*, 2021.

[147] Isabelle Braccini, Robert P Grasso, and Serge Pérez. Conformational and configurational features of acidic polysaccharides and their interactions with calcium ions: a molecular modeling investigation. *Carbohydrate Research*, 317:119–130, 1999.

[148] Hadas Hecht and Simcha Srebnik. Structural characterization of sodium alginate and calcium alginate. *Biomacromolecules*, 17:2160–2167, 6 2016.

[149] Hadas Hecht and Simcha Srebnik. Sequence-dependent association of alginate with sodium and calcium counterions. *Carbohydrate Polymers*, 157:1144–1152, 2 2017.

[150] Lovorka Peric, Cristina S. Pereira, Serge Perez, and Philippe H. Hunenberger. Conformation, dynamics and ion-binding properties of single-chain polyuronates: A molecular dynamics study. *Molecular Simulation*, 34:421–446, 4 2008.

[151] Lovorka Perić-Hassler and Philippe H. Hünenberger. Interaction of alginate single-chain polyguluronate segments with mono- and divalent metal cations: A comparative molecular dynamics study. *Molecular Simulation*, 36:778–795, 9 2010.

[152] Zezhong John Li, Simcha Srebnik, and Orlando J. Rojas. Revisiting cation complexation and hydrogen bonding of single-chain polyguluronate alginate. *Biomacromolecules*, 22:4027–4036, 9 2021.

[153] Aidan P. Thompson, H. Metin Aktulga, Richard Berger, Dan S. Bolintineanu, W. Michael Brown, Paul S. Crozier, Pieter J. in 't Veld, Axel Kohlmeyer, Stan G. Moore, Trung Dac Nguyen, Ray Shan, Mark J. Stevens, Julien Tranchida, Christian Trott, and Steven J. Plimpton. Lammps - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Computer Physics Communications*, 271, 2 2022.

[154] David Van Der Spoel, Erik Lindahl, Berk Hess, Gerrit Groenhof, Alan E. Mark, and Herman J.C. Berendsen. Gromacs: Fast, flexible, and free, 12 2005.

[155] Loup Verlet. Computer "experiments" on classical fluids. i. thermodynamical properties of lennard-jones molecules. *Physical Review*, 159:98, 7 1967.

[156] Michael P. Allen and Dominic J. Tildesley. *Computer Simulation of Liquids: Second Edition*. Oxford University Press, 11 2017.

[157] R.W. Hockney. The potenitial calculation and some applications. *Methods Comput. Phys.*, 20:135–, 1970.

[158] William C. Swope, Hans C. Andersen, Peter H. Berens, and Kent R. Wilson. A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: Application to small water clusters. *The Journal of Chemical Physics*, 76:637–649, 1 1982.

[159] Stacy L Chin, Qing Lu, Eric L Dane, Laura Dominguez, Christopher J Mcknight, John E Straub, and Mark W Grinstaff. Combined molecular dynamics simula-

tions and experimental studies of the structure and dynamics of poly-amido-saccharides. 2016.

[160] Poornima Ramamohan, István Furó, and Jakob Wohlert. Timescales for convergence in all-atom molecular dynamics simulations of hydrated amorphous xylan. *Carbohydrate Polymers*, 286:119263, 6 2022.

[161] Thomas E Gartner and Arthi Jayaraman. Modeling and simulations of polymers: A roadmap. 2019.

[162] David A. Case, Thomas E. Cheatham III, Carlos Simmerling, Adrian Roitberg, Kenneth M. Merz, Ross C. Walker, Ray Luo, Pengfei Li, Tom Darden, Celeste Sagui, Feng Pan, Junmei Wang, Daniel R. Roe, Jason Swails, Andreas W. Götz, Jamie Smith, David Cerutti, Taisung Lee, Darrin York, Timothy Giese, Tyler Luchko, Negin Forouzesh, Vinícius Wilian D. Cruzeiro, Gérald Monard, Yinglong Miao, Jinan Wang, Charles Lin, G. Andrés Cisneros, Ali Rahnamoun, Zhen Li, Akhil Shajan, Madushanka Manathunga, Joshua T. Berryman, István Kolossváry, Nikolai R. Skrynnikov, Oleg Mikhailovskii, Yi Xue, Sergei A. Izmailov, Koushik Kasavajhala, Kellon Belfon, Jana Shen, Julie Harris, Alexey Onufriev, Saeed Izadi, Xiongwu Wu, Holger Gohlke, Stephan Schott-Verdugo, Yongxian Wu, Qiang Zhu, Zhen Huang, Cizhang Zhao, George Giambaşu, Jian Liu, Hai Nguyen, Scott R. Brozell, Andriy Kovalenko, Mike Gilson, Ido Ben-Shalom, Tom Kurtzman, Sergio Pantano, Matias Machado, H. Metin Aktulga, Mehmet Cagri Kaymak, Kurt A. O'Hearn, and Peter A. Kollman. Amber 2024, 2024.

[163] Jan Norberg and Lennart Nilsson. On the truncation of long-range electrostatic interactions in dna. *Biophysical Journal*, 79:1537, 2000.

[164] Tom Darden, Darrin York, and Lee Pedersen. Particle mesh ewald: An nlog(n) method for ewald sums in large systemscut. *The Journal of Chemical Physics*, 98:5648, 1993.

[165] Anu George, Sandip Mondal, Madhura Purnaprajna, and Prashanth Athri. Review of electrostatic force calculation methods and their acceleration in molecular dynamics packages using graphics processors. *ACS Omega*, 7:32877–32896, 9 2022.

[166] Jean Paul Ryckaert, Giovanni Ciccotti, and Herman J.C. Berendsen. Numerical integration of the cartesian equations of motion of a system with constraints: molecular dynamics of n-alkanes. *Journal of Computational Physics*, 23:327–341, 3 1977.

[167] Berk Hess, Henk Bekker, Herman J.C. Berendsen, and Johannes G.E.M. Fraaije. Lincs: A linear constraint solver for molecular simulations. *Journal of Computational Chemistry*, 18:1463–1472, 1997.

[168] Shuichi Miyamoto and Peter A. Kollman. Settle: An analytical version of the shake and rattle algorithm for rigid water models. *Journal of Computational Chemistry*, 13:952–962, 10 1992.

[169] Karl N. Kirschner, Austin B. Yongye, Sarah M. Tschampel, Jorge González-Outeiriño, Charlisa R. Daniels, B. Lachele Foley, and Robert J. Woods. Glycam06: A generalizable biomolecular force field. carbohydrates. *Journal of Computational Chemistry*, 29:622–655, 3 2008.

[170] Maria M. Pérez-Madrigal, Joan Torras, Jordi Casanovas, Marleen Häring, Carlos Alemán, and David Díaz Díaz. Paradigm shift for preparing versatile m2+-free gels from unmodified sodium alginate. *Biomacromolecules*, 18:2967–2979, 9 2017.

[171] Margrethe Gaardløs, Anders Lervik, and Sergey A. Samsonov. Computational modeling of the molecular basis for the calcium-dependence of the mannuronan c-5 epimerase avalge6 from azotobacter vinelandii. *Computational and Structural Biotechnology Journal*, 21:2188–2196, 1 2023.

[172] Bin Zhang, Xue er Qi, Jun long Mao, and Xiao guo Ying. Trehalose and alginate oligosaccharides affect the stability of myosin in whiteleg shrimp (litopenaeus

vannamei): The water-replacement mechanism confirmed by molecular dynamic simulation. *LWT*, 127, 6 2020.

[173] B. R. Brooks, C. L. Brooks, A. D. Mackerell, L. Nilsson, R. J. Petrella, B. Roux, Y. Won, G. Archontis, C. Bartels, S. Boresch, A. Caflisch, L. Caves, Q. Cui, A. R. Dinner, M. Feig, S. Fischer, J. Gao, M. Hodoscek, W. Im, K. Kuczera, T. Lazaridis, J. Ma, V. Ovchinnikov, E. Paci, R. W. Pastor, C. B. Post, J. Z. Pu, M. Schaefer, B. Tidor, R. M. Venable, H. L. Woodcock, X. Wu, W. Yang, D. M. York, and M. Karplus. Charmm: The biomolecular simulation program. *Journal of Computational Chemistry*, 30:1545–1614, 7 2009.

[174] Agnieszka Brzyska and Wojciech Płaziński. Modeling conformational changes in alginic acid oligomers induced by external forces. *Carbohydrate Research*, 545, 11 2024.

[175] Matthew B. Stewart, Darli T. Myat, Michael Kuiper, Richard J. Manning, Stephen R. Gray, and John D. Orbell. A structural basis for the amphiphilic character of alginates – implications for membrane fouling. *Carbohydrate Polymers*, 164:162–169, 5 2017.

[176] William L. Jorgensen, David S. Maxwell, and Julian Tirado-Rives. Development and testing of the opls all-atom force field on conformational energetics and properties of organic liquids. *Journal of the American Chemical Society*, 118:11225–11236, 11 1996.

[177] Yuan Xiang, Rong Guang Xu, and Yongsheng Leng. How alginate monomers contribute to organic fouling on polyamide membrane surfaces? *Journal of Membrane Science*, 643, 3 2022.

[178] Pengfei Li and Kenneth M. Merz. Taking into account the ion-induced dipole interaction in the nonbonded model of ions. *Journal of Chemical Theory and Computation*, 10:289–297, 1 2014.

[179] Pengfei Li, Lin Frank Song, and Kenneth M. Merz. Systematic parameterization

of monovalent ions employing the nonbonded model. *Journal of Chemical Theory and Computation*, 11:1645–1657, 4 2015.

[180] Anna Wolnik, Luca Albertin, Landry Charlier, and Karim Mazeau. Probing the helical forms of ca2+-guluronan junction zones in alginate gels by molecular dynamics 1: Duplexes. *Biopolymers*, 99:562–571, 8 2013.

[181] Thomas D. IV Perry, Randall T. Cygan, and Ralph Mitchell. Molecular models of alginic acid: Interactions with calcium ions and calcite surfaces. *Geochimica et Cosmochimica Acta*, 70:3508–3532, 7 2006.

[182] Avery A. Agles and Ian C. Bourg. Structure-thermodynamic relationship of a polysaccharide gel (alginate) as a function of water content and counterion type (na vs ca). *Journal of Physical Chemistry B*, 127:1828–1841, 3 2023.

[183] William L Jorgensen, Jayaraman Chandrasekhar, and Jeffry D Madura. Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys*, 79:926–935, 1983.

[184] H. J.C. Berendsen, J. P.M. Postma, W. F. Van Gunsteren, A. Dinola, and J. R. Haak. Molecular dynamics with coupling to an external bath. *The Journal of Chemical Physics*, 81:3684–3690, 10 1984.

[185] Giovanni Bussi, Davide Donadio, and Michele Parrinello. Canonical sampling through velocity rescaling. *The Journal of Chemical Physics*, 126, 1 2007.

[186] R Metzler and J Klafter. The random walk:s guide to anomalous diffusion: A fractional dynamics approach the random walk's guide to anomalous di!usion: a fractional dynamics approach, 2000.

[187] David A. Case, Hasan Metin Aktulga, Kellon Belfon, David S. Cerutti, G. Andrés Cisneros, Vinícius Wilian D. Cruzeiro, Negin Forouzesh, Timothy J. Giese, Andreas W. Götz, Holger Gohlke, Saeed Izadi, Koushik Kasavajhala, Mehmet C. Kaymak, Edward King, Tom Kurtzman, Tai Sung Lee, Pengfei Li, Jian Liu, Tyler Luchko, Ray Luo, Madushanka Manathunga, Matias R. Machado, Hai Minh

Nguyen, Kurt A. O'Hearn, Alexey V. Onufriev, Feng Pan, Sergio Pantano, Ruxi Qi, Ali Rahnamoun, Ali Risheh, Stephan Schott-Verdugo, Akhil Shajan, Jason Swails, Junmei Wang, Haixin Wei, Xiongwu Wu, Yongxian Wu, Shi Zhang, Shiji Zhao, Qiang Zhu, Thomas E. Cheatham, Daniel R. Roe, Adrian Roitberg, Carlos Simmerling, Darrin M. York, Maria C. Nagan, and Kenneth M. Merz. Ambertools. *Journal of Chemical Information and Modeling*, 63:6183–6191, 10 2023.

[188] James C. Phillips, David J. Hardy, Julio D.C. Maia, John E. Stone, João V. Ribeiro, Rafael C. Bernardi, Ronak Buch, Giacomo Fiorin, Jérôme Hénin, Wei Jiang, Ryan McGreevy, Marcelo C.R. Melo, Brian K. Radak, Robert D. Skeel, Abhishek Singharoy, Yi Wang, Benoît Roux, Aleksei Aksimentiev, Zaida Luthey-Schulten, Laxmikant V. Kalé, Klaus Schulten, Christophe Chipot, and Emad Tajkhorshid. Scalable molecular dynamics on cpu and gpu architectures with namd. *Journal of Chemical Physics*, 153, 7 2020.

[189] Mark Abraham, Andrey Alekseenko, Brian Andrews, Vladimir Basov, Paul Bauer, Hugh Bird, Eliane Briand, Ania Brown, Mahesh Doijade, Giacomo Fiorin, Stefan Fleischmann, Sergey Gorelov, Gilles Gouaillardet, Alan Gray, M. Eric Irrgang, Farzaneh Jalalypour, Petter Johansson, Carsten Kutzner, Grzegorz Łazarski, Justin A. Lemkul, Magnus Lundborg, Pascal Merz, Vedran Miletić, Dmitry Morozov, Lukas Müllender, Julien Nabet, Szilárd Páll, Andrea Pasquadibisceglie, Michele Pellegrino, Nicola Piasentin, Daniele Rapetti, Muhammad Umair Sadiq, Hubert Santuz, Roland Schulz, Michael Shirts, Tatiana Shugaeva, Alexey Shvetsov, Philip Turner, Alessandra Villa, Sebastian Wingbermühle, Berk Hess, and Erik Lindahl. Gromacs documentation release 2025.2, 3 2025.

[190] Martin Karplus and J. Andrew McCammon. Molecular dynamics simulations of biomolecules. *Nature Structural Biology*, 9:646–652, 9 2002.

[191] Christoph Niethammer, Stefan Becker, Martin Bernreuther, Martin Buchholz, Wolfgang Eckhardt, Alexander Heinecke, Stephan Werth, Hans Joachim Bungartz, Colin W. Glass, Hans Hasse, Jadran Vrabec, and Martin Horsch. ls1 mar-

dyn: The massively parallel molecular dynamics code for large systems. *Journal of Chemical Theory and Computation*, 10:4455–4464, 10 2014.

[192] Stefano Piana, John L. Klepeis, and David E. Shaw. Assessing the accuracy of physical models used in protein-folding simulations: quantitative evidence from long molecular dynamics simulations. *Current Opinion in Structural Biology*, 24:98–105, 2 2014.

[193] D. C. Montgomery. Design and analysis of experiments, john wiley sons. *Mycological Research*, 106:1323–1330, 2017.

[194] Benjamin Durakovic. Design of experiments application, concepts, examples: State of the art. *Periodicals of Engineering and Natural Sciences*, 5:421–439, 12 2017.

[195] Minitab, LLC. Minitab.

[196] Anthony C. Atkinson and David C. Woods. Designs for generalized linear models. 10 2015.

[197] Araz Jakalian, Bruce L. Bush, David B. Jack, and Christopher I. Bayly. Fast, efficient generation of high-quality atomic charges. am1-bcc model: I. method. *Journal of Computational Chemistry*, 21:132–146, 1 2000.

[198] Araz Jakalian, David B. Jack, and Christopher I. Bayly. Fast, efficient generation of high-quality atomic charges. am1-bcc model: Ii. parameterization and validation. *Journal of Computational Chemistry*, 23:1623–1641, 12 2002.

[199] Christopher I Bayly, Piotr Cieplak, Wendy D Cornell, and Peter A Kollman. A well-behaved electrostatic potential based method using charge restraints for deriving atomic charges: The resp model, 1993.

[200] Attila Szabo and Neil S. Ostlund. *Modern quantum chemistry: introduction to advanced electronic structure theory*. Dover Publications, 1996.

[201] Levine and Ira N. *Pearson Advanced Chemistry Series*. 7 edition, 2014.

[202] Research Collaboratory for Structural Bioinformatics. About rcsb pdb: Enabling breakthroughs in scientific and biomedical research and education.

[203] Keke Zhang, Tao Liu, Weizhi Liua, and Qianqian Lyua. Structural insights into the substrate-binding cleft of alyf reveal the first long-chain alginate-binding mode. *Acta Crystallographica Section D: Structural Biology*, 77:336–346, 3 2021.

[204] W Liu, Q. Lyu, and K Zhang. Pdb entry - 7bz0, 2021.

[205] F. Fredslund, D.H. Welner, and C. Wilkens. Pdb entry - 7ncz, 2022.

[206] William Humphrey, Andrew Dalke, and Klaus Schulten. Vmd: Visual molecular dynamics, 1996.

[207] Marcus D Hanwell, Donald E Curtis, David C Lonie, Tim Vandermeersch, Eva Zurek, and Geoffrey R Hutchison. Software open access avogadro: an advanced semantic chemical editor, visualization, and analysis platform, 2012.

[208] Pekka Mark and Lennart Nilsson. Structure and dynamics of the tip3p, spc, and spc/e water models at 298 k. *Journal of Physical Chemistry A*, 105:9954–9960, 11 2001.

[209] A. Martinsen, G. Skjåk-Bræk, and O. Smidsrød. Alginate as immobilization material: I. correlation between chemical and physical properties of alginate gel beads. *Biotechnology and Bioengineering*, 33:79–89, 1989.

[210] B. S. Roopa and Suvendu Bhattacharya. Alginate gels: I. characterization of textural attributes. *Journal of Food Engineering*, 85:123–131, 3 2008.

[211] Tezar Ramdhan, Su Hung Ching, Sangeeta Prakash, and Bhesh Bhandari. Time dependent gelling properties of cuboid alginate gels made by external gelation method: Effects of alginate-cacl2 solution ratios and ph. *Food Hydrocolloids*, 90:232–240, 5 2019.

[212] Jeanie L. Drury, Robert G. Dennis, and David J. Mooney. The tensile properties of alginate hydrogels. *Biomaterials*, 25:3187–3199, 2004.

[213] Chungeun Jeong, Seonghui Kim, Chanmin Lee, Suengmok Cho, and Seon Bong Kim. Changes in the physical properties of calcium alginate gel beads under a wide range of gelation temperature conditions. *Foods*, 9, 2020.

[214] Michelle A. LeRoux, Farshid Guilak, and Lori A. Setton. Compressive and shear properties of alginate gel: Effects of sodium ions and alginate concentration. *Journal of Biomedical Materials Research*, 47:46–53, 1999.

[215] H. Kakita and H. Kamishima. Some properties of alginate gels derived from algal sodium alginate. *Journal of Applied Phycology*, 20:543–549, 10 2008.

[216] Shao Fu, Ankur Thacker, Diana M. Sperger, Riccardo L. Boni, Ira S. Buckner, Sachin Velankar, Eric J. Munson, and Lawrence H. Block. Relevance of rheological properties of sodium alginate in solution to calcium alginate gel properties. *AAPS PharmSciTech*, 12:453–460, 2011.

[217] Elin Hermansson, Erich Schuster, Lars Lindgren, Annika Altskär, and Anna Ström. Impact of solvent quality on the network strength and structure of alginate gels. *Carbohydrate Polymers*, 144:289–296, 6 2016.

[218] Sijin Saji, Andrew Hebden, Parikshit Goswami, and Chenyu Du. A brief review on the development of alginate extraction process and its sustainability. *Sustainability*, 14:5181, 5 2022.

[219] Gustavo Hernández-Carmona, Dennis J. McHugh, Dora Luz Arvizu-Higuera, and Y. Elizabeth Rodríguez-Montesinos. Pilot plant scale extraction of alginates from macrocystis pyrifera 4. conversion of alginic acid to sodium alginate, drying and milling. *Journal of Applied Phycology*, 14:445–451, 12 2002.

[220] Hylenne Bojorges, Amparo López-Rubio, Antonio Martínez-Abad, and María José Fabra. Overview of alginate extraction processes: Impact on alginate molecular structure and techno-functional properties. *Trends in Food Science Technology*, 140:104142, 10 2023.

[221] Kazuaki Yamagiwa, Taro Kozawa, and Akira Ohkawa. Effects of alginate composition and gelling conditions on diffusional and mechanical properties of calcium-alginate gel beads. *Journal of Chemical Engineering of Japan*, 28:462–467, 1995.

[222] Toshihisa Yotsuyanagi, Tsuneo Ohkubo, Takafumi Ohhashi, and Ken Ikeda. Calcium-induced gelation of alginic acid and ph-sensitive reswelling of dried gels. *Chemical and Pharmaceutical Bulletin*, 35:1555–1563, 1986.

[223] Sabine Leick, Stefan Henning, Patrick Degen, Dieter Suter, and Heinz Rehage. Deformation of liquid-filled calcium alginate capsules in a spinning drop apparatus. *Physical Chemistry Chemical Physics*, 12:2950–2958, 2010.

[224] Szilárd Páll, Artem Zhmurov, Paul Bauer, Mark Abraham, Magnus Lundborg, Alan Gray, Berk Hess, and Erik Lindahl. Heterogeneous parallelization and acceleration of molecular dynamics simulations in gromacs. *Journal of Chemical Physics*, 153, 9 2020.

[225] Nikolay Kondratyuk, Vsevolod Nikolskiy, Daniil Pavlov, and Vladimir Stegailov. Gpu-accelerated molecular dynamics: State-of-art software performance and porting from nvidia cuda to amd hip. *The International Journal of High Performance Computing Applications*, 35:312–324, 7 2021.

[226] Stefano Piana, Kresten Lindorff-Larsen, Robert M. Dirks, John K. Salmon, Ron O. Dror, and David E. Shaw. Evaluating the effects of cutoffs and treatment of long-range electrostatics in protein folding simulations. *PLOS ONE*, 7:e39918, 6 2012.

[227] Xiaoliang Qian, Daniel Strahs, and Tamar Schlick. A new program for optimizing periodic boundary models of solvated biomolecules (pbcaid). *Journal of Computational Chemistry*, 22:1843–1850, 11 2001.

[228] M. Tuckerman, B. J. Berne, and G. J. Martyna. Reversible multiple time scale molecular dynamics. *The Journal of Chemical Physics*, 97:1990–2001, 8 1992.

[229] Wilfred F. van Gunsteren, Xavier Daura, and Alan E. Mark. Gromos force field. *Encyclopedia of Computational Chemistry*, 9 1998.

[230] Frank Eisenhaber, Philip Lijnzaad, Patrick Argos, Chris Sander, and Michael Scharf. The double cubic lattice method: Efficient approaches to numerical integration of surface area and volume and to dot surface contouring of molecular assemblies. *Journal of Computational Chemistry*, 16:273–284, 1995.

[231] Pierre Agulhon, Velina Markova, Mike Robitzer, Françoise Quignard, and Tzonka Mineva. Structure of alginate gels: Interaction of diuronate units with divalent cations from density functional calculations. *Biomacromolecules*, 13:1899–1907, 6 2012.

[232] Yuan Xiang, Yaolin Liu, Baoxia Mi, and Yongsheng Leng. Molecular dynamics simulations of polyamide membrane, calcium alginate gel, and their interactions in aqueous solution. *Langmuir*, 30:9098–9106, 8 2014.

[233] Matthew B. Stewart, Stephen R. Gray, Todor Vasiljevic, and John D. Orbell. Exploring the molecular basis for the metal-mediated assembly of alginate gels. *Carbohydrate Polymers*, 102:246–253, 2 2014.

[234] Karina Panczyk, Karolina Gaweda, Mateusz Drach, and Wojciech Plazinski. Extension of the gromos 56a6carbo/carbo$_r force field for charged, protonated, and esterified uronates. The Journal of Phy$
$3696 - -3710, 42018.

[235] Richard L. Anderson, H. Christopher Greenwell, James L. Suter, Peter V. Coveney, and Mary Ann Thyveetil. Determining materials properties of natural composites using molecular simulation. *Journal of Materials Chemistry*, 19:7251–7262, 10 2009.

Appendix A

GLYCAM-06j, ion, and TIP3P force field parameters

The GLYCAM-06j force field is the carbohydrate-specific AMBER force field and was paired with the AMBER-recommended ion parameters and TIP3P water model. Presented in the following tables and equations are the equations and parameters from these models which were applied to the simulations in this work. Table A.1 shows the atom types with which atoms they refer to. Tables A.2-A.4 show the parameters involved in bond stretching, angle bending and dihedral torsions. The non-bonded parameters are presented in Table A.5.

Table A.1: Naming, types, descriptions, and masses of atoms used in alginate simulations.

Atom Type	Atom Description	Alginate Atoms	Atom Mass
Cg	sp3 aliphatic carbon	Ring carbons	12.01
C	sp2 carbonyl group carbon	Carboxylate carbon	12.01
O2	hydroxyl group oxygen	OC1/2	16.00
Oh	ether oxygen	O1-O4	16.00
Os	carboxyl group oxygen	OR OG	16.00
Ho	hydroxyl group hydrogen	Hydroxyl hydrogens	1.008
H1	aliphatic hydrogen bonded to carbon with 1 electron withdrawal group	Hydrogens bonded to ring carbons	1.008
H2	aliphatic hydrogen bonded to carbon with 2 electron withdrawal groups	Hydrogens bonded to ring carbons	1.008
OW	water oxygen		15.994
HW	water hydrogen		1.008
Na^+	sodium ion		22.99
Ca^{2+}	calcium ion		40.078
Cl^-	chlorine ion		35.453

Table A.2: Atoms and parameters involved in bond stretching.

Atom Types Involved in Bond	l^0 (Å)	k_b (kcal/mol)
Cg-Cg	1.52	310
Cg-C	1.53	220
Cg-Oh	1.43	320
Cg-Os	1.46	285
Cg-H1	1.09	340
Cg-H2	1.09	340
C-O2	1.25	656
Oh-Ho	0.96	553
OW-HW	0.9572	450

Table A.3: Atoms and parameters involved in angle stretching.

Atom Types Involved in Angle	θ^0 (°)	k_a (kcal/mol)
Cg-Cg-Cg	113.5	45
Cg-Cg-Oh	107.5	70
Cg-Cg-H1	111	45
Cg-Oh-Ho	109.5	55
Cg-Os-Cg	111.6	50
Cg-Cg-Os	108.5	70
Cg-Cg-H2	111	45
Oh-Cg-H1	110	60
Cg-Cg-C	111.1	63
Cg-C-O2	115	70
Os-Cg-H2	110	60
C-Cg-Os	112.36	63
C-Cg-H1	109.5	50
O2-C-O2	126	80
Oh-Cg-Os	112	100
Oh-Cg-H2	110	60
Os-Cg-H1	110	60
Os-Cg-Os	112	100
HW-OW-HW	104.52	55

Table A.4: Atoms and parameters involved in dihedral torsions.

Atom Types Involved in Dihedrals	$k_{d,n}$ (kcal/mol)	γ (°)	n (unitless)
Cg-Cg-Cg-Cg	0.45	0	1
Cg-Cg-Cg-Oh	0.1	0	3
Cg-Cg-Cg-H1	0.15	0	3
Cg-Cg-Oh-Ho	0.18	0	3
Cg-Os-Cg-C	0.32	0	3
	0.45	0	2
	-0.6	0	1
Cg-Os-Cg-H1	0.27	0	3
Cg-Cg-Os-Cg	0.16	0	3
Oh-Cg-Cg-Oh	-0.1	0	1
	0.95	0	2
	0.55	0	3
Oh-Cg-Cg-Os	-1.1	0	1
	0.25	0	2
Oh-Cg-Cg-H1	0.05	0	3
Oh-Cg-Cg-H2	0.05	0	3
Cg-Cg-Cg-C	0.45	0	1
Cg-Cg-Cg-Os	-0.27	0	1
Cg-Cg-Cg-H2	0.15	0	3
Cg-Cg-C-O2	0.01	0	3
	-0.41	0	2
	0.005	0	1
Cg-Os-Cg-Oh	0.96	0	3
	1.38	0	2
	1.08	0	1
Cg-Os-Cg-H2	0.6	0	2
	0.1	0	3
Cg-Os-Cg-Os	0.96	0	3
	1.38	0	2
	1.08	0	1
Os-Cg-Oh-Ho	0.18	0	3
Os-Cg-Cg-H1	0.05	0	3
Os-Cg-Cg-Os	0.4	0	2
C-Cg-Cg-Oh	0.1	0	3
	0.2	0	2
	-2.5	0	1
C-Cg-Cg-H1	0.1	0	3
C-Cg-Cg-Os	-0.1	0	3
	0.1	0	2
	-1	0	1
O2-C-Cg-Os	0.02	0	3
	-0.725	0	2
	0.02	0	1
O2-C-Cg-H1	0	0	1
H1-Cg-Cg-H2	0.17	0	3
H1-Cg-Cg-H1	0.17	0	3

Table A.5: Non-bonded parameters.

Atom Type	σ (Å)	ϵ (kcal/mol)
Cg	3.39967	0.10940
C	3.39967	0.08600
Oh	3.06647	0.21040
Os	3.00001	0.17000
O2	2.95992	0.21000
H1	2.47135	0.01570
H2	2.29317	0.01570
Ho	0.35636	0.03000
OW	3.15070	0.10200
HW	1.00000	0.00000
Na^+	2.62459	0.03118
Ca^{2+}	2.92571	0.10186
Cl^-	3.83086	0.52153

Continuation of Table A.4			
Atom Types Involved in Dihedrals	$k_{d,n}$ (kcal/mol)	γ (°)	n (unitless)
Ho-Oh-Cg-H1	0.18	0	3
Ho-Oh-Cg-H2	0.18	0	3
C-O2-O2-Cg (improper)	87.864	180	2

Appendix B

Example design of experiments calculations

Included in this appendix is an example design of experiments calculation which shows how the t-test is applied to a dataset to determine statistical significance. The example dataset is small (three values) and fictional which produced very large errors.

An example of how to calculate the standard error for an intercept effect and a factor (sodium alginate G:M ratio) is given in Equations B.1-B.3.

$$\text{Let } X = \begin{bmatrix} +1 & -1 \\ +1 & -1 \\ +1 & +1 \end{bmatrix}$$

where column 1 is the intercept, and column 2 is the coded levels of factor sodium alginate G:M ratio.

$$\text{Therefore, } Y = \begin{bmatrix} 10.16 \\ 22.20 \\ 21.37 \end{bmatrix}$$

This produces a $\hat{\beta}$ matrix of

$$\hat{\beta} = \begin{bmatrix} 1201.6 \\ 166.08 \end{bmatrix} \tag{B.1}$$

We can use this to find the model predicted values:

$$\hat{Y} = X\hat{\beta} = \begin{bmatrix} +1 & -1 \\ +1 & -1 \\ +1 & +1 \end{bmatrix} \begin{bmatrix} 1201.6 \\ 166.08 \end{bmatrix}$$
$$= \begin{bmatrix} 1035.52 \\ 1035.52 \\ 1367.68 \end{bmatrix}$$

Now we calculate the residuals: (B.2)

$$e = Y - \hat{Y} = \begin{bmatrix} 10.16 \\ 22.20 \\ 21.37 \end{bmatrix} - \begin{bmatrix} 1035.52 \\ 1035.52 \\ 1367.68 \end{bmatrix} = \begin{bmatrix} -1025.36 \\ -1013.32 \\ -1346.31 \end{bmatrix}$$

We use this to calculate the Sum of Squares Error (SSE):

$$SSE = \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 = (-1025.36)^2 + (-1013.32)^2 + (1346.31)^2 = 3890731$$

Now we calculate the Mean Squared Error (MSE) using the Degrees of Freedom (DF):

$$DF = n - p = 3 - 2 = 1$$

Where n is the number of runs and p is the number of parameters.

$$MSE = \frac{SSE}{DF} = \frac{3890731}{1} = 3890731$$

We now use the matrix $(X^T X)^{-1}$ which we calculated when finding $\hat{\beta}$ to find the Standard Error (SE) for the intercept and sodium alginate G:M ratio:

$$SE(\hat{\beta}_1) = \sqrt{MSE \times (X^T X)^{-1}_{11}}$$

$$SE(\hat{\beta}_2) = \sqrt{MSE \times (X^T X)^{-1}_{22}}$$

Where $(X^T X)^{-1}_{11}$ and $(X^T X)^{-1}_{22}$ denote the row (first number) and column (second number) to pull from the matrix, and links to which factor is being assessed: the first (intercept) or the second (sodium alginate G:M ratio).

$$SE(\hat{\beta}_1) = \sqrt{3890731 \times 3} = 11672194$$

$$SE(\hat{\beta}_2) = \sqrt{3890731 \times 3} = 11672194$$

(B.3)

Following the calculation of the standard error for each factor, and compounded factor, we can calculate the t-statistic for each of these effects. The null hypothesis

here is that $\beta = 0$ and the factor has no effect on the response. The alternative hypothesis is $\beta \neq 0$ and the factor does have an effect on the response. The examples given in Equations B.1-B.3 are continued in Equation B.4. We use the DF value and the confidence level of the test (0.05) to find our critical t-value. This employs t-tables which are readily available. For $DF = 1$, $\alpha = 0.05$, and a two-tailed test, the critical t-value is 12.71. If our t-statistics are above 12.71 or below -12.71, we reject the null hypothesis. If it is between -12.71 and 12.71, we accept the null hypothesis. Since both the test-statistics found in Equation B.4 are within the critical t-value range, we accept the null hypothesis and can conclude that in this small example dataset neither the intercept nor the sodium alginate G:M ratio had a statistically significant effect on the response values.

$$
\begin{aligned}
t &= \frac{\hat{\beta}_i}{SE(\hat{\beta}_i)} \\
t_1 &= \frac{\hat{\beta}_1}{SE(\hat{\beta}_1)} = \frac{1201.6}{11672194} = 1.03 \times 10^{-4} \\
t_2 &= \frac{\hat{\beta}_2}{SE(\hat{\beta}_2)} = \frac{166.08}{11672194} = 1.42 \times 10^{-5}
\end{aligned}
\tag{B.4}
$$

Appendix C

Example Calculation of Mulliken Charges

As an example of calculating Mulliken charges, we will look at the partial atomic charges on a carbon to oxygen (C-O) bond. Each of these atoms have a 2s, $2p_x$, $2p_y$ and $2p_z$ orbital. For simplicity we will look only at the bond formed by the 2s orbitals. To construct the molecular orbital we will use Linear Combination of Atomic Orbitals (LCAO) as shown in Equation C.1, where ψ is the molecular or bonding orbital created when the C and O atoms bond, c is a coefficient which shows how much each atom contributes to the bond and therefore determines bond polarity, χ is the atomic basis function, and C and O represent the atoms involved in the bond.

$$\psi_{bonding} = c_C\chi_C + c_O\chi_O \tag{C.1}$$

To find the coefficients c_C and c_O, we use Equation C.2 where H is the Coulomb integral which represents the interaction of an electron with the nucleus of the atom it belongs to, and E is the bonding orbital energy. We use empirical values as seen in Equation C.2.

$$\begin{aligned} \frac{c_C}{c_O} &= \frac{H_{OO} - E}{E - H_{CC}} \\ \frac{c_C}{c_O} &= \frac{-32.3eV - (-25eV)}{-25eV - (-21.4eV)} \\ \frac{c_C}{c_O} &= 2.03eV \end{aligned} \tag{C.2}$$

We then normalise the molecular orbital to ensure that the total probability of

finding an electron is 1. The normalisation condition given in Equation C.1 where S_{CO} is the overlap integral between the carbon atomic orbital and the oxygen atomic orbital and is approximated to 0.2, and we use Equation C.2 to simplify the calculation.

$$\begin{aligned} c_C^2 + c_O^2 + 2c_C c_O S_{CO} &= 1 \\ (2.03c_O)^2 + c_O^2 + 2(2.03c_O)c_O(0.2) &= 1 \\ c_O &= \pm 0.41 \\ \text{therefore: } c_C &= \pm 0.83 \end{aligned} \tag{C.3}$$

From these coefficients, we can calculate the density matrix elements of the bond. The elements describe how the electrons are distributed among orbitals and atoms. For closed-shell systems we use Equation C.4.

$$\begin{aligned} P_{\mu\nu} &= 2\sum_i c_{i\mu}c_{i\nu} \\ P_{CC} &= 2\sum c_C^2 = 1.38 \\ P_{OO} &= 2\sum c_O^2 = 0.34 \\ P_{CO} &= 2\sum c_C c_O = 0.68 \end{aligned} \tag{C.4}$$

We can now calculate the Mulliken partial charges using Equation C.5.

$$\begin{aligned} q_i &= P_{ii}S_{ii} + P_{CO}S_{CO} \\ q_C &= P_{CC}S_{CC} + P_{CO}S_{CO} = 1.38 \times 1 + 0.68 \times 0.2 = +1.52 \\ q_O &= P_{OO}S_{OO} + P_{CO}S_{CO} = 0.34 \times 1 + 0.68 \times 0.2 = +0.48 \end{aligned} \tag{C.5}$$

These values mean that the carbon atom in the C-O bond has 1.52 electrons of the 2 available and oxygen has 0.48 electrons; creating a negative C atom and positive O atom.

Jakalian *et al.* published a list of bond charge corrections which can be applied directly to the bond in question. The bond charge correction for a C-O bond as we have been using in our example is 0.04. We apply this to the Mulliken partial charges calculated previously as shown in Equation C.6. This gives us final charge distribution

of C getting 1.48 electrons of the 2 available and oxygen getting the rest, meaning that the carbon atom is negatively charged and the oxygen atom is positively charged.

$$\begin{aligned} Q_i &= q_i \pm BCC \\ Q_C &= q_C - BCC = +1.52 - 0.04 = +1.48 \\ Q_O &= q_O + BCC = +0.48 + 0.04 = +0.52 \end{aligned} \tag{C.6}$$

Appendix D

Further results and images from Chapter 3

Included below are the images and charge results of all structures study in analysis of geometric fluctuation effects on partial atomic charges. The images of all G and M monomer structures, dimer structures, and trimer structures are included in Figures D.1 and D.2, D.5 and D.6, and D.9 and D.10, respectively. The partial atomic charge results are included in Figures D.3 and D.4, D.7 and D.8, and D.11 and D.12, respectively.

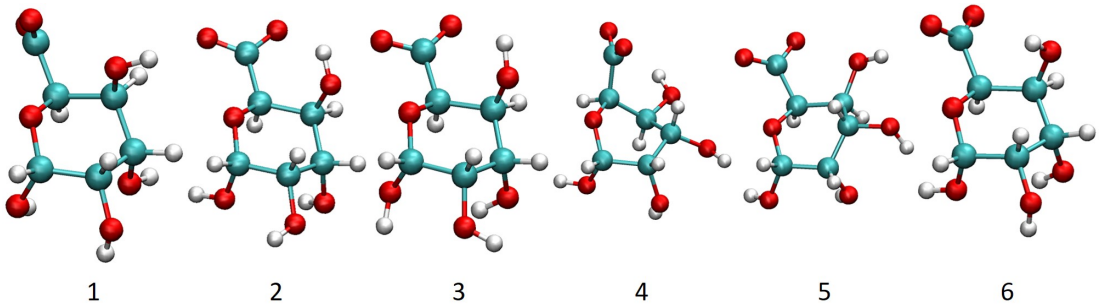


Figure D.1: All G monomer structures used in the study of effects of geometric fluctuations on partial atomic charges.

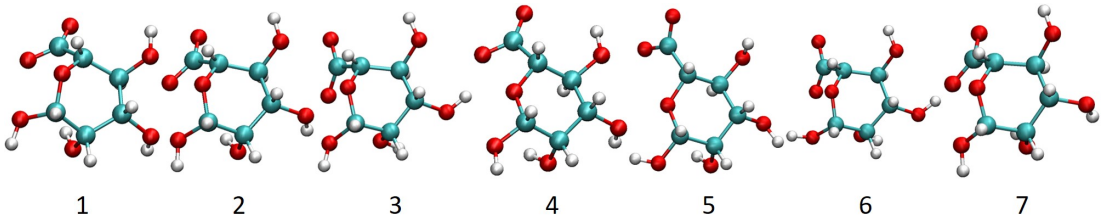


Figure D.2: All M monomer structures used in the study of effects of geometric fluctuations on partial atomic charges.

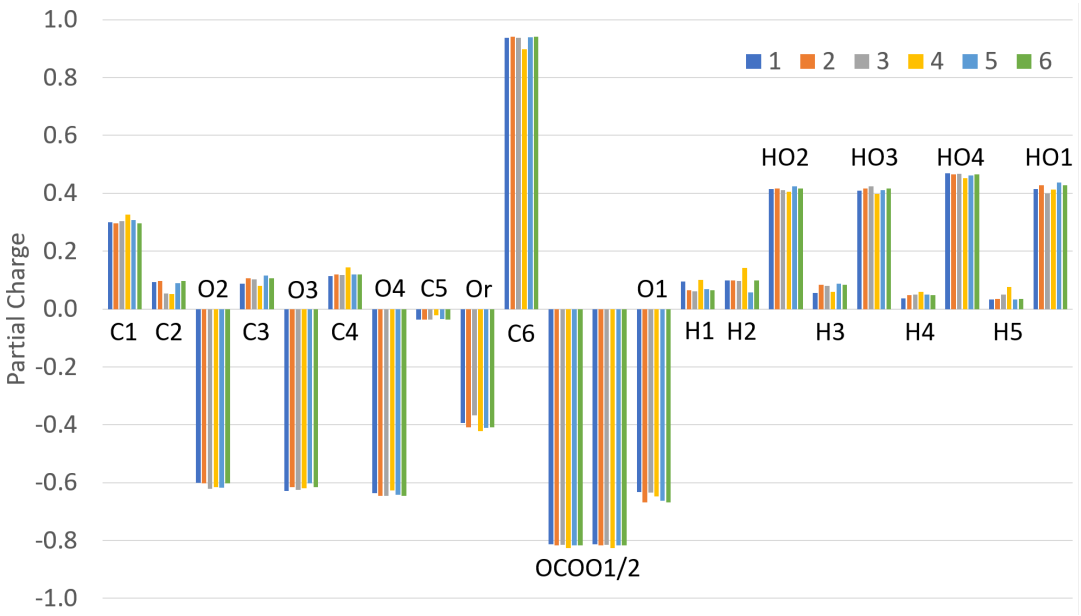


Figure D.3: Atomic partial charges for all G monomer structures used in the study of effects of geometric fluctuations on partial atomic charges.

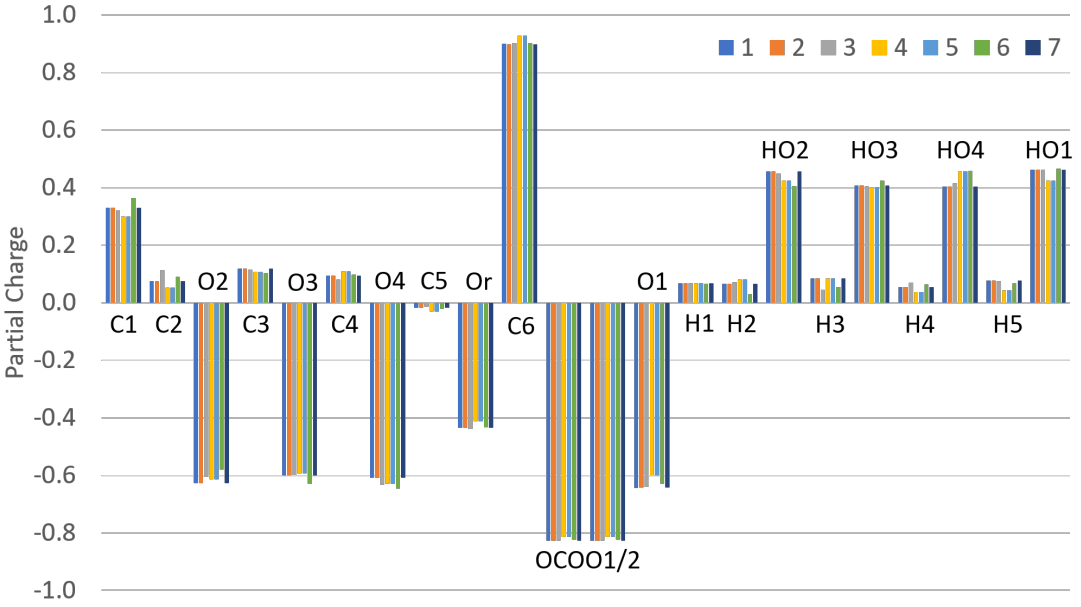


Figure D.4: Atomic partial charges for all M monomer structures used in the study of effects of geometric fluctuations on partial atomic charges.

The following images show the individual atomic partial charges calculated in the study which investigated the effects of chain lengthening on the partial atomic charges.

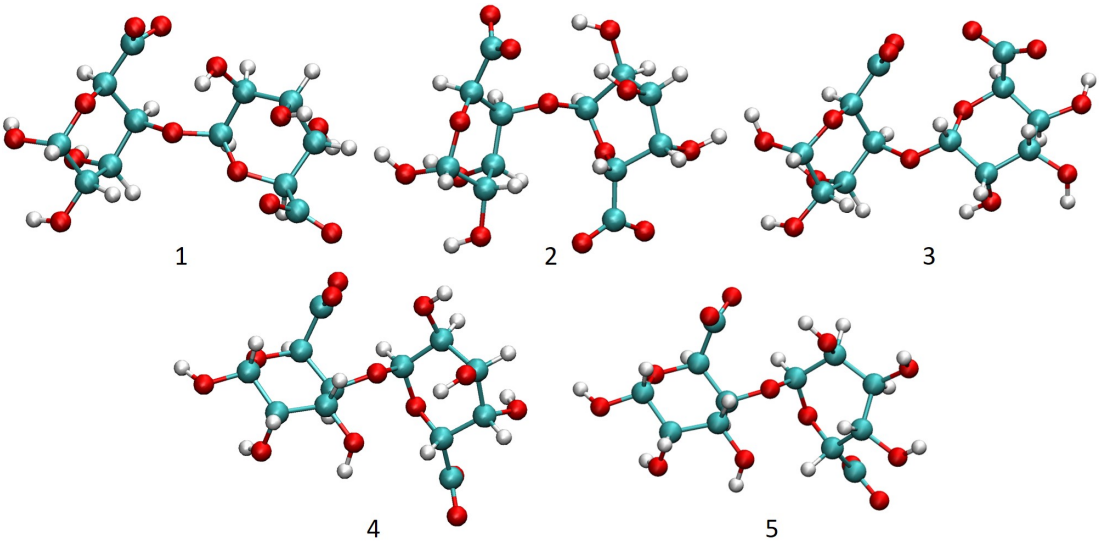


Figure D.5: All G dimer structures used in the study of effects of geometric fluctuations on partial atomic charges.

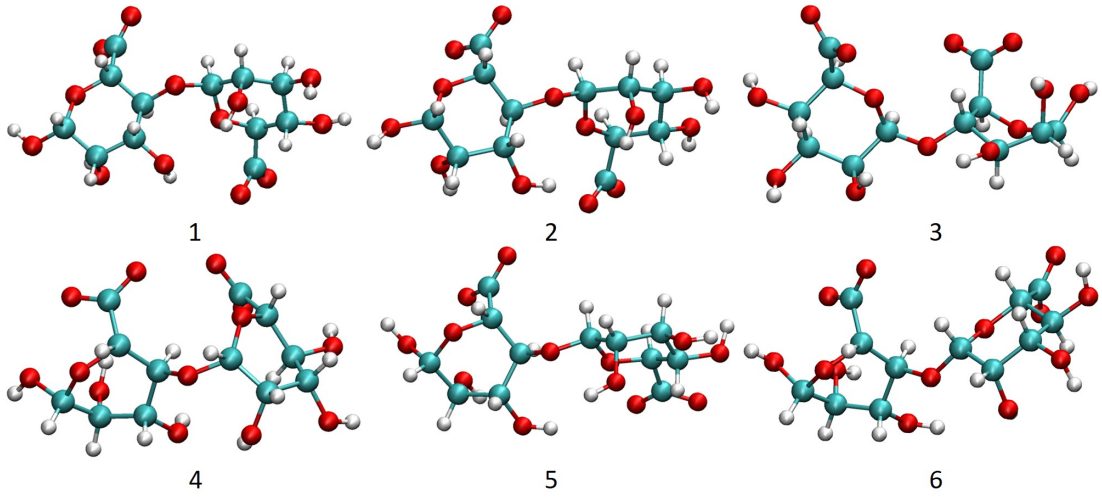


Figure D.6: All M dimer structures used in the study of effects of geometric fluctuations on partial atomic charges.

The results include charges for the 1'-end (Figure D.13), middle (Figure D.14), and 4'-end (Figure D.15) monomer types.

A comparison of the partial atomic charges calculated in this work with values used in literature is presented in Table D.1. The partial atomic charges taken from the works by Li *et al.* 2021, Perry *et al.* 2006, and Panczyk *et al.* 2018 are from lists of general

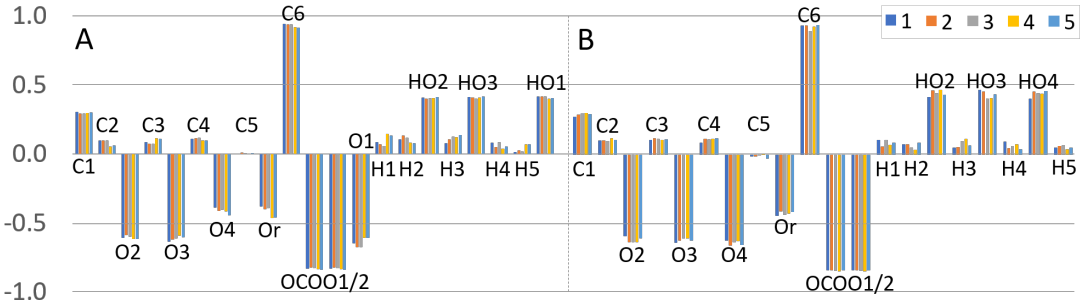


Figure D.7: Atomic partial charges for all G dimer structures used in the study of effects of geometric fluctuations on partial atomic charges.

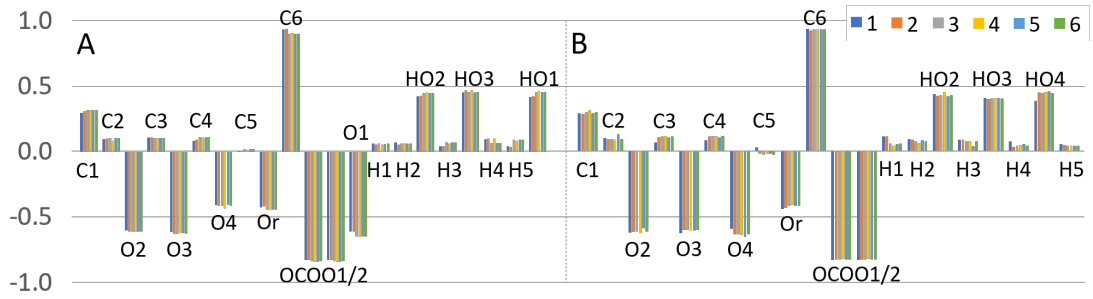


Figure D.8: Atomic partial charges for all M dimer structures used in the study of effects of geometric fluctuations on partial atomic charges.

charges where each atom type is assigned a charge, which is used in every instance of that atom type [152, 181, 234]. The work by Anderson *et al.* 2009 found the charges for a specific tetramer of alginate (and a dimer which is not included here) and so both middle monomers for the tetramer are included in Table D.1. The charges from literature presented in Table D.1 are commonly scaled to give a molecule charge of -N where N is the number of monomers included in the alginate chain. This produces an average charge of -1 per monomer.

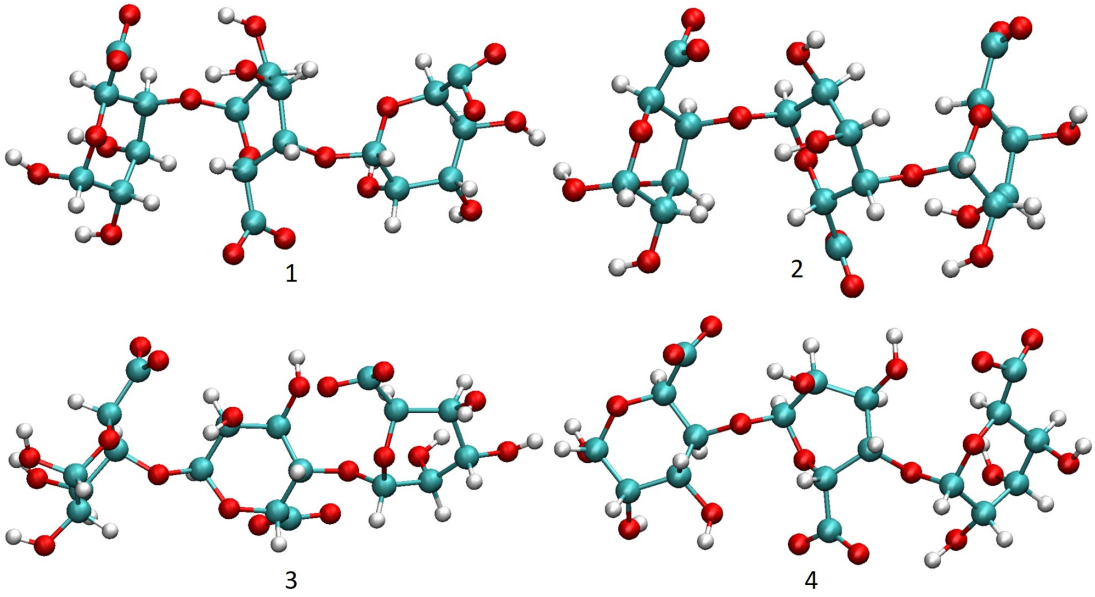


Figure D.9: All G trimer structures used in the study of effects of geometric fluctuations on partial atomic charges.

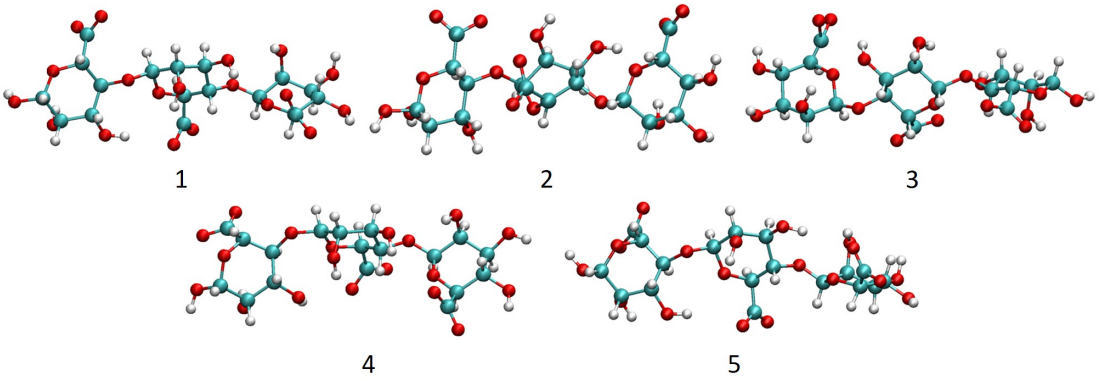


Figure D.10: All M trimer structures used in the study of effects of geometric fluctuations on partial atomic charges.

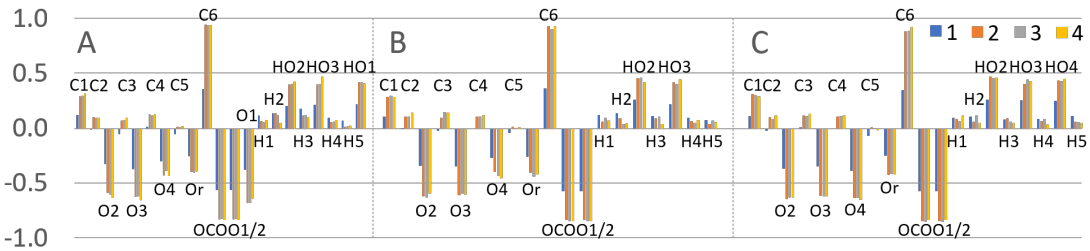


Figure D.11: Atomic partial charges for all G trimer structures used in the study of effects of geometric fluctuations on partial atomic charges.

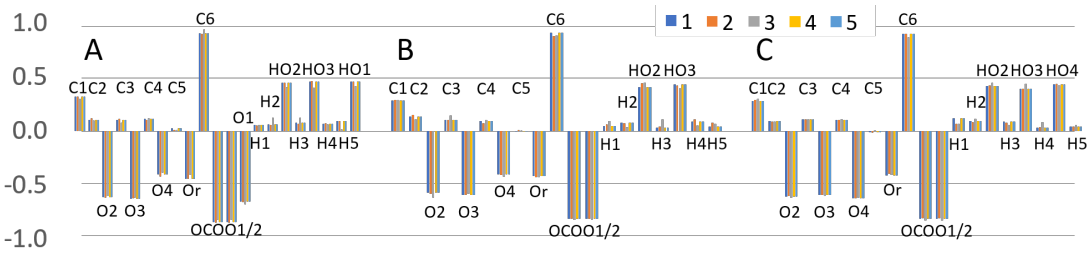


Figure D.12: Atomic partial charges for all M trimer structures used in the study of effects of geometric fluctuations on partial atomic charges

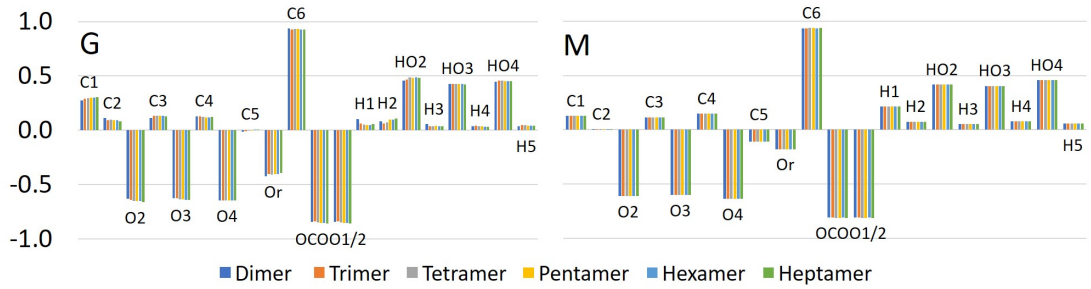


Figure D.13: Atomic partial charges for 1'-end monomers in all chain lengths used in the study of effects of chain lengthening on partial atomic charges.

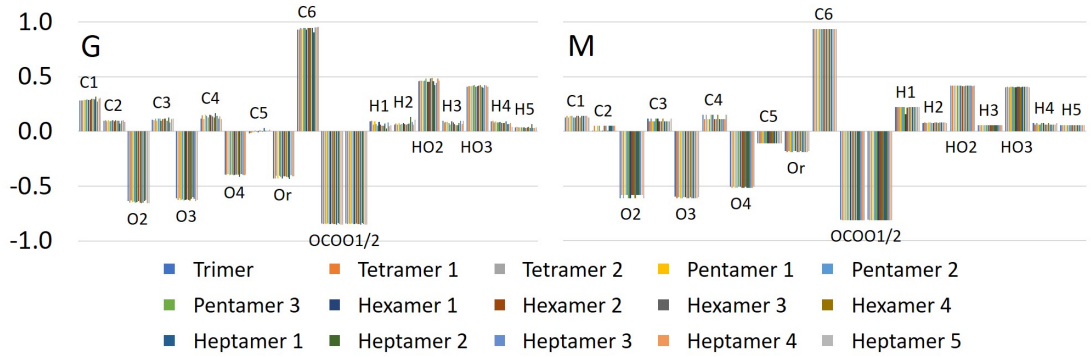


Figure D.14: Atomic partial charges for middle monomers in all chain lengths used in the study of effects of chain lengthening on partial atomic charges.

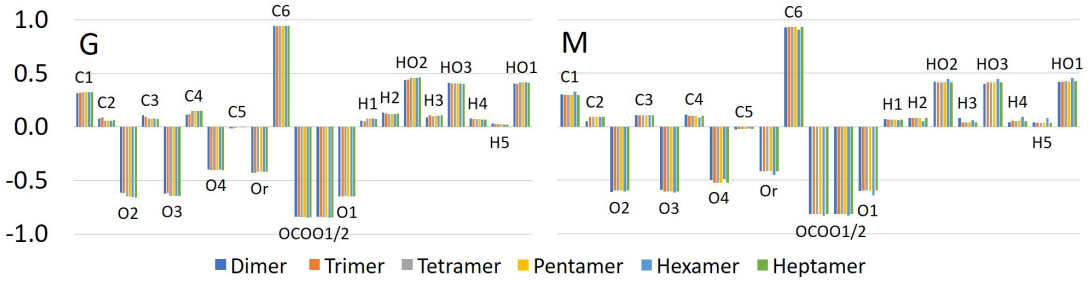


Figure D.15: Atomic partial charges for 4'-end monomers in all chain lengths used in the study of effects of chain lengthening on partial atomic charges.

Table D.1: The charge lists for M and G middle monomers calculated and optimised for use in simulated alginate systems in this work in comparison with the same values taken from literature. The values taken from history are not dependent on the residue type.

Atom	**Current Work**		**Values From Literature**				
	G	**M**	**Li *et al.* 2021[152]**	Anderson *et al.* 2009[235]		Perry *et al.* 2006[181]	Panczyk *et al.*[234]
C1	0.2978	0.1356	0.3121	0.1140	0.1120	-0.0700	
C2	0.0978	0.0349	0.2171	-0.0100	-0.0080	-0.0700	
O2	-0.6571	-0.5894	-0.7179	-0.3340	-0.3320	-0.3800	
C3	0.1117	0.1004	0.2171	0.0030	-0.0060	-0.0700	
O3	-0.6341	-0.6073	-0.7179	-0.3370	-0.3370	-0.3800	
C4	0.1393	0.1257	0.1821	0.0040	-0.0170	-0.0700	
O4	-0.4038	-0.5109	-0.3879	-0.3760	-0.3130	-0.3000	
C5	0.0027	-0.1093	-0.1479	0.0870	-0.0480	-0.0700	
Or	-0.4239	-0.1871	-0.3879	-0.2730	-0.2850	-0.3000	
C6	0.9578	0.9328	0.7121	0.3540	0.3570	0.1400	0.2840
OC2	-0.8612	-0.8099	-0.7879	-0.5820	-0.5540	-0.5700	-0.6420
OC1	-0.8612	-0.8099	-0.7879	-0.5440	-0.5670	-0.5700	-0.6420
H1	0.0657	0.2152	0.1121	0.1410	0.1040	0.1000	
H2	0.0768	0.0796	0.0721	0.1410	0.1180	0.1000	
HO2	0.4719	0.4146	0.4771	0.2290	0.2340	0.3500	
H3	0.0792	0.0552	0.0721	0.1140	0.0880	0.1000	
HO3	0.4220	0.4056	0.4771	0.2280	0.2600	0.3500	
H4	0.0795	0.0670	0.0421	0.0850	0.1430	0.1000	
H5	0.0391	0.0572	0.0421	0.0880	0.0860	0.3500	
Total	-1.0000	-1.0000	-1.3880	-1.1810	-1.3410	-1.5600	

Within this thesis, the work by Li *et al.* 2021[152] is referenced often, especially in Chapters 4 and 7, as a comparison between alginate systems. There are minimal differences between the charges calculated in Chapter 3 and those used by Li *et al.* 2021, as shown in Table D.1. The differences in the oxygen atoms, which interact with the system cations, are particularly minimal. This shows that differences seen in the behaviour of alginate, water, and cations in simulated systems in this work and the work by Li *et al.* 2021[152] is not likely to be due to differences in the alginate partial atomic charges.

Appendix E

Python script used to create alginate structures

The script included in Listing E.1 creates a structure file for a user defined chain length of either G or M residues. The script included in Listing E.2 is required to run Listing E.1 as it contains all the necessary data relevant to the G and M residues.

Listing E.1: Python script to create ordered G and M chains of any length. Listing E.2 must be in working directory.

```
#Python script to write molecule template files for M or G monomers, MG combination dimers, or pure chains of any length for the create_atoms LAMMPS method.

############################
#Import necessary software
############################

import pickle
import coordinates

################
#Step 1: Header
################

#Get number of monomers in chain required and whether it is M or G monomers
chain_length = int(input('Chain length? '))
monomer_type = input('monomer type? ')
#if chain_length != 2:
```

```
#    monomer_type = input('monomer type? ')
if chain_length == 2:
    monomer1_type = input('monomer 1 type? ')
    monomer2_type = input('monomer 2 type? ')
chain_description = input('Chain length and monomer type? (for header in mol temp file only) ')

#Calculate number of atoms, bonds, angles, dihedrals, and impropers for chain length specified
if chain_length == 1:
    atoms     = 22
    bonds     = 22
    angles    = 38
    dihedrals = 58
else:
    atoms     = 21 + (19 * (chain_length - 2)) + 20
    bonds     = 21 + (20 * (chain_length - 2)) + 21
    angles    = 41 + (37 * (chain_length - 2)) + 34
    dihedrals = 65 + (58 * (chain_length - 2)) + 51
impropers = chain_length

#Write header to molecule template file
with open('polymer_moltemp', 'w') as f:
    f.write('#Molecule template file created through python script for chain of ' + chain_description + '\n')

#Write number of atoms, bonds, angles, etc. to molecule template file
    f.write('\n' + str(atoms)     + ' atoms')
    f.write('\n' + str(bonds)     + ' bonds')
    f.write('\n' + str(angles)    + ' angles')
    f.write('\n' + str(dihedrals) + ' dihedrals')
    f.write('\n' + str(impropers) + ' impropers' + '\n' )

##############################
#Step 2: Coordinates of atoms
##############################

```

```
#import pickled lists
M_x_coords = pickle.load(open("M_x_coords.p", "rb"))
M_y_coords = pickle.load(open("M_y_coords.p", "rb"))
M_z_coords = pickle.load(open("M_z_coords.p", "rb"))
G_x_coords = pickle.load(open("G_x_coords.p", "rb"))
G_y_coords = pickle.load(open("G_y_coords.p", "rb"))
G_z_coords = pickle.load(open("G_z_coords.p", "rb"))

#Create atom lists for each monomer type and dimer types
atom_types_full = []
for atom in M_x_coords:
    atom_types_full.append(atom)

atom_types_1end = []
for atom in M_x_coords:
    atom_types_1end.append(atom)
del atom_types_1end[-3]

        #don't delete O1 as needed in positioning monomer
atom_types_mid = []
for atom in M_x_coords:
    atom_types_mid.append(atom)
del atom_types_mid[-1]
del atom_types_mid[-2]
del atom_types_mid[12]

        #don't delete O1 as needed in positioning monomer
atom_types_4end = []
for atom in M_x_coords:
    atom_types_4end.append(atom)
del atom_types_4end[-1]
del atom_types_4end[12]

atom_names = ['1C1', '1C2', '1O2', '1C3', '1O3', '1C4', '1O4', '1C5', '1
    Or', '1C6', '1O(COO2)', '1O(COO1)', '1O1', '1H1', '1H2', '1H(OH2)', '
    1H3', '1H(OH3)', '1H4', '1H5', '1H(OH1)', '2C1', '2C2', '2O2', '2C3',
     '2O3', '2C4', '2O4', '2C5', '2Or', '2C6', '2O(COO2)', '2O(COO1)', '2
```

```
    H1', '2H2', '2H(OH2)', '2H3', '2H(OH3)', '2H4', '2O(OH4)', '2H5']

#Insert coordinates section title
with open('polymer_moltemp', 'a') as f:
    f.write('\n' + 'Coords          #atom coordinates [ID x y z]' + '\n'
        + '\n')

#A - Monomers
if chain_length == 1:

    #A1 - M monomer
    if monomer_type == 'M' or monomer_type == 'm':

        #Write each atom's x, y, and z coordinates preceded by the atom
            number to file
        with open("polymer_moltemp", "a") as f:
            for atom, n in zip(atom_types_full, range(1,23)):
                f.write(str(n) + ' ' + str(M_x_coords[atom]) + ' ' + str
                    (M_y_coords[atom]) + ' ' + str(M_z_coords[atom]) + '\
                    n' )

    #A2 - G monomer
    elif monomer_type == 'G' or monomer_type == 'g':

        #Write each atom's x, y, and z coordinates preceded by the atom
            number to file
        with open("polymer_moltemp", "a") as f:
            for atom, n in zip(atom_types_full, range(1,23)):
                f.write(str(n) + ' ' + str(G_x_coords[atom]) + ' ' + str
                    (G_y_coords[atom]) + ' ' + str(G_z_coords[atom]) + '\
                    n' )

#C - Chains
else:
    if monomer_type == 'M' or monomer_type == 'm':
        with open('polymer_moltemp', 'a') as f:
            for atom, n in zip(atom_types_1end, range(1,22)):
```

```
115                 f.write(str(n) + ' ' + str(M_x_coords[atom]) + ' ' + str
                        (M_y_coords[atom]) + ' ' + str(M_z_coords[atom]) + '\
                        n')
116
117             k = 22
118             for i in range(1, chain_length - 1):
119                 if i % 2 == 0:
120                     c = 1
121                 else:
122                     c = -1
123                 for atom in atom_types_mid:
124                     if c == -1:
125                         f.write(str(k) + ' ' + str(pow(-1, i) *
                                M_x_coords[atom]) + ' ' + str(M_y_coords[atom
                                ] - (3 * i)) + ' ' + str(pow(-1, i) *
                                M_z_coords[atom] - 3) + '\n')
126                         k += 1
127                     elif c == 1:
128                         f.write(str(k) + ' ' + str(pow(-1, i) *
                                M_x_coords[atom]) + ' ' + str(M_y_coords[atom
                                ] - (3 * i)) + ' ' + str(pow(-1, i) *
                                M_z_coords[atom]) + '\n')
129                         k += 1
130
131             for atom in atom_types_4end:
132                 if chain_length % 2 == 0:
133                     c = -1
134                 else:
135                     c = 1
136                 if c == -1:
137                     f.write(str(k) + ' ' + str(pow(-1, chain_length - 1)
                            * M_x_coords[atom]) + ' ' + str(M_y_coords[atom]
                            - (3 * (chain_length - 1))) + ' ' + str(pow(-1,
                            chain_length - 1) * M_z_coords[atom] - 3) + '\n')
138                     k += 1
139                 elif c == 1:
140                     f.write(str(k) + ' ' + str(pow(-1, chain_length - 1)
```

```
                        * M_x_coords[atom]) + ' ' + str(M_y_coords[atom]
                        - (3 * (chain_length - 1))) + ' ' + str(pow(-1,
                       chain_length - 1) * M_z_coords[atom]) + '\n')
141                 k += 1
142
143     elif monomer_type == 'G' or monomer_type == 'g':
144         with open('polymer_moltemp', 'a') as f:
145             for atom, n in zip(atom_types_1end, range(1,22)):
146                 f.write(str(n) + ' ' + str(G_x_coords[atom]) + ' ' + str
                    (G_y_coords[atom]) + ' ' + str(G_z_coords[atom]) + '\
                    n')
147
148             k = 22
149             for i in range(1, chain_length - 1):
150                 if i % 2 == 0:
151                     c = 1
152                 else:
153                     c = -1
154                 for atom in atom_types_mid:
155                     if c == -1:
156                         f.write(str(k) + ' ' + str(-1 * G_x_coords[atom
                            ]) + ' ' + str(G_y_coords[atom] - (3 * i)) +
                            ' ' + str(-1 * G_z_coords[atom] + 3) + '\n')
157                         k += 1
158                     elif c == 1:
159                         f.write(str(k) + ' ' + str(G_x_coords[atom]) + '
                             ' + str(G_y_coords[atom] - (3 * i)) + ' ' +
                            str(G_z_coords[atom]) + '\n')
160                         k += 1
161
162             for atom in atom_types_4end:
163                 if chain_length % 2 == 0:
164                     c = -1
165                 else:
166                     c = 1
167                 if c == -1:
168                     f.write(str(k) + ' ' + str(-1 * G_x_coords[atom]) +
```

```
                        ' ' + str(G_y_coords[atom] - (3 * (chain_length -
                         1))) + ' ' + str(-1 * G_z_coords[atom] + 3) + '\
                        n')
169                     k += 1
170                 elif c == 1:
171                     f.write(str(k) + ' ' + str(G_x_coords[atom]) + ' ' +
                            str(G_y_coords[atom] - (3 * (chain_length - 1)))
                            + ' ' + str(G_z_coords[atom]) + '\n')
172                     k += 1
173
174
175 ###############
176 #Step 3: Types
177 ###############
178
179 #import pickled lists
180 atom_types = pickle.load(open('atom_types.p', 'rb'))
181
182 #Create atom lists for each of the four monomer types
183 atom_types_full = []
184 for atom in atom_types:
185     atom_types_full.append(atom)
186
187 atom_types_1end = []
188 for atom in atom_types:
189     atom_types_1end.append(atom)
190 del atom_types_1end[-3]
191
192 atom_types_mid = []
193 for atom in atom_types:
194     atom_types_mid.append(atom)
195 del atom_types_mid[-1]
196 del atom_types_mid[-2]
197 del atom_types_mid[12]
198
199 atom_types_4end = []
200 for atom in atom_types:
```

```
    atom_types_4end.append(atom)
del atom_types_4end[-1]
del atom_types_4end[12]

#Create lists of atom types for monomers and chains
types_full = []
types_1end = []
types_mid = []
types_4end = []
types_chain = []

if chain_length == 1:
    for atom in atom_types_full:
        types_chain.append(atom_types[atom])

elif chain_length > 1:
    for atom in atom_types_1end:
        types_chain.append(atom_types[atom])
    types_chain[6] = 5
    for i in range(1, (chain_length - 1)):
        for atom in atom_types_mid:
            types_chain.append(atom_types[atom])
        types_chain[-13] = 5
    for atom in atom_types_4end:
        types_chain.append(atom_types[atom])

#Write atom types and atom numbers to file
with open('polymer_moltemp', 'a') as f:
    f.write('\n' + 'Types          #atom types [ID type]' + '\n')
    a = 1
    for i in range(len(types_chain)):
        f.write('\n' + str(a) + ' ' + str(types_chain[i]))
        a += 1

#################
#Step 4: Charges
#################
```

```
238
239 #import relevant pickled lists
240 #M
241 M_charges_full = pickle.load(open('M_charges_full.p', 'rb'))
242 M_charges_1end = pickle.load(open('M_charges_1end.p', 'rb'))
243 M_charges_mid  = pickle.load(open('M_charges_mid.p',  'rb'))
244 M_charges_4end = pickle.load(open('M_charges_4end.p', 'rb'))
245
246 #G
247 G_charges_full = pickle.load(open('G_charges_full.p', 'rb'))
248 G_charges_1end = pickle.load(open('G_charges_1end.p', 'rb'))
249 G_charges_mid  = pickle.load(open('G_charges_mid.p',  'rb'))
250 G_charges_4end = pickle.load(open('G_charges_4end.p', 'rb'))
251
252 #Write header and charges to file
253 with open('polymer_moltemp', 'a') as f:
254     f.write('\n' + '\n' + 'Charges            #charges [ID charge]' + '\
        n')
255     i = 1
256
257     if chain_length == 1:
258         if monomer_type == 'm' or monomer_type == 'M':
259             for charges in M_charges_full:
260                 f.write('\n' + str(i) + ' ' + str(M_charges_full[charges
                    ]))
261                 i += 1
262         elif monomer_type == 'g' or monomer_type == 'G':
263             for charges in G_charges_full:
264                 f.write('\n' + str(i) + ' ' + str(G_charges_full[charges
                    ]))
265                 i += 1
266
267     elif chain_length == 2:
268         if monomer1_type == 'm' or monomer1_type == 'M':
269             for charges in M_charges_1end:
270                 f.write('\n' + str(i) + ' ' + str(M_charges_1end[charges
                    ]))
```

```
271                 i += 1
272         elif monomer1_type == 'g' or monomer1_type == 'G':
273             for charges in G_charges_1end:
274                 f.write('\n' + str(i) + ' ' + str(G_charges_1end[charges
                        ]))
275                 i += 1
276         if monomer2_type == 'm' or monomer2_type == 'M':
277             for charges in M_charges_4end:
278                 f.write('\n' + str(i) + ' ' + str(M_charges_4end[charges
                        ]))
279                 i += 1
280         elif monomer2_type == 'g' or monomer2_type == 'G':
281             for charges in G_charges_4end:
282                 f.write('\n' + str(i) + ' ' + str(G_charges_4end[charges
                        ]))
283                 i += 1
284
285     else:
286         if monomer_type == 'm' or monomer_type == 'M':
287             for charges in M_charges_1end:
288                 f.write('\n' + str(i) + ' ' + str(M_charges_1end[charges
                        ]))
289                 i += 1
290             for j in range(chain_length - 2):
291                 for charges in M_charges_mid:
292                     f.write('\n' + str(i) + ' ' + str(M_charges_mid[
                            charges]))
293                     i += 1
294             for charges in M_charges_4end:
295                 f.write('\n' + str(i) + ' ' + str(M_charges_4end[charges
                        ]))
296                 i += 1
297
298         elif monomer_type == 'g' or monomer_type == 'G':
299             for charges in G_charges_1end:
300                 f.write('\n' + str(i) + ' ' + str(G_charges_1end[charges
                        ]))
```

```
                i += 1
            for j in range(chain_length - 2):
                for charges in G_charges_mid:
                    f.write('\n' + str(i) + ' ' + str(G_charges_mid[
                        charges]))
                    i += 1
            for charges in G_charges_4end:
                f.write('\n' + str(i) + ' ' + str(G_charges_4end[charges
                    ]))
                i += 1

###############
#Step 5: Bonds
###############

#load necessary pickled lists
b_type_full  = pickle.load(open('bond_type_full.p', 'rb'))
b_atom1_full = pickle.load(open('bond_atom1_full.p', 'rb'))
b_atom2_full = pickle.load(open('bond_atom2_full.p', 'rb'))

b_type_1end  = pickle.load(open('bond_type_1end.p', 'rb'))
b_atom1_1end = pickle.load(open('bond_atom1_1end.p', 'rb'))
b_atom2_1end = pickle.load(open('bond_atom2_1end.p', 'rb'))

b_type_mid   = pickle.load(open('bond_type_mid.p', 'rb'))
b_atom1_mid  = pickle.load(open('bond_atom1_mid.p', 'rb'))
b_atom2_mid  = pickle.load(open('bond_atom2_mid.p', 'rb'))

b_type_4end  = pickle.load(open('bond_type_4end.p', 'rb'))
b_atom1_4end = pickle.load(open('bond_atom1_4end.p', 'rb'))
b_atom2_4end = pickle.load(open('bond_atom2_4end.p', 'rb'))

#create empty lists to add pickled values to
bond_type_full, bond_atom1_full, bond_atom2_full, bond_type_1end,
    bond_atom1_1end, bond_atom2_1end, bond_type_mid, bond_atom1_mid,
    bond_atom2_mid, bond_type_4end, bond_atom1_4end, bond_atom2_4end =
    [], [], [], [], [], [], [], [], [], [], [], []
```

```
333
334 #create lists of necessary values
335 for a, b, c in zip(b_type_full, b_atom1_full, b_atom2_full):
336     bond_type_full.append(b_type_full[a])
337     bond_atom1_full.append(b_atom1_full[b])
338     bond_atom2_full.append(b_atom2_full[c])
339
340 for d, e, f in zip(b_type_1end, b_atom1_1end, b_atom2_1end):
341     bond_type_1end.append(b_type_1end[d])
342     bond_atom1_1end.append(b_atom1_1end[e])
343     bond_atom2_1end.append(b_atom2_1end[f])
344
345 for g, h, i in zip(b_type_mid, b_atom1_mid, b_atom2_mid):
346     bond_type_mid.append(b_type_mid[g])
347     bond_atom1_mid.append(b_atom1_mid[h])
348     bond_atom2_mid.append(b_atom2_mid[i])
349
350 for j, k, l in zip(b_type_4end, b_atom1_4end, b_atom2_4end):
351     bond_type_4end.append(b_type_4end[j])
352     bond_atom1_4end.append(b_atom1_4end[k])
353     bond_atom2_4end.append(b_atom2_4end[l])
354
355 #Write header
356 with open('polymer_moltemp', 'a') as f:
357     f.write('\n' + '\n' + 'Bonds          #bonds [ID type atom1 atom2]'
        + '\n')
358
359 #Monomers
360 if chain_length == 1:
361     with open('polymer_moltemp', 'a') as f:
362         for i in range(len(bond_type_full)):
363             f.write('\n' + str(i + 1) + ' ' + str(bond_type_full[i]) + '
                 ' + str(bond_atom1_full[i]) + ' ' + str(bond_atom2_full[
                i]))
364
365 #Dimers and chains
366 else:
```

```
367     with open('polymer_moltemp', 'a') as f:
368         for i in range(len(bond_type_1end)):
369             f.write('\n' + str(i + 1) + ' ' + str(bond_type_1end[i]) + '
                 ' + str(bond_atom1_1end[i]) + ' ' + str(bond_atom2_1end[
                i]))
370         f.write('\n' + str(i + 2) + ' ' + str(4) + ' ' + str(7) + ' ' +
            str(22))
371
372         j = 23
373         l = 21
374         for k in range(chain_length - 2):
375             for i in range(len(bond_type_mid)):
376                 f.write('\n' + str(j) + ' ' + str(bond_type_mid[i]) + '
                     ' + str(bond_atom1_mid[i] + l) + ' ' + str(
                     bond_atom2_mid[i] + l))
377                 j += 1
378             f.write('\n' + str(j) + ' ' + str(4) + ' ' + str(7 + l) + '
                 ' + str(1 + l + 19))
379             j += 1
380             l += 19
381         for i in range(len(bond_type_4end)):
382             f.write('\n' + str(j) + ' ' + str(bond_type_4end[i]) + ' ' +
                 str(bond_atom1_4end[i] + l) + ' ' + str(bond_atom2_4end[i
                ] + l))
383             j += 1
384
385 ################
386 #Step 6: Angles
387 ################
388
389 #load necessary pickled lists
390 a_type_full  = pickle.load(open('angle_type_full.p', 'rb'))
391 a_atom1_full = pickle.load(open('angle_atom1_full.p', 'rb'))
392 a_atom2_full = pickle.load(open('angle_atom2_full.p', 'rb'))
393 a_atom3_full = pickle.load(open('angle_atom3_full.p', 'rb'))
394
395 a_type_1end  = pickle.load(open('angle_type_1end.p', 'rb'))
```

```
396 a_atom1_1end = pickle.load(open('angle_atom1_1end.p', 'rb'))
397 a_atom2_1end = pickle.load(open('angle_atom2_1end.p', 'rb'))
398 a_atom3_1end = pickle.load(open('angle_atom3_1end.p', 'rb'))
399
400 a_type_mid  = pickle.load(open('angle_type_mid.p', 'rb'))
401 a_atom1_mid = pickle.load(open('angle_atom1_mid.p', 'rb'))
402 a_atom2_mid = pickle.load(open('angle_atom2_mid.p', 'rb'))
403 a_atom3_mid = pickle.load(open('angle_atom3_mid.p', 'rb'))
404
405 a_type_4end  = pickle.load(open('angle_type_4end.p', 'rb'))
406 a_atom1_4end = pickle.load(open('angle_atom1_4end.p', 'rb'))
407 a_atom2_4end = pickle.load(open('angle_atom2_4end.p', 'rb'))
408 a_atom3_4end = pickle.load(open('angle_atom3_4end.p', 'rb'))
409
410 #create empty lists to add necessary values to
411 angle_type_full, angle_atom1_full, angle_atom2_full, angle_atom3_full,
    angle_type_1end, angle_atom1_1end, angle_atom2_1end, angle_atom3_1end
    , angle_type_mid, angle_atom1_mid, angle_atom2_mid, angle_atom3_mid,
    angle_type_4end, angle_atom1_4end, angle_atom2_4end, angle_atom3_4end
     = [], [], [], [], [], [], [], [], [], [], [], [], [], [], []
412
413 #create lists of angle parameters
414 for a, b, c, d in zip(a_type_full, a_atom1_full, a_atom2_full,
    a_atom3_full):
415         angle_type_full.append(a_type_full[a])
416         angle_atom1_full.append(a_atom1_full[b])
417         angle_atom2_full.append(a_atom2_full[c])
418         angle_atom3_full.append(a_atom3_full[d])
419
420 for e, f, g, h in zip(a_type_1end, a_atom1_1end, a_atom2_1end,
    a_atom3_1end):
421         angle_type_1end.append(a_type_1end[e])
422         angle_atom1_1end.append(a_atom1_1end[f])
423         angle_atom2_1end.append(a_atom2_1end[g])
424         angle_atom3_1end.append(a_atom3_1end[h])
425
426 for i, j, k, l in zip(a_type_mid, a_atom1_mid, a_atom2_mid, a_atom3_mid)
```

```
    :
427         angle_type_mid.append(a_type_mid[i])
428         angle_atom1_mid.append(a_atom1_mid[j])
429         angle_atom2_mid.append(a_atom2_mid[k])
430         angle_atom3_mid.append(a_atom3_mid[l])
431
432 for m, n, o, p in zip(a_type_4end, a_atom1_4end, a_atom2_4end,
    a_atom3_4end):
433         angle_type_4end.append(a_type_4end[m])
434         angle_atom1_4end.append(a_atom1_4end[n])
435         angle_atom2_4end.append(a_atom2_4end[o])
436         angle_atom3_4end.append(a_atom3_4end[p])
437
438 #Write to File
439 with open('polymer_moltemp', 'a') as f:
440         f.write('\n' + '\n' + 'Angles                    #angles [ID type
              atom1 atom2 atom3]' + '\n')
441
442 #monomers
443 if chain_length == 1:
444         with open('polymer_moltemp', 'a') as f:
445                 for i in range(len(angle_type_full)):
446                         f.write('\n' + str(i + 1) + ' ' + str(
                            angle_type_full[i]) + ' ' + str(
                            angle_atom1_full[i]) + ' ' + str(
                            angle_atom2_full[i]) + ' ' + str(
                            angle_atom3_full[i]))
447
448 #dimers and chains
449 else:
450     with open('polymer_moltemp', 'a') as f:
451         for i in range(len(angle_type_1end)):
452             f.write('\n' + str(i + 1) + ' ' + str(angle_type_1end[i]) +
                    ' ' + str(angle_atom1_1end[i]) + ' ' + str(
                    angle_atom2_1end[i]) + ' ' + str(angle_atom3_1end[i]))
453             i += 1
454         f.write('\n' + str(i + 1) + ' ' + str(8)  + ' ' + str(6) + ' ' +
```

```
            str(7)  + ' ' + str(22))
        f.write('\n' + str(i + 2) + ' ' + str(13) + ' ' + str(7) + ' ' +
            str(22) + ' ' + str(23))
        f.write('\n' + str(i + 3) + ' ' + str(12) + ' ' + str(7) + ' ' +
            str(22) + ' ' + str(30))
        f.write('\n' + str(i + 4) + ' ' + str(13) + ' ' + str(7) + ' ' +
            str(22) + ' ' + str(35))

        j = 42
        l = 21
        for k in range(chain_length - 2):
            for i in range(len(angle_type_mid)):
                f.write('\n' + str(j) + ' ' + str(angle_type_mid[i]) + '
                    ' + str(angle_atom1_mid[i] + l) + ' ' + str(
                    angle_atom2_mid[i] + l) + ' ' + str(angle_atom3_mid[i
                    ] + l))
                j += 1
            f.write('\n' + str(j)     + ' ' + str(8)  + ' ' + str(6 + l)
                + ' ' + str(7 + l)      + ' ' + str(1 + l + 19))
            f.write('\n' + str(j + 1) + ' ' + str(13) + ' ' + str(7 + l)
                + ' ' + str(1 + l + 19) + ' ' + str(2  + l + 19))
            f.write('\n' + str(j + 2) + ' ' + str(12) + ' ' + str(7 + l)
                + ' ' + str(1 + l + 19) + ' ' + str(9  + l + 19))
            f.write('\n' + str(j + 3) + ' ' + str(13) + ' ' + str(7 + l)
                + ' ' + str(1 + l + 19) + ' ' + str(14 + l + 19))

            j += 4
            l += 19
        for i in range(len(angle_type_4end)):
                f.write('\n' + str(j) + ' ' + str(angle_type_4end[i]) +
                    ' ' + str(angle_atom1_4end[i] + l) + ' ' + str(
                    angle_atom2_4end[i] + l) + ' ' + str(angle_atom3_4end
                    [i] + l))
                j += 1

##################
#Step 7: Dihedrals
```

```
478 ##################
479
480 #load necessary pickled lists
481 d_type_full  = pickle.load(open('dihed_type_full.p',  'rb'))
482 d_atom1_full = pickle.load(open('dihed_atom1_full.p', 'rb'))
483 d_atom2_full = pickle.load(open('dihed_atom2_full.p', 'rb'))
484 d_atom3_full = pickle.load(open('dihed_atom3_full.p', 'rb'))
485 d_atom4_full = pickle.load(open('dihed_atom4_full.p', 'rb'))
486
487 d_type_1end  = pickle.load(open('dihed_type_1end.p',  'rb'))
488 d_atom1_1end = pickle.load(open('dihed_atom1_1end.p', 'rb'))
489 d_atom2_1end = pickle.load(open('dihed_atom2_1end.p', 'rb'))
490 d_atom3_1end = pickle.load(open('dihed_atom3_1end.p', 'rb'))
491 d_atom4_1end = pickle.load(open('dihed_atom4_1end.p', 'rb'))
492
493 d_type_mid   = pickle.load(open('dihed_type_mid.p',   'rb'))
494 d_atom1_mid  = pickle.load(open('dihed_atom1_mid.p',  'rb'))
495 d_atom2_mid  = pickle.load(open('dihed_atom2_mid.p',  'rb'))
496 d_atom3_mid  = pickle.load(open('dihed_atom3_mid.p',  'rb'))
497 d_atom4_mid  = pickle.load(open('dihed_atom4_mid.p',  'rb'))
498
499 d_type_4end  = pickle.load(open('dihed_type_4end.p',  'rb'))
500 d_atom1_4end = pickle.load(open('dihed_atom1_4end.p', 'rb'))
501 d_atom2_4end = pickle.load(open('dihed_atom2_4end.p', 'rb'))
502 d_atom3_4end = pickle.load(open('dihed_atom3_4end.p', 'rb'))
503 d_atom4_4end = pickle.load(open('dihed_atom4_4end.p', 'rb'))
504
505 #create empty lists for pickled values
506 dihed_type_full, dihed_atom1_full, dihed_atom2_full, dihed_atom3_full,
      dihed_atom4_full, dihed_type_1end, dihed_atom1_1end, dihed_atom2_1end
      , dihed_atom3_1end, dihed_atom4_1end, dihed_type_mid, dihed_atom1_mid
      , dihed_atom2_mid, dihed_atom3_mid, dihed_atom4_mid, dihed_type_4end,
       dihed_atom1_4end, dihed_atom2_4end, dihed_atom3_4end,
      dihed_atom4_4end = [], [], [], [], [], [], [], [], [], [], [], [],
      [], [], [], [], [], [], []
507
508 #create lists of necessary values
```

```
509 for a, b, c, d, e in zip(d_type_full, d_atom1_full, d_atom2_full,
        d_atom3_full, d_atom4_full):
510         dihed_type_full.append(d_type_full[a])
511         dihed_atom1_full.append(d_atom1_full[b])
512         dihed_atom2_full.append(d_atom2_full[c])
513         dihed_atom3_full.append(d_atom3_full[d])
514         dihed_atom4_full.append(d_atom4_full[e])
515
516 for f, g, h, i, j in zip(d_type_1end, d_atom1_1end, d_atom2_1end,
        d_atom3_1end, d_atom4_1end):
517         dihed_type_1end.append(d_type_1end[f])
518         dihed_atom1_1end.append(d_atom1_1end[g])
519         dihed_atom2_1end.append(d_atom2_1end[h])
520         dihed_atom3_1end.append(d_atom3_1end[i])
521         dihed_atom4_1end.append(d_atom4_1end[j])
522
523 for k, l, m, n, o in zip(d_type_mid, d_atom1_mid, d_atom2_mid,
        d_atom3_mid, d_atom4_mid):
524         dihed_type_mid.append(d_type_mid[k])
525         dihed_atom1_mid.append(d_atom1_mid[l])
526         dihed_atom2_mid.append(d_atom2_mid[m])
527         dihed_atom3_mid.append(d_atom3_mid[n])
528         dihed_atom4_mid.append(d_atom4_mid[o])
529
530 for p, q, r, s, t in zip(d_type_4end, d_atom1_4end, d_atom2_4end,
        d_atom3_4end, d_atom4_4end):
531         dihed_type_4end.append(d_type_4end[p])
532         dihed_atom1_4end.append(d_atom1_4end[q])
533         dihed_atom2_4end.append(d_atom2_4end[r])
534         dihed_atom3_4end.append(d_atom3_4end[s])
535         dihed_atom4_4end.append(d_atom4_4end[t])
536
537 #Write to file
538 with open('polymer_moltemp', 'a') as f:
539         f.write('\n' + '\n' + 'Dihedrals                          #
            dihedrals [ID type atom1 atom2 atom3 atom4]' + '\n')
540
```

```
#monomers
if chain_length == 1:
        with open('polymer_moltemp', 'a') as f:
                for i in range(len(dihed_type_full)):
                        f.write('\n' + str(i + 1) + ' ' + str(
                            dihed_type_full[i]) + ' ' + str(
                            dihed_atom1_full[i]) + ' ' + str(
                            dihed_atom2_full[i]) + ' ' + str(
                            dihed_atom3_full[i]) + ' ' + str(
                            dihed_atom4_full[i]))

#chains
else:
    with open('polymer_moltemp', 'a') as f:
        for i in range(len(dihed_type_1end)):
            f.write('\n' + str(i + 1) + ' ' + str(dihed_type_1end[i]) +
                ' ' + str(dihed_atom1_1end[i]) + ' ' + str(
                dihed_atom2_1end[i]) + ' ' + str(dihed_atom3_1end[i]) + '
                 ' + str(dihed_atom4_1end[i]))
        f.write('\n' + str(i + 2)  + ' ' + str(6)  + ' ' + str(1 + 21) +
             ' ' + str(7)      + ' ' + str(6)      + ' ' + str(19))
        f.write('\n' + str(i + 3)  + ' ' + str(7)  + ' ' + str(8)      +
             ' ' + str(6)      + ' ' + str(7)      + ' ' + str(1 + 21))
        f.write('\n' + str(i + 4)  + ' ' + str(7)  + ' ' + str(4)      +
             ' ' + str(6)      + ' ' + str(7)      + ' ' + str(1 + 21))
        f.write('\n' + str(i + 5)  + ' ' + str(14) + ' ' + str(6)      +
             ' ' + str(7)      + ' ' + str(1 + 21) + ' ' + str(13 + 21))
        f.write('\n' + str(i + 6)  + ' ' + str(7)  + ' ' + str(2 + 21) +
             ' ' + str(1 + 21) + ' ' + str(7)      + ' ' + str(6))
        f.write('\n' + str(i + 7)  + ' ' + str(13) + ' ' + str(6)      +
             ' ' + str(7)      + ' ' + str(1 + 21) + ' ' + str(9 + 21))
        f.write('\n' + str(i + 8)  + ' ' + str(13) + ' ' + str(8 + 21) +
             ' ' + str(9 + 21) + ' ' + str(1 + 21) + ' ' + str(7))
        f.write('\n' + str(i + 9)  + ' ' + str(10) + ' ' + str(7)      +
             ' ' + str(1 + 21) + ' ' + str(2 + 21) + ' ' + str(14 + 21))
        f.write('\n' + str(i + 10) + ' ' + str(9)  + ' ' + str(3 + 21) +
             ' ' + str(2 + 21) + ' ' + str(1 + 21) + ' ' + str(7))
```

```
561         f.write('\n' + str(i + 11) + ' ' + str(11) + ' ' + str(4 + 21) +
                ' ' + str(2 + 21) + ' ' + str(1 + 21) + ' ' + str(7))
562
563         j = i + 12
564         l = 21
565         for k in range(chain_length - 2):
566             for i in range(len(dihed_type_mid)):
567                 f.write('\n' + str(j) + ' ' + str(dihed_type_mid[i]) + '
                        ' + str(dihed_atom1_mid[i] + l) + ' ' + str(
                        dihed_atom2_mid[i] + l) + ' ' + str(dihed_atom3_mid[i
                        ] + l) + ' ' + str(dihed_atom4_mid[i] + l))
568                 j += 1
569             f.write('\n' + str(j)      + ' ' + str(6)  + ' ' + str(1 + l
                    + 19) + ' ' + str(7 + l)      + ' ' + str(6 + l)      +
                    ' ' + str(19  + l))
570             f.write('\n' + str(j + 1)  + ' ' + str(7)  + ' ' + str(8 + l
                    )      + ' ' + str(6 + l)      + ' ' + str(7 + l)      +
                    ' ' + str(1  + l + 19))
571             f.write('\n' + str(j + 2)  + ' ' + str(7)  + ' ' + str(4 + l
                    )      + ' ' + str(6 + l)      + ' ' + str(7 + l)      +
                    ' ' + str(1  + l + 19))
572             f.write('\n' + str(j + 3)  + ' ' + str(14) + ' ' + str(6 + l
                    )      + ' ' + str(7 + l)      + ' ' + str(1 + l + 19) +
                    ' ' + str(13 + l + 19))
573             f.write('\n' + str(j + 4)  + ' ' + str(7)  + ' ' + str(2 + l
                    + 19) + ' ' + str(1 + l + 19) + ' ' + str(7 + l)      +
                    ' ' + str(6  + l))
574             f.write('\n' + str(j + 5)  + ' ' + str(13) + ' ' + str(6 + l
                    )      + ' ' + str(7 + l)      + ' ' + str(1 + l + 19) +
                    ' ' + str(9  + l + 19))
575             f.write('\n' + str(j + 6)  + ' ' + str(13) + ' ' + str(8 + l
                    + 19) + ' ' + str(9 + l + 19) + ' ' + str(1 + l + 19) +
                    ' ' + str(7  + l))
576             f.write('\n' + str(j + 7)  + ' ' + str(10) + ' ' + str(7 + l
                    )      + ' ' + str(1 + l + 19) + ' ' + str(2 + l + 19) +
                    ' ' + str(14 + l + 19))
577             f.write('\n' + str(j + 8)  + ' ' + str(9)  + ' ' + str(3 + l
```

```
                + 19) + ' ' + str(2 + l + 19) + ' ' + str(1 + l + 19) +
                ' ' + str(7  + l))
578             f.write('\n' + str(j + 9)  + ' ' + str(11) + ' ' + str(4 + l
                + 19) + ' ' + str(2 + l + 19) + ' ' + str(1 + l + 19) +
                ' ' + str(7  + l))
579
580             j += 10
581             l += 19
582
583         for i in range(len(dihed_type_4end)):
584             f.write('\n' + str(j) + ' ' + str(dihed_type_4end[i]) + ' '
                + str(dihed_atom1_4end[i] + l) + ' ' + str(
                dihed_atom2_4end[i] + l) + ' ' + str(dihed_atom3_4end[i]
                + l) + ' ' + str(dihed_atom4_4end[i] + l))
585             j += 1
586
587 ###################
588 #Step 8: Impropers
589 ###################
590
591 parameters = pickle.load(open('improp_params.p', 'rb'))
592
593 improper_parameters = []
594
595 for a in parameters:
596     improper_parameters.append(parameters[a])
597
598 #Write to file
599 with open('polymer_moltemp', 'a') as f:
600     f.write('\n' + '\n' + 'Impropers            #impropers [ID type
        atom1 atom2 atom3 atom4]' + '\n')
601
602 #monomers
603 if chain_length == 1:
604     with open('polymer_moltemp', 'a') as f:
605         f.write('\n' + str(1) + ' ' + str(improper_parameters[0]) + ' '
            + str(improper_parameters[1]) + ' ' + str(improper_parameters
```

```
            [2]) + ' ' + str(improper_parameters[3]) + ' ' + str(
            improper_parameters[4]))

606
607 #dimers and chains
608 else:
609     with open('polymer_moltemp', 'a') as f:
610         k = 1
611         f.write('\n' + str(k) + ' ' + str(improper_parameters[0]) + ' '
                + str(improper_parameters[1]) + ' ' + str(improper_parameters
                [2]) + ' ' + str(improper_parameters[3]) + ' ' + str(
                improper_parameters[4]))
612
613         j = 21
614         k += 1
615         for i in range(chain_length - 1):
616             f.write('\n' + str(k) + ' ' + str(improper_parameters[0]) +
                    ' ' + str(improper_parameters[1] + j) + ' ' + str(
                    improper_parameters[2] + j) + ' ' + str(
                    improper_parameters[3] + j) + ' ' + str(
                    improper_parameters[4] + j))
617             j += 19
618             k += 1
```

Listing E.2: Python script that contains and pickles all necessary data for Listing E.1.

```
1  #Pickles for coordinates
2
3  import pickle
4
5  #############
6  #COORDINATES
7  #############
8
9  #Atom Types
10 atom_types = {"C1": 1, "C2": 1, "O2": 4, "C3": 1, "O3": 4, "C4": 1, "O4"
       : 4, "C5": 1, "Or": 5, "C6": 2, "O(COO2)": 3, "O(COO1)": 3, "O1": 4,
       "H1": 8, "H2": 7, "H(OH2)": 6, "H3": 7, "H(OH3)": 6, "H4": 7, "H(OH4)
       ": 6, "H5": 7, "H(OH1)": 6}
```

```
11
12 pickle.dump(atom_types, open('atom_types.p', 'wb'))
13
14 ##############
15
16 #M
17 #M full monomer
18 M_x_coords = {"C1": 1.353, "C2": 1.965, "O2": 2.080, "C3": 1.058, "O3":
       1.583, "C4": -0.346, "O4": -1.219, "C5": -0.868, "Or": 0.036, "C6":
       -2.226, "O(COO2)": -2.380, "O(COO1)": -3.265, "O1": 2.164, "H1":
       1.304, "H2": 2.952, "H(OH2)": 2.638, "H3": 1.010, "H(OH3)": 2.472, "
       H4": -0.306, "H(OH4)": -2.127, "H5": -0.947, "H(OH1)": 1.835}
19
20 M_y_coords = {"C1": 1.165, "C2": -0.110, "O2": 0.023, "C3": -1.300, "O3"
       : -2.484, "C4": -1.025, "O4": -2.096, "C5": 0.286, "Or": 1.345, "C6":
        0.594, "O(COO2)": 1.570, "O(COO1)": -0.216, "O1": 2.288, "H1":
       1.079, "H2": -0.273, "H(OH2)": 0.762, "H3": -1.432, "H(OH3)": -2.718,
        "H4": -0.942, "H(OH4)": -1.988, "H5": 0.188, "H(OH1)": 3.132}
21
22 M_z_coords = {"C1": -0.692, "C2": -0.106, "O2": 1.312, "C3": -0.437, "O3
       ": 0.169, "C4": 0.109, "O4": -0.257, "C5": -0.486, "Or": -0.167, "C6"
       : 0.092, "O(COO2)": 0.787, "O(COO1)": -0.167, "O1": -0.339, "H1":
       -1.777, "H2": -0.538, "H(OH2)": 1.593, "H3": -1.518, "H(OH3)":
       -0.131, "H4": 1.195, "H(OH4)": 0.056, "H5": -1.568, "H(OH1)": -0.677}
23
24 pickle.dump(M_x_coords, open("M_x_coords.p", "wb"))
25 pickle.dump(M_y_coords, open("M_y_coords.p", "wb"))
26 pickle.dump(M_z_coords, open("M_z_coords.p", "wb"))
27
28 ###############
29
30 #G
31 #G full monomer
32 G_x_coords = {"C1": -1.185, "C2": -1.774, "O2": -3.192, "C3": -1.165, "
       O3": -1.525, "C4": 0.359, "O4": 0.721, "C5": 0.866, "Or": 0.238, "C6"
       : 2.360, "O(COO2)": 3.166, "O(COO1)": 2.832, "O1": -1.536, "H1":
       -1.582, "H2": -1.543, "H(OH2)": -3.643, "H3": -1.540, "H(OH3)":
```

```
    -2.478, "H4": 0.804, "H(OH4)": 1.674, "H5": 0.626, "H(OH1)": -1.195}
33
34  G_y_coords = {"C1": 1.216, "C2": -0.158, "O2": -0.125, "C3": -1.204, "O3
    ": -0.902, "C4": -1.176, "O4": -1.542, "C5": 0.237, "Or": 1.164, "C6"
    : 0.287, "O(COO2)": -0.429, "O(COO1)": 0.973, "O1": 1.577, "H1":
    1.957, "H2": -0.416, "H(OH2)": 0.521, "H3": -2.193, "H(OH3)": -0.895,
     "H4": -1.880, "H(OH4)": -1.546, "H5": 0.500, "H(OH1)": 2.439}
35
36  G_z_coords = {"C1": 0.417, "C2": 0.746, "O2": 0.569, "C3": -0.193, "O3":
     -1.543, "C4": -0.050, "O4": 1.283, "C5": -0.348, "Or": 0.540, "C6":
    -0.152, "O(COO2)": -0.951, "O(COO1)": 0.724, "O1": -0.920, "H1":
    1.110, "H2": 1.780, "H(OH2)": 1.130, "H3": 0.071, "H(OH3)": -1.704, "
    H4": -0.754, "H(OH4)": 1.445, "H5": -1.378, "H(OH1)": -1.197}
37
38
39  pickle.dump(G_x_coords, open("G_x_coords.p", "wb"))
40  pickle.dump(G_y_coords, open("G_y_coords.p", "wb"))
41  pickle.dump(G_z_coords, open("G_z_coords.p", "wb"))
42
43
44  ########
45  #CHARGES
46  ########
47
48
49  G_charges_full = {"C1": 0.2955, "C2": 0.0973, "O2": -0.603, "C3":
    0.1053, "O3": -0.616, "C4": 0.1193, "O4": -0.6451, "C5": -0.037, "Or"
    : -0.4094, "C6": 0.9404, "O(COO2)": -0.8169, "O(COO1)": -0.8169, "O1"
    : -0.6691, "H1": 0.0648, "H2": 0.0989, "H(OH2)": 0.4158, "H3":
    0.0829, "H(OH3)": 0.4168, "H4": 0.0478, "H(OH4)": 0.4659, "H5":
    0.0348, "H(OH1)": 0.4279}
50
51  M_charges_full = {"C1": 0.2996, "C2": 0.052, "O2": -0.6132, "C3": 0.107,
     "O3": -0.5922, "C4": 0.109, "O4": -0.6282, "C5": -0.0319, "Or":
    -0.4132, "C6": 0.9268, "O(COO2)": -0.8135, "O(COO1)": -0.8135, "O1":
    -0.6012, "H1": 0.0666, "H2": 0.0806, "H(OH2)": 0.4226, "H3": 0.0836,
    "H(OH3)": 0.4016, "H4": 0.0377, "H(OH4)": 0.4545, "H5": 0.0417, "H(
```

```
    OH1)": 0.4236}
52
53  G_charges_1end = {"C1": 0.3364, "C2": 0.0621, "O2": -0.6727, "C3":
        0.081, "O3": -0.6631, "C4": 0.1556, "O4": -0.4136, "C5": -0.0045, "Or
        ": -0.4322, "C6": 0.9759, "O(COO2)": -0.8724, "O(COO1)": -0.8724, "O1
        ": -0.6685, "H1": 0.081, "H2": 0.1247, "H(OH2)": 0.476, "H3": 0.1087,
         "H(OH3)": 0.4186, "H4": 0.0744, "H5": 0.0271, "H(OH1)": 0.4279}
54
55  M_charges_1end = {"C1": 0.2877, "C2": 0.086, "O2": -0.567, "C3": 0.1007,
         "O3": -0.5769, "C4": 0.0942, "O4": -0.4875, "C5": -0.0176, "Or":
        -0.4027, "C6": 0.8774, "O(COO2)": -0.7787, "O(COO1)": -0.7787, "O1":
        -0.5765, "H1": 0.0619, "H2": 0.0699, "H(OH2)": 0.3999, "H3": 0.0427,
        "H(OH3)": 0.4013, "H4": 0.0598, "H5": 0.0464, "H(OH1)": 0.4077}
56
57  G_charges_mid = {"C1": 0.2978, "C2": 0.0978, "O2": -0.6571, "C3":
        0.1117, "O3": -0.6341, "C4": 0.1393, "O4": -0.4038, "C5": 0.0027, "Or
        ": -0.4239, "C6": 0.9578, "O(COO2)": -0.8612, "O(COO1)": -0.8612, "H1
        ": 0.0657, "H2": 0.0768, "H(OH2)": 0.4719, "H3": 0.0792, "H(OH3)":
        0.422, "H4": 0.0795, "H5": 0.0391}
58
59  M_charges_mid = {"C1": 0.1356, "C2": 0.0349, "O2": -0.5894, "C3":
        0.1004, "O3": -0.6073, "C4": 0.1257, "O4": -0.5109, "C5": -0.1093, "
        Or": -0.1871, "C6": 0.9328, "O(COO2)": -0.8099, "O(COO1)": -0.8099, "
        H1": 0.2152, "H2": 0.0796, "H(OH2)": 0.4146, "H3": 0.0552, "H(OH3)":
        0.4056, "H4": 0.067, "H5": 0.0572}
60
61  G_charges_4end = {"C1": 0.2645, "C2": 0.0797, "O2": -0.5755, "C3":
        0.1157, "O3": -0.5605, "C4": 0.1065, "O4": -0.5678, "C5": -0.0005, "
        Or": -0.3528, "C6": 0.8175, "O(COO2)": -0.7505, "O(COO1)": -0.7505, "
        H1": 0.0442, "H2": 0.0815, "H(OH2)": 0.4253, "H3": 0.0338, "H(OH3)":
        0.375, "H4": 0.0301, "H(OH4)": 0.3976, "H5": 0.0367}
62
63  M_charges_4end = {"C1": 0.1415, "C2": 0.0046, "O2": -0.6806, "C3":
        0.1272, "O3": -0.6689, "C4": 0.1674, "O4": -0.71, "C5": -0.1237, "Or"
        : -0.2012, "C6": 1.0434, "O(COO2)": -0.9038, "O(COO1)": -0.9038, "H1"
        : 0.2406, "H2": 0.0833, "H(OH2)": 0.4651, "H3": 0.061, "H(OH3)":
        0.4494, "H4": 0.0867, "H(OH4)": 0.5086, "H5": 0.0632}
```

```
64
65 pickle.dump(M_charges_full, open('M_charges_full.p', 'wb'))
66 pickle.dump(G_charges_full, open('G_charges_full.p', 'wb'))
67 pickle.dump(M_charges_1end, open('M_charges_1end.p', 'wb'))
68 pickle.dump(G_charges_1end, open('G_charges_1end.p', 'wb'))
69 pickle.dump(M_charges_mid,  open('M_charges_mid.p', 'wb'))
70 pickle.dump(G_charges_mid,  open('G_charges_mid.p', 'wb'))
71 pickle.dump(M_charges_4end, open('M_charges_4end.p', 'wb'))
72 pickle.dump(G_charges_4end, open('G_charges_4end.p', 'wb'))
73
74 ######
75 #BONDS
76 ######
77
78 #bond_type = {1: 1, 2: 1, 3: 1, 4: 1, 5: 2, 6: 3, 7: 3, 8: 4, 9: 4, 10:
       4, 11: 4, 12: 5, 13: 5, 14: 6, 15: 6, 16: 6, 17: 6, 18: 6, 19: 7, 20:
        7, 21: 7, 22: 7}
79 #bond_atom1 = {1: 1, 2: 2, 3: 4, 4: 6, 5: 8, 6: 10, 7: 10, 8: 1, 9: 2,
       10: 4, 11: 6, 12: 1, 13: 8, 14: 2, 15: 4, 16: 6, 17: 8, 18: 1, 19: 3,
        20: 5, 21: 7, 22: 13}
80 #bond_atom2 = {1: 2, 2: 4, 3: 6, 4: 8, 5: 10, 6: 11, 7: 12, 8: 13, 9: 3,
        10: 5, 11: 7, 12: 9, 13: 9, 14: 15, 15: 17, 16: 19, 17: 21, 18: 14,
       19: 16, 20: 18, 21: 20, 22: 22}
81
82 #pickle.dump(bond_type, open('bond_type.p', 'wb'))
83 #pickle.dump(bond_atom1, open('bond_atom1.p', 'wb'))
84 #pickle.dump(bond_atom2, open('bond_atom2.p', 'wb'))
85
86 #Full monomer
87 bond_type_full = {1: 1, 2: 1, 3: 1, 4: 1, 5: 2, 6: 3, 7: 3, 8: 4, 9: 4,
       10: 4, 11: 4, 12: 5, 13: 5, 14: 6, 15: 6, 16: 6, 17: 6, 18: 6, 19: 7,
        20: 7, 21: 7, 22: 7}
88 bond_atom1_full = {1: 1, 2: 2, 3: 4, 4: 6, 5: 8, 6: 10, 7: 10, 8: 1, 9:
       2, 10: 4, 11: 6, 12: 1, 13: 8, 14: 2, 15: 4, 16: 6, 17: 8, 18: 1, 19:
        3, 20: 5, 21: 7, 22: 13}
89 bond_atom2_full = {1: 2, 2: 4, 3: 6, 4: 8, 5: 10, 6: 11, 7: 12, 8: 13,
       9: 3, 10: 5, 11: 7, 12: 9, 13: 9, 14: 15, 15: 17, 16: 19, 17: 21, 18:
```

```
    14, 19: 16, 20: 18, 21: 20, 22: 22}
90
91  pickle.dump(bond_type_full, open('bond_type_full.p', 'wb'))
92  pickle.dump(bond_atom1_full, open('bond_atom1_full.p', 'wb'))
93  pickle.dump(bond_atom2_full, open('bond_atom2_full.p', 'wb'))
94
95  #1'-end monomer
96  bond_type_1end = {1: 1, 2: 1, 3: 1, 4: 1, 5: 2, 6: 3, 7: 3, 8: 4, 9: 4,
        10: 4, 11: 4, 12: 5, 13: 5, 14: 6, 15: 6, 16: 6, 17: 6, 18: 6, 19: 7,
        20: 7, 21: 7}
97  bond_atom1_1end = {1: 1, 2: 2, 3: 4, 4: 6, 5: 8, 6: 10, 7: 10, 8: 1, 9:
        2, 10: 4, 11: 6, 12: 1, 13: 8, 14: 2, 15: 4, 16: 6, 17: 8, 18: 1, 19:
        3, 20: 5, 21: 13}
98  bond_atom2_1end = {1: 2, 2: 4, 3: 6, 4: 8, 5: 10, 6: 11, 7: 12, 8: 13,
        9: 3, 10: 5, 11: 7, 12: 9, 13: 9, 14: 15, 15: 17, 16: 19, 17: 20, 18:
        14, 19: 16, 20: 18, 21: 21}
99
100 pickle.dump(bond_type_1end, open('bond_type_1end.p', 'wb'))
101 pickle.dump(bond_atom1_1end, open('bond_atom1_1end.p', 'wb'))
102 pickle.dump(bond_atom2_1end, open('bond_atom2_1end.p', 'wb'))
103
104 #Middle monomer
105
106 bond_type_mid = {1: 1, 2: 1, 3: 1, 4: 1, 5: 2, 6: 3, 7: 3, 8: 4, 9: 4,
        10: 4, 11: 5, 12: 5, 13: 6, 14: 6, 15: 6, 16: 6, 17: 6, 18: 7, 19: 7}
107 bond_atom1_mid = {1: 1, 2: 2, 3: 4, 4: 6, 5: 8, 6: 10, 7: 10, 8: 2, 9:
        4, 10: 6, 11: 1, 12: 8, 13: 2, 14: 4, 15: 6, 16: 8, 17: 1, 18: 3, 19:
        5}
108 bond_atom2_mid = {1: 2, 2: 4, 3: 6, 4: 8, 5: 10, 6: 11, 7: 12, 8: 3, 9:
        5, 10: 7, 11: 9, 12: 9, 13: 14, 14: 16, 15: 18, 16: 19, 17: 13, 18:
        15, 19: 17}
109
110 pickle.dump(bond_type_mid, open('bond_type_mid.p', 'wb'))
111 pickle.dump(bond_atom1_mid, open('bond_atom1_mid.p', 'wb'))
112 pickle.dump(bond_atom2_mid, open('bond_atom2_mid.p', 'wb'))
113
114 #4'-end monomer
```

```
115
116 bond_type_4end = {1: 1, 2: 1, 3: 1, 4: 1, 5: 2, 6: 3, 7: 3, 8: 4, 9: 4,
        10: 4, 11: 5, 12: 5, 13: 6, 14: 6, 15: 6, 16: 6, 17: 6, 18: 7, 19: 7,
         20: 7}
117 bond_atom1_4end = {1: 1, 2: 2, 3: 4, 4: 6, 5: 8, 6: 10, 7: 10, 8: 2, 9:
        4, 10: 6, 11: 1, 12: 8, 13: 2, 14: 4, 15: 6, 16: 8, 17: 1, 18: 3, 19:
         5, 20: 7}
118 bond_atom2_4end = {1: 2, 2: 4, 3: 6, 4: 8, 5: 10, 6: 11, 7: 12, 8: 3, 9:
         5, 10: 7, 11: 9, 12: 9, 13: 14, 14: 16, 15: 18, 16: 20, 17: 13, 18:
        15, 19: 17, 20: 19}
119
120 pickle.dump(bond_type_4end, open('bond_type_4end.p', 'wb'))
121 pickle.dump(bond_atom1_4end, open('bond_atom1_4end.p', 'wb'))
122 pickle.dump(bond_atom2_4end, open('bond_atom2_4end.p', 'wb'))
123
124 #######
125 #ANGLES
126 #######
127
128 #Full monomer
129
130 angle_type_full = {1: 1, 2: 1, 3: 1, 4: 2, 5: 3, 6: 3, 7: 3, 8: 3, 9: 3,
         10: 3, 11: 3, 12: 4, 13: 4, 14: 5, 15: 5, 16: 5, 17: 5, 18: 5, 19:
        5, 20: 5, 21: 5, 22: 6, 23: 6, 24: 7, 25: 7, 26: 7, 27: 7, 28: 8, 29:
         9, 30: 10, 31: 11, 32: 12, 33: 13, 34: 13, 35: 13, 36: 13, 37: 13,
        38: 13}
131 angle_atom1_full = {1: 1, 2: 2, 3: 4, 4: 6, 5: 2, 6: 1, 7: 4, 8: 2, 9:
        6, 10: 4, 11: 8, 12: 2, 13: 6, 14: 1, 15: 4, 16: 2, 17: 6, 18: 4, 19:
         8, 20: 6, 21: 2, 22: 8, 23: 8, 24: 2, 25: 4, 26: 6, 27: 1, 28: 1,
        29: 10, 30: 10, 31: 11, 32: 13, 33: 3, 34: 5, 35: 7, 36: 13, 37: 7,
        38: 9}
132 angle_atom2_full = {1: 2, 2: 4, 3: 6, 4: 8, 5: 1, 6: 2, 7: 2, 8: 4, 9:
        4, 10: 6, 11: 6, 12: 1, 13: 8, 14: 2, 15: 2, 16: 4, 17: 4, 18: 6, 19:
         6, 20: 8, 21: 1, 22: 10, 23: 10, 24: 3, 25: 5, 26: 7, 27: 13, 28: 9,
         29: 8, 30: 8, 31: 10, 32: 1, 33: 2, 34: 4, 35: 6, 36: 1, 37: 8, 38:
        1}
133 angle_atom3_full = {1: 4, 2: 6, 3: 8, 4: 10, 5: 13, 6: 3, 7: 3, 8: 5, 9:
```

```
    5, 10: 7, 11: 7, 12: 9, 13: 9, 14: 15, 15: 15, 16: 17, 17: 17, 18:
    19, 19: 19, 20: 21, 21: 14, 22: 11, 23: 12, 24: 16, 25: 18, 26: 20,
    27: 22, 28: 8, 29: 9, 30: 21, 31: 12, 32: 9, 33: 15, 34: 17, 35: 19,
    36: 14, 37: 21, 38: 14}
134
135 pickle.dump(angle_type_full, open('angle_type_full.p', 'wb'))
136 pickle.dump(angle_atom1_full, open('angle_atom1_full.p', 'wb'))
137 pickle.dump(angle_atom2_full, open('angle_atom2_full.p', 'wb'))
138 pickle.dump(angle_atom3_full, open('angle_atom3_full.p', 'wb'))
139
140 #1'end monomer
141
142 angle_type_1end = {1: 1, 2: 1, 3: 1, 4: 2, 5: 3, 6: 3, 7: 3, 8: 3, 9: 3,
    10: 3, 11: 3, 12: 4, 13: 4, 14: 5, 15: 5, 16: 5, 17: 5, 18: 5, 19:
    5, 20: 5, 21: 5, 22: 6, 23: 6, 24: 7, 25: 7, 26: 7, 27: 8, 28: 9, 29:
    10, 30: 11, 31: 12, 32: 13, 33: 13, 34: 13, 35: 13, 36: 13, 37: 13}
143 angle_atom1_1end = {1: 1, 2: 2, 3: 4, 4: 6, 5: 2, 6: 1, 7: 4, 8: 2, 9:
    6, 10: 4, 11: 8, 12: 2, 13: 6, 14: 1, 15: 4, 16: 2, 17: 6, 18: 4, 19:
    8, 20: 6, 21: 2, 22: 8, 23: 8, 24: 2, 25: 4, 26: 1, 27: 1, 28: 10,
    29: 10, 30: 11, 31: 13, 32: 3, 33: 5, 34: 7, 35: 13, 36: 7, 37: 9}
144 angle_atom2_1end = {1: 2, 2: 4, 3: 6, 4: 8, 5: 1, 6: 2, 7: 2, 8: 4, 9:
    4, 10: 6, 11: 6, 12: 1, 13: 8, 14: 2, 15: 2, 16: 4, 17: 4, 18: 6, 19:
    6, 20: 8, 21: 1, 22: 10, 23: 10, 24: 3, 25: 5, 26: 13, 27: 9, 28: 8,
    29: 8, 30: 10, 31: 1, 32: 2, 33: 4, 34: 6, 35: 1, 36: 8, 37: 1}
145 angle_atom3_1end = {1: 4, 2: 6, 3: 8, 4: 10, 5: 13, 6: 3, 7: 3, 8: 5, 9:
    5, 10: 7, 11: 7, 12: 9, 13: 9, 14: 15, 15: 15, 16: 17, 17: 17, 18:
    19, 19: 19, 20: 20, 21: 14, 22: 11, 23: 12, 24: 16, 25: 18, 26: 21,
    27: 8, 28: 9, 29: 20, 30: 12, 31: 9, 32: 15, 33: 17, 34: 19, 35: 14,
    36: 20, 37: 14}
146
147 pickle.dump(angle_type_1end, open('angle_type_1end.p', 'wb'))
148 pickle.dump(angle_atom1_1end, open('angle_atom1_1end.p', 'wb'))
149 pickle.dump(angle_atom2_1end, open('angle_atom2_1end.p', 'wb'))
150 pickle.dump(angle_atom3_1end, open('angle_atom3_1end.p', 'wb'))
151
152 #Middle monomer
153
```

```
154 angle_type_mid = {1: 1, 2: 1, 3: 1, 4: 2, 5: 3, 6: 3, 7: 3, 8: 3, 9: 3,
        10: 3, 11: 4, 12: 4, 13: 5, 14: 5, 15: 5, 16: 5, 17: 5, 18: 5, 19: 5,
         20: 5, 21: 6, 22: 6, 23: 7, 24: 7, 25: 8, 26: 9, 27: 10, 28: 11, 29:
         13, 30: 13, 31: 13, 32: 13, 33: 13}
155 angle_atom1_mid = {1: 1, 2: 2, 3: 4, 4: 6, 5: 1, 6: 4, 7: 2, 8: 6, 9: 4,
         10: 8, 11: 2, 12: 6, 13: 1, 14: 4, 15: 2, 16: 6, 17: 4, 18: 8, 19:
        6, 20: 2, 21: 8, 22: 8, 23: 2, 24: 4, 25: 1, 26: 10, 27: 10, 28: 11,
        29: 3, 30: 5, 31: 7, 32: 7, 33: 9}
156 angle_atom2_mid = {1: 2, 2: 4, 3: 6, 4: 8, 5: 2, 6: 2, 7: 4, 8: 4, 9: 6,
         10: 6, 11: 1, 12: 8, 13: 2, 14: 2, 15: 4, 16: 4, 17: 6, 18: 6, 19:
        8, 20: 1, 21: 10, 22: 10, 23: 3, 24: 5, 25: 9, 26: 8, 27: 8, 28: 10,
        29: 2, 30: 4, 31: 6, 32: 8, 33: 1}
157 angle_atom3_mid = {1: 4, 2: 6, 3: 8, 4: 10, 5: 3, 6: 3, 7: 5, 8: 5, 9:
        7, 10: 7, 11: 9, 12: 9, 13: 14, 14: 14, 15: 16, 16: 16, 17: 18, 18:
        18, 19: 19, 20: 13, 21: 11, 22: 12, 23: 15, 24: 17, 25: 8, 26: 9, 27:
         19, 28: 12, 29: 14, 30: 16, 31: 18, 32: 19, 33: 13}
158
159 pickle.dump(angle_type_mid, open('angle_type_mid.p', 'wb'))
160 pickle.dump(angle_atom1_mid, open('angle_atom1_mid.p', 'wb'))
161 pickle.dump(angle_atom2_mid, open('angle_atom2_mid.p', 'wb'))
162 pickle.dump(angle_atom3_mid, open('angle_atom3_mid.p', 'wb'))
163
164 #4'end monomer
165
166 angle_type_4end = {1: 1, 2: 1, 3: 1, 4: 2, 5: 3, 6: 3, 7: 3, 8: 3, 9: 3,
         10: 3, 11: 4, 12: 4, 13: 5, 14: 5, 15: 5, 16: 5, 17: 5, 18: 5, 19:
        5, 20: 5, 21: 6, 22: 6, 23: 7, 24: 7, 25: 7, 26: 8, 27: 9, 28: 10,
        29: 11, 30: 13, 31: 13, 32: 13, 33: 13, 34: 13}
167 angle_atom1_4end = {1: 1, 2: 2, 3: 4, 4: 6, 5: 1, 6: 4, 7: 2, 8: 6, 9:
        4, 10: 8, 11: 2, 12: 6, 13: 1, 14: 4, 15: 2, 16: 6, 17: 4, 18: 8, 19:
         6, 20: 2, 21: 8, 22: 8, 23: 2, 24: 4, 25: 6, 26: 1, 27: 10, 28: 10,
        29: 11, 30: 3, 31: 5, 32: 7, 33: 7, 34: 9}
168 angle_atom2_4end = {1: 2, 2: 4, 3: 6, 4: 8, 5: 2, 6: 2, 7: 4, 8: 4, 9:
        6, 10: 6, 11: 1, 12: 8, 13: 2, 14: 2, 15: 4, 16: 4, 17: 6, 18: 6, 19:
         8, 20: 1, 21: 10, 22: 10, 23: 3, 24: 5, 25: 7, 26: 9, 27: 8, 28: 8,
        29: 10, 30: 2, 31: 4, 32: 6, 33: 8, 34: 1}
169 angle_atom3_4end = {1: 4, 2: 6, 3: 8, 4: 10, 5: 3, 6: 3, 7: 5, 8: 5, 9:
```

```
    7, 10: 7, 11: 9, 12: 9, 13: 14, 14: 14, 15: 16, 16: 16, 17: 18, 18:
    18, 19: 20, 20: 13, 21: 11, 22: 12, 23: 15, 24: 17, 25: 19, 26: 8,
    27: 9, 28: 20, 29: 12, 30: 14, 31: 16, 32: 18, 33: 20, 34: 13}
170
171 pickle.dump(angle_type_4end, open('angle_type_4end.p', 'wb'))
172 pickle.dump(angle_atom1_4end, open('angle_atom1_4end.p', 'wb'))
173 pickle.dump(angle_atom2_4end, open('angle_atom2_4end.p', 'wb'))
174 pickle.dump(angle_atom3_4end, open('angle_atom3_4end.p', 'wb'))
175
176 ##########
177 #DIHEDRALS
178 ##########
179
180 #Full monomer
181
182 dihed_type_full = {1: 1, 2: 1, 3: 1, 4: 2, 5: 2, 6: 2, 7: 2, 8: 2, 9:
    15, 10: 2, 11: 11, 12: 11, 13: 3, 14: 3, 15: 3, 16: 3, 17: 3, 18: 3,
    19: 12, 20: 12, 21: 4, 22: 4, 23: 4, 24: 4, 25: 4, 25: 4, 26: 4, 27:
    4, 28: 4, 29: 19, 30: 19, 31: 19, 32: 19, 33: 7, 34: 7, 35: 5, 36:
    13, 37: 14, 38: 6, 39: 16, 40: 16, 41: 17, 42: 17, 43: 8, 44: 8, 45:
    8, 46: 9, 47: 9, 48: 10, 49: 10, 50: 10, 51: 10, 52: 10, 53: 10, 54:
    10, 55: 18, 56: 18, 57: 18, 58: 18}
183 dihed_atom1_full = {1: 1, 2: 2, 3: 4, 4: 13, 5: 5, 6: 3, 7: 7, 8: 5, 9:
    7, 10: 19, 11: 9, 12: 9, 13: 17, 14: 15, 15: 19, 16: 17, 17: 21, 18:
    14, 19: 6, 20: 6, 21: 16, 22: 22, 23: 16, 24: 18, 25: 18, 26: 20, 27:
     20, 28: 9, 29: 16, 30: 18, 31: 20, 32: 22, 33: 8, 34: 1, 35: 1, 36:
    13, 37: 14, 38: 21, 39: 9, 40: 9, 41: 21, 42: 12, 43: 13, 44: 3, 45:
    5, 46: 3, 47: 7, 48: 13, 49: 3, 50: 5, 51: 7, 52: 3, 53: 15, 54: 19,
    55: 15, 56: 17, 57: 19, 58: 15}
184 dihed_atom2_full = {1: 2, 2: 4, 3: 6, 4: 1, 5: 4, 6: 2, 7: 6, 8: 4, 9:
    6, 10: 6, 11: 1, 12: 8, 13: 4, 14: 2, 15: 6, 16: 4, 17: 8, 18: 1, 19:
     8, 20: 8, 21: 3, 22: 13, 23: 3, 24: 5, 25: 5, 26: 7, 27: 7, 28: 1,
    29: 3, 30: 5, 31: 7, 32: 13, 33: 9, 34: 9, 35: 8, 36: 1, 37: 1, 38:
    8, 39: 8, 40: 8, 41: 8, 42: 10, 43: 1, 44: 2, 45: 4, 46: 2, 47: 6,
    48: 1, 49: 2, 50: 4, 51: 6, 52: 2, 53: 2, 54: 6, 55: 2, 56: 4, 57: 6,
     58: 2}
185 dihed_atom3_full = {1: 4, 2: 6, 3: 8, 4: 2, 5: 2, 6: 4, 7: 4, 8: 6, 9:
```

```
    8, 10: 8, 11: 2, 12: 6, 13: 2, 14: 4, 15: 4, 16: 6, 17: 6, 18: 2, 19:
     10, 20: 10, 21: 2, 22: 1, 23: 2, 24: 4, 25: 4, 26: 6, 27: 6, 28: 13,
     29: 2, 30: 4, 31: 6, 32: 1, 33: 1, 34: 8, 35: 9, 36: 9, 37: 9, 38:
    9, 39: 10, 40: 10, 41: 10, 42: 8, 43: 2, 44: 4, 45: 6, 46: 1, 47: 8,
    48: 2, 49: 4, 50: 2, 51: 8, 52: 1, 53: 1, 54: 8, 55: 4, 56: 6, 57: 8,
     58: 1}
dihed_atom4_full = {1: 6, 2: 8, 3: 10, 4: 4, 5: 1, 6: 6, 7: 2, 8: 8, 9:
    10, 10: 10, 11: 4, 12: 4, 13: 1, 14: 6, 15: 2, 16: 8, 17: 4, 18: 4,
    19: 11, 20: 12, 21: 1, 22: 2, 23: 4, 24: 2, 25: 6, 26: 4, 27: 8, 28:
    22, 29: 15, 30: 17, 31: 19, 32: 14, 33: 2, 34: 6, 35: 1, 36: 8, 37:
    8, 38: 1, 39: 11, 40: 12, 41: 11, 42: 21, 43: 3, 44: 5, 45: 7, 46: 9,
     47: 9, 48: 15, 49: 17, 50: 15, 51: 21, 52: 14, 53: 9, 54: 9, 55: 17,
     56: 19, 57: 21, 58: 14}

pickle.dump(dihed_type_full, open('dihed_type_full.p', 'wb'))
pickle.dump(dihed_atom1_full, open('dihed_atom1_full.p', 'wb'))
pickle.dump(dihed_atom2_full, open('dihed_atom2_full.p', 'wb'))
pickle.dump(dihed_atom3_full, open('dihed_atom3_full.p', 'wb'))
pickle.dump(dihed_atom4_full, open('dihed_atom4_full.p', 'wb'))

#1'end monomer

dihed_type_1end = {1: 1, 2: 1, 3: 1, 4: 2, 5: 2, 6: 2, 7: 11, 8: 2, 9:
    20, 10: 2, 11: 11, 12: 11, 13: 3, 14: 3, 15: 3, 16: 3, 17: 3, 18: 3,
    19: 12, 20: 12, 21: 4, 22: 4, 23: 4, 24: 4, 25: 4, 26: 4, 27: 19, 28:
     19, 29: 19, 30: 7, 31: 7, 32: 5, 33: 13, 34: 14, 35: 6, 36: 16, 37:
    16, 38: 17, 39: 17, 40: 8, 41: 8, 42: 9, 43: 9, 44: 21, 45: 10, 46:
    10, 47: 10, 48: 10, 49: 10, 50: 10, 51: 10, 52: 18, 53: 18, 54: 18,
    55: 18}
dihed_atom1_1end = {1: 1, 2: 2, 3: 4, 4: 13, 5: 5, 6: 3, 7: 7, 8: 5, 9:
    7, 10: 19, 11: 9, 12: 9, 13: 17, 14: 15, 15: 19, 16: 17, 17: 20, 18:
    14, 19: 6, 20: 6, 21: 16, 22: 21, 23: 16, 24: 18, 25: 18, 26: 9, 27:
    16, 28: 18, 29: 21, 30: 8, 31: 1, 32: 1, 33: 13, 34: 14, 35: 1, 36:
    9, 37: 9, 38: 20, 39: 20, 40: 13, 41: 3, 42: 5, 43: 3, 44: 7, 45: 13,
     46: 3, 47: 5, 48: 20, 49: 3, 50: 15, 51: 19, 52: 15, 53: 17, 54: 19,
     55: 15}
dihed_atom2_1end = {1: 2, 2: 4, 3: 6, 4: 1, 5: 4, 6: 2, 7: 6, 8: 4, 9:
```

```
    6, 10: 6, 11: 1, 12: 8, 13: 4, 14: 2, 15: 6, 16: 4, 17: 8, 18: 1, 19:
     8, 20: 8, 21: 3, 22: 13, 23: 3, 24: 5, 25: 5, 26: 1, 27: 3, 28: 5,
    29: 13, 30: 9, 31: 9, 32: 9, 33: 1, 34: 1, 35: 9, 36: 8, 37: 8, 38:
    8, 39: 8, 40: 1, 41: 2, 42: 4, 43: 2, 44: 6, 45: 1, 46: 2, 47: 4, 48:
     8, 49: 2, 50: 2, 51: 6, 52: 2, 53: 4, 54: 6, 55: 2}
dihed_atom3_1end = {1: 4, 2: 6, 3: 8, 4: 2, 5: 2, 6: 4, 7: 4, 8: 6, 9:
    8, 10: 8, 11: 2, 12: 6, 13: 2, 14: 4, 15: 4, 16: 6, 17: 6, 18: 2, 19:
     10, 20: 10, 21: 2, 22: 1, 23: 2, 24: 4, 25: 4, 26: 13, 27: 2, 28: 4,
     29: 1, 30: 1, 31: 8, 32: 8, 33: 9, 34: 9, 35: 8, 36: 10, 37: 10, 38:
     10, 39: 10, 40: 2, 41: 4, 42: 6, 43: 1, 44: 8, 45: 2, 46: 4, 47: 2,
    48: 6, 49: 1, 50: 1, 51: 8, 52: 4, 53: 6, 54: 8, 55: 1}
dihed_atom4_1end = {1: 6, 2: 8, 3: 10, 4: 4, 5: 1, 6: 6, 7: 2, 8: 8, 9:
    10, 10: 10, 11: 4, 12: 4, 13: 1, 14: 6, 15: 2, 16: 8, 17: 4, 18: 4,
    19: 11, 20: 12, 21: 1, 22: 2, 23: 4, 24: 2, 25: 6, 26: 21, 27: 15,
    28: 17, 29: 14, 30: 2, 31: 6, 32: 10, 33: 8, 34: 8, 35: 20, 36: 11,
    37: 12, 38: 11, 39: 12, 40: 3, 41: 5, 42: 7, 43: 9, 44: 9, 45: 15,
    46: 17, 47: 15, 48: 7, 49: 14, 50: 9, 51: 9, 52: 17, 53: 19, 54: 20,
    55: 14}

pickle.dump(dihed_type_1end, open('dihed_type_1end.p', 'wb'))
pickle.dump(dihed_atom1_1end, open('dihed_atom1_1end.p', 'wb'))
pickle.dump(dihed_atom2_1end, open('dihed_atom2_1end.p', 'wb'))
pickle.dump(dihed_atom3_1end, open('dihed_atom3_1end.p', 'wb'))
pickle.dump(dihed_atom4_1end, open('dihed_atom4_1end.p', 'wb'))

#Middle monomer

dihed_type_mid = {1: 1, 2: 1, 3: 1, 4: 2, 5: 2, 6: 11, 7: 2, 8: 20, 9:
    2, 10: 11, 11: 11, 12: 3, 13: 3, 14: 3, 15: 3, 16: 3, 17: 3, 18: 12,
    19: 12, 20: 4, 21: 4, 22: 4, 23: 4, 24: 19, 25: 19, 26: 7, 27: 7, 28:
     5, 29: 14, 30: 6, 31: 16, 32: 16, 33: 17, 34: 17, 35: 8, 36: 9, 37:
    9, 38: 21, 39: 10, 40: 10, 41: 10, 42: 10, 43: 10, 44: 10, 45: 18,
    46: 18, 47: 18, 48: 18}
dihed_atom1_mid = {1: 1, 2: 2, 3: 4, 4: 5, 5: 3, 6: 7, 7: 5, 8: 10, 9:
    18, 10: 9, 11: 9, 12: 16, 13: 14, 14: 18, 15: 16, 16: 19, 17: 13, 18:
     6, 19: 6, 20: 15, 21: 15, 22: 17, 23: 17, 24: 15, 25: 17, 26: 8, 27:
     1, 28: 1, 29: 13, 30: 19, 31: 9, 32: 9, 33: 19, 34: 19, 35: 3, 36:
```

```
    5, 37: 3, 38: 7, 39: 3, 40: 5, 41: 7, 42: 3, 43: 14, 44: 18, 45: 14,
    46: 16, 47: 18, 48: 14}
dihed_atom2_mid = {1: 2, 2: 4, 3: 6, 4: 4, 5: 2, 6: 6, 7: 4, 8: 8, 9: 6,
     10: 1, 11: 8, 12: 4, 13: 2, 14: 6, 15: 4, 16: 8, 17: 1, 18: 8, 19:
    8, 20: 3, 21: 3, 22: 5, 23: 5, 24: 3, 25: 5, 26: 9, 27: 9, 28: 9, 29:
     1, 30: 8, 31: 8, 32: 8, 33: 8, 34: 8, 35: 2, 36: 4, 37: 2, 38: 6,
    39: 2, 40: 4, 41: 6, 42: 2, 43: 2, 44: 6, 45: 2, 46: 4, 47: 6, 48: 2}
dihed_atom3_mid = {1: 4, 2: 6, 3: 8, 4: 2, 5: 4, 6: 4, 7: 6, 8: 6, 9: 8,
     10: 2, 11: 6, 12: 2, 13: 4, 14: 4, 15: 6, 16: 6, 17: 2, 18: 10, 19:
    10, 20: 2, 21: 2, 22: 4, 23: 4, 24: 2, 25: 4, 26: 1, 27: 8, 28: 8,
    29: 9, 30: 9, 31: 10, 32: 10, 33: 10, 34: 10, 35: 4, 36: 6, 37: 1,
    38: 8, 39: 4, 40: 2, 41: 8, 42: 1, 43: 1, 44: 8, 45: 4, 46: 6, 47: 8,
     48: 1}
dihed_atom4_mid = {1: 6, 2: 8, 3: 10, 4: 1, 5: 6, 6: 2, 7: 8, 8: 7, 9:
    10, 10: 4, 11: 4, 12: 1, 13: 6, 14: 2, 15: 8, 16: 4, 17: 4, 18: 11,
    19: 12, 20: 1, 21: 4, 22: 2, 23: 6, 24: 14, 25: 16, 26: 2, 27: 6, 28:
     10, 29: 8, 30: 1, 31: 11, 32: 12, 33: 11, 34: 12, 35: 5, 36: 7, 37:
    9, 38: 9, 39: 16, 40: 14, 41: 19, 42: 13, 43: 9, 44: 9, 45: 16, 46:
    18, 47: 19, 48: 13}

pickle.dump(dihed_type_mid, open('dihed_type_mid.p', 'wb'))
pickle.dump(dihed_atom1_mid, open('dihed_atom1_mid.p', 'wb'))
pickle.dump(dihed_atom2_mid, open('dihed_atom2_mid.p', 'wb'))
pickle.dump(dihed_atom3_mid, open('dihed_atom3_mid.p', 'wb'))
pickle.dump(dihed_atom4_mid, open('dihed_atom4_mid.p', 'wb'))

#4end monomer

dihed_type_4end = {1: 1, 2: 1, 3: 1, 4: 2, 5: 2, 6: 2, 7: 2, 8: 15, 9:
    2, 10: 11, 11: 11, 12: 3, 13: 3, 14: 3, 15: 3, 16: 3, 17: 3, 18: 12,
    19: 12, 20: 4, 21: 4, 22: 4, 23: 4, 24: 4, 25: 4, 26: 19, 27: 19, 28:
     19, 29: 7, 30: 7, 31: 5, 32: 14, 33: 6, 34: 16, 35: 16, 36: 17, 37:
    17, 38: 8, 39: 8, 40: 9, 41: 9, 42: 10, 43: 10, 44: 10, 45: 10, 46:
    10, 47: 10, 48: 18, 49: 18, 50: 18, 51: 18}
dihed_atom1_4end = {1: 1, 2: 2, 3: 4, 4: 5, 5: 3, 6: 7, 7: 5, 8: 10, 9:
    18, 10: 9, 11: 9, 12: 16, 13: 14, 14: 18, 15: 16, 16: 20, 17: 13, 18:
     6, 19: 6, 20: 15, 21: 15, 22: 17, 23: 17, 24: 19, 25: 19, 26: 15,
```

```
    27: 17, 28: 19, 29: 8, 30: 1, 31: 1, 32: 13, 33: 20, 34: 9, 35: 9,
    36: 20, 37: 20, 38: 3, 39: 5, 40: 3, 41: 7, 42: 3, 43: 5, 44: 7, 45:
    3, 46: 14, 47: 18, 48: 14, 49: 16, 50: 18, 51: 14}
dihed_atom2_4end = {1: 2, 2: 4, 3: 6, 4: 4, 5: 2, 6: 6, 7: 4, 8: 8, 9:
    6, 10: 1, 11: 8, 12: 4, 13: 2, 14: 6, 15: 4, 16: 8, 17: 1, 18: 8, 19:
     8, 20: 3, 21: 3, 22: 5, 23: 5, 24: 7, 25: 7, 26: 3, 27: 5, 28: 7,
    29: 9, 30: 9, 31: 9, 32: 1, 33: 8, 34: 8, 35: 8, 36: 8, 37: 8, 38: 2,
     39: 4, 40: 2, 41: 6, 42: 2, 43: 4, 44: 6, 45: 2, 46: 2, 47: 6, 48:
    2, 49: 4, 50: 6, 51: 2}
dihed_atom3_4end = {1: 4, 2: 6, 3: 8, 4: 2, 5: 4, 6: 4, 7: 6, 8: 6, 9:
    8, 10: 2, 11: 6, 12: 2, 13: 4, 14: 4, 15: 6, 16: 6, 17: 2, 18: 10,
    19: 10, 20: 2, 21: 2, 22: 4, 23: 4, 24: 6, 25: 6, 26: 2, 27: 4, 28:
    6, 29: 1, 30: 8, 31: 8, 32: 9, 33: 9, 34: 10, 35: 10, 36: 10, 37: 10,
     38: 4, 39: 6, 40: 1, 41: 8, 42: 4, 43: 2, 44: 8, 45: 1, 46: 1, 47:
    8, 48: 4, 49: 6, 50: 8, 51: 1}
dihed_atom4_4end = {1: 6, 2: 8, 3: 10, 4: 1, 5: 6, 6: 2, 7: 8, 8: 7, 9:
    10, 10: 4, 11: 4, 12: 1, 13: 6, 14: 2, 15: 8, 16: 4, 17: 4, 18: 11,
    19: 12, 20: 1, 21: 4, 22: 2, 23: 6, 24: 4, 25: 8, 26: 14, 27: 16, 28:
     18, 29: 2, 30: 6, 31: 10, 32: 8, 33: 1, 34: 11, 35: 12, 36: 11, 37:
    12, 38: 5, 39: 7, 40: 9, 41: 9, 42: 16, 43: 14, 44: 20, 45: 13, 46:
    9, 47: 9, 48: 16, 49: 18, 50: 20, 51: 13}

pickle.dump(dihed_type_4end, open('dihed_type_4end.p', 'wb'))
pickle.dump(dihed_atom1_4end, open('dihed_atom1_4end.p', 'wb'))
pickle.dump(dihed_atom2_4end, open('dihed_atom2_4end.p', 'wb'))
pickle.dump(dihed_atom3_4end, open('dihed_atom3_4end.p', 'wb'))
pickle.dump(dihed_atom4_4end, open('dihed_atom4_4end.p', 'wb'))

##########
#IMPROPERS
##########

#All monomers

improper_params = {1: 1, 2: 10, 3: 11, 4: 12, 5: 8}

pickle.dump(improper_params, open('improp_params.p', 'wb'))
```

Appendix F

Bootstrapping methodology for error calculation

Bootstrap methodology is a useful error estimation methodology when only one dataset is available. The methodology uses random sample values from the full dataset, and calculates the value being investigated from these samples. The sample sets can contain multiples of the same value, and are discarded if their variance is zero, i.e. all samples are the same value from the original dataset. The random sample calculated values are then ranked and the lowest and highest values are given as the range in which the value being investigated lies within. It is then common to factor in a confidence level to avoid including outliers.

For the bootstrap methodology used to calculated the diffusion constant and α values from the mean squared displacement data in Chapter 4, 1,000 random sample datasets were extracted and a 95 % confidence level was applied. A 95 % confidence level means that the top and bottom 2.5 % of calculated values were discarded and the new lowest and highest values were presented as the bootstrap error range.

Appendix G

Further results from Chapter 4

This section includes figures and results referenced in Chapter 4.

The grouping methodology of the different oxygen atoms along the trimers is shown in Figure G.1.

The C4-O4-C1 glycosidic angle distributions for all systems studied in Chapter 4 are shown in Figure G.2. Little to no difference is seen between the different systems.

The dihedrals angle between the 4'-end and middle and middle and 1'-end monomers for all trimers which presented with data in more than one quadrant of the dihedral plot and the angle from the first frame of the simulation are given in Figure **??**.

The diffusion coefficients for all systems and species are shown in Figure G.4.

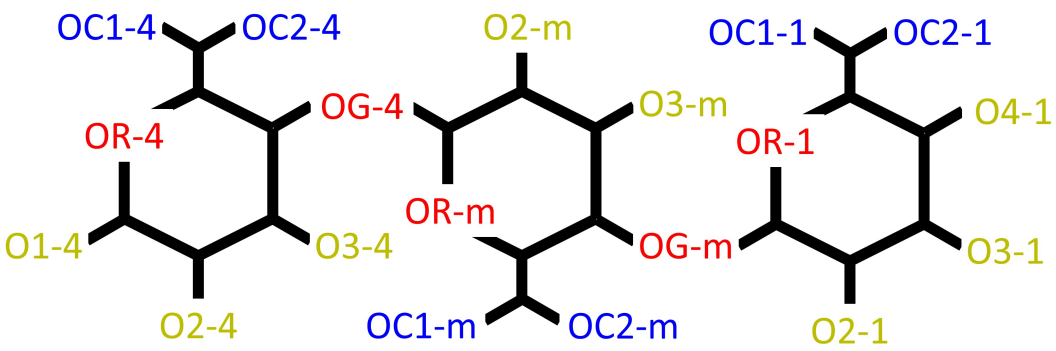


Figure G.1: An alginate trimer with oxygen atoms labelled. The oxygens are split into three groups: (blue) carboxylate, (yellow) hydroxyl, and (red) glycosidic and ring.

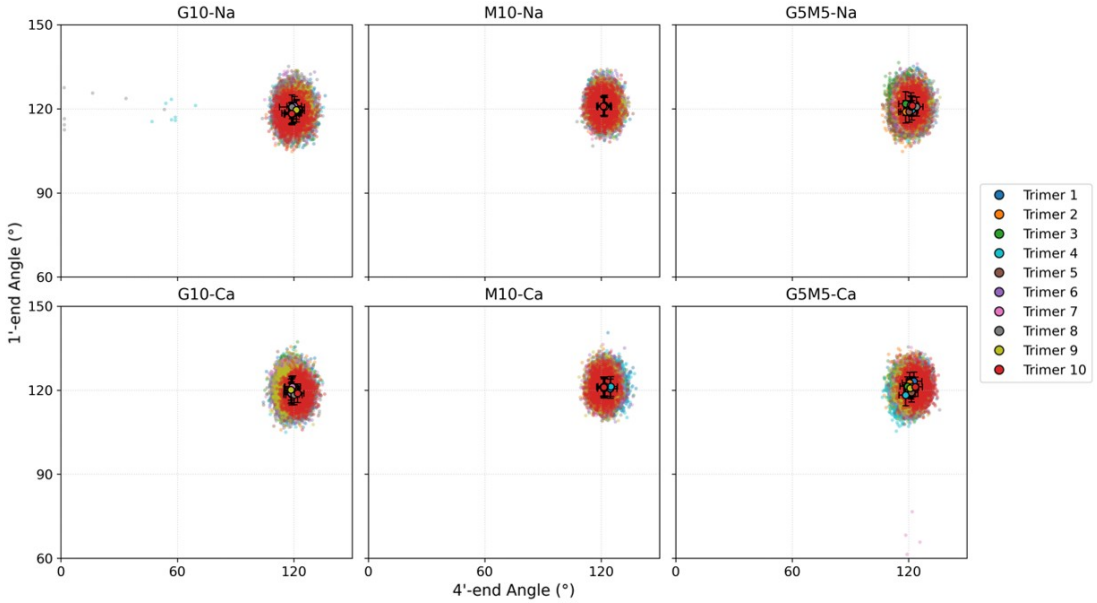


Figure G.2: Glycosidic angle distributions for all systems.

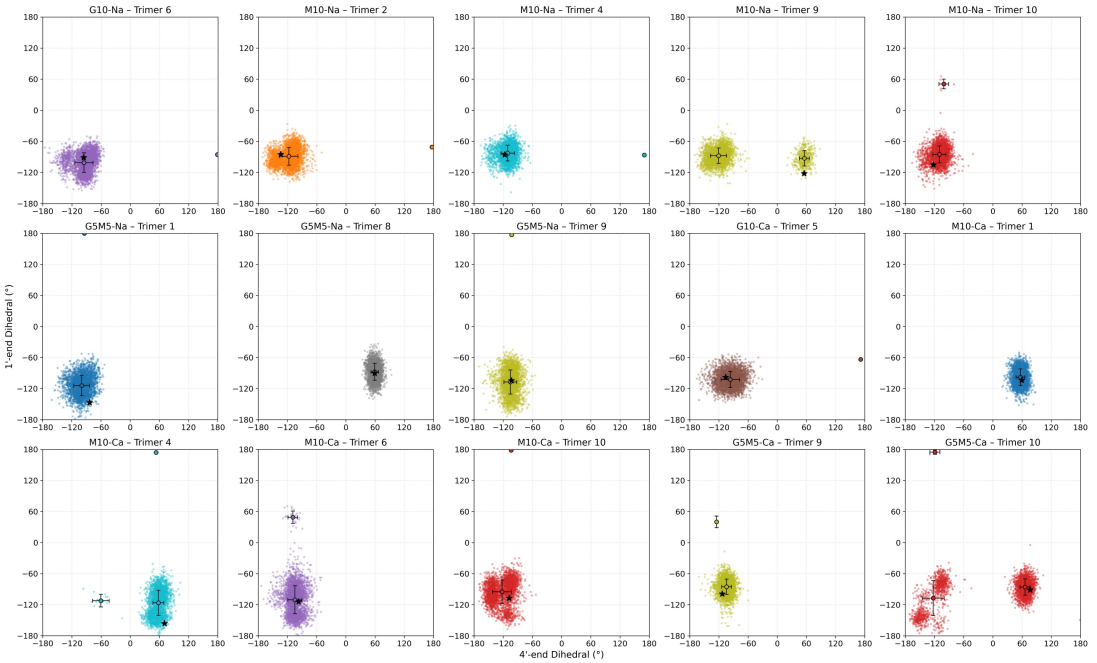


Figure G.3: Glycosidic dihedral angles for all trimers which accessed more than one quadrant during the simulations. The angle which occurred at the first frame of the simulation for each included trimer is also included.

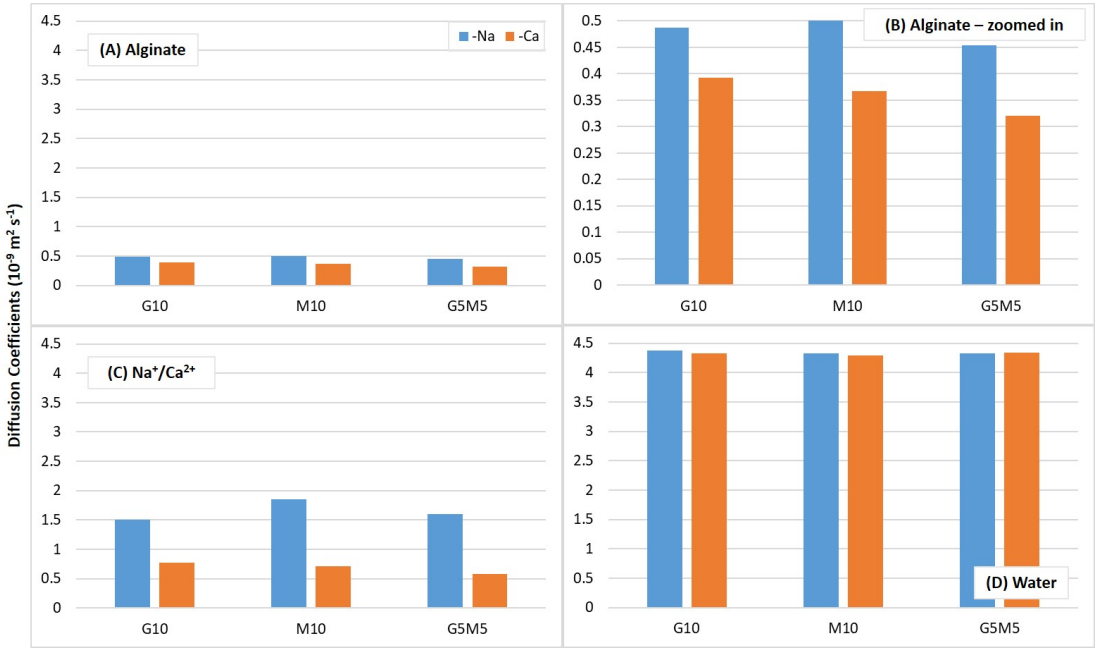


Figure G.4: Diffusion coefficients of the different species in each system found from the gradient of the MSD slopes. Deviation from the averages found through bootstrap resampling at 95 % confidence intervals with 1000 replicates gave value ranges within 2 % of the average; these ranges were hardly visible on the plots and have been removed for simplicity.

Appendix H

Bonded Potentials Comparison in LAMMPS and GROMACS

This section includes examples of the bond, dihedral, and improper energies which were calculated for an alginate trimer in a system with three Na^+ ions in both LAMMPS and GROMACS. Figure H.1 shows the potential of a C-C bond, Figure H.2 shows the potential of a C-C-C-C dihedral, and Figure H.3 shows the potential of the C-C-O-O improper. The unusual shape of the GROMACS C-C bond potential arose due to the GROMACS accuracy and does not reflect on the potential calculated.

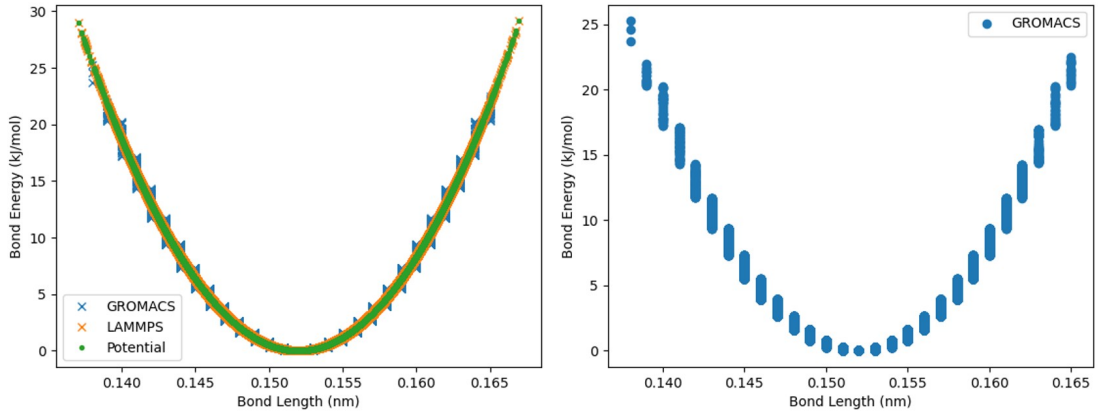


Figure H.1: C-C bond potential calculated in LAMMPS, GROMACS, and theoretically.

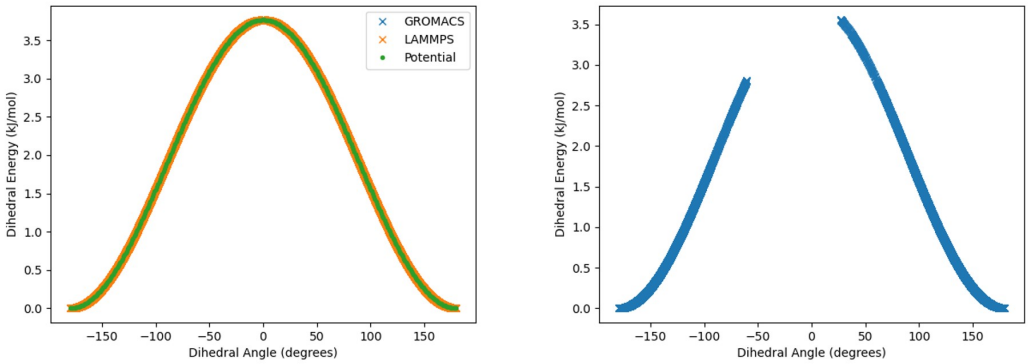


Figure H.2: C-C-C-C dihedral potential calculated in LAMMPS, GROMACS, and theoretically.

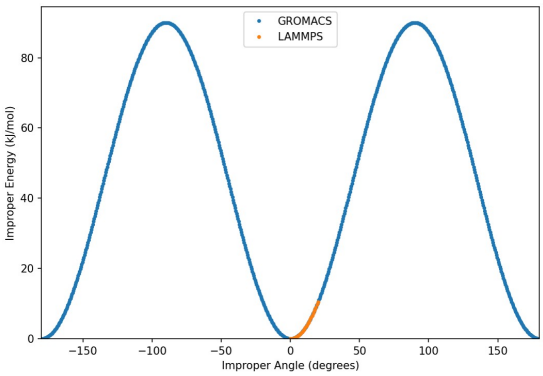


Figure H.3: C-C-O-O improper potential calculated in LAMMPS and GROMACS.

Appendix I

GLYCAM06j Alginate Parameters For Use in GROMACS

The non-bonded and bonded parameters for alginate systems adapted for use in GROMACS are shown in Listings I.1 and I.2.

Listing I.1: Non-bonded parameters used in simulated alginate systems in GROMACS.

```
[ atomtypes ]
; name      at.num    mass    charge ptype  sigma      epsilon
Cg            6       12.01    0.0000   A    0.339967   0.4577296
C             6       12.01    0.0000   A    0.339967   0.3598240
Oh            8       16.00    0.0000   A    0.306647   0.8803136
Os            8       16.00    0.0000   A    0.300001   0.7112800
O2            8       16.00    0.0000   A    0.295992   0.8786400
H1            1        1.008   0.0000   A    0.247135   0.0656888
H2            1        1.008   0.0000   A    0.229317   0.0656888
Ho            1        1.008   0.0000   A    0.035636   0.1255200
OW            8       16.00    0.0000   A    0.315070   0.4267680
HW            1        1.008   0.0000   A    0.100000   0.0000000
Ca           20       40.08    0.0000   A    0.292571   0.4261812
Cl           17       35.45    0.0000   A    0.383086   2.1820915
Na           11       22.99    0.0000   A    0.262459   0.1304459
```

Listing I.2: Bonded parameters used in simulated alginate systems in GROMACS.

```
[ bondtypes ]
; i   j          func   b0        kb
  Cg  Cg         1     0.1520     259408.0
```

```
  Cg  Oh          1     0.1430     267776.0
  Cg  Os          1     0.1460     238488.0
  Cg  H2          1     0.1090     284512.0
  Cg  H1          1     0.1090     284512.0
  Oh  Ho          1     0.0960     462750.4
  Cg  C           1     0.1530     184096.0
  C   O2          1     0.1250     548940.8

[ angletypes ]
; i    j    k      func  th0      kth
  Cg   Cg   Cg     1     113.50   376.560
  Cg   Cg   Oh     1     107.50   585.760
  Cg   Cg   H1     1     111.00   376.560
  Cg   Oh   Ho     1     109.50   460.240
  Cg   Os   Cg     1     111.60   418.400
  Cg   Cg   Os     1     108.50   585.760
  Cg   Cg   H2     1     111.00   376.560
  Oh   Cg   H1     1     110.00   502.080
  Cg   Cg   C      1     111.10   527.184
  Cg   C    O2     1     115.00   585.760
  Os   Cg   H2     1     110.00   502.080
  C    Cg   Os     1     112.36   527.184
  C    Cg   H1     1     109.50   418.400
  O2   C    O2     1     126.00   669.440
  Oh   Cg   Os     1     112.00   836.800
  Oh   Cg   H2     1     110.00   502.080
  Os   Cg   H1     1     110.00   502.080
  Os   Cg   Os     1     112.00   836.800 ; added for trimer (not in monomer)

[ dihedraltypes ]
; i    j    k    l    func phs kphi        mult
  Cg   Cg   Cg   Cg   1       0    1.882800 1
  Cg   Cg   Cg   Oh   1       0    0.418400 3
  Cg   Cg   Cg   H1   1       0    0.627600 3
  Cg   Cg   Oh   Ho   1       0    0.753120 3
  Cg   Os   Cg   C    9       0    1.338880 3
  Cg   Os   Cg   C    9       0    1.882800 2
```

```
41  Cg   Os   Cg   C    9       0   -2.510400 1
42  Cg   Os   Cg   H1   1       0    1.129680 3
43  Cg   Cg   Os   Cg   1       0    0.669440 3
44  Oh   Cg   Cg   Oh   9       0   -0.418400 1
45  Oh   Cg   Cg   Oh   9       0    3.974800 2
46  Oh   Cg   Cg   Oh   9       0    2.301200 3
47  Oh   Cg   Cg   Os   9       0   -4.602400 1
48  Oh   Cg   Cg   Os   9       0    1.046000 2
49  Oh   Cg   Cg   H1   1       0    0.209200 3
50  Oh   Cg   Cg   H2   1       0    0.209200 3
51  Cg   Cg   Cg   C    1       0    1.882800 1
52  Cg   Cg   Cg   Os   1       0   -1.129680 1
53  Cg   Cg   Cg   H2   1       0    0.627600 3
54  Cg   Cg   C    O2   9       0    0.041840 3
55  Cg   Cg   C    O2   9       0   -1.715440 2
56  Cg   Cg   C    O2   9       0    0.020920 1
57  Cg   Os   Cg   Oh   9       0    4.016640 3
58  Cg   Os   Cg   Oh   9       0    5.773920 2
59  Cg   Os   Cg   Oh   9       0    4.518720 1
60  Cg   Os   Cg   H2   9       0    2.510400 2
61  Cg   Os   Cg   H2   9       0    0.418400 3
62  Cg   Os   Cg   Os   9       0    4.016640 3 ; added for trimer (not in
    monomer)
63  Cg   Os   Cg   Os   9       0    5.773920 2
64  Cg   Os   Cg   Os   9       0    4.518720 1
65  Os   Cg   Oh   Ho   1       0    0.753120 3
66  Os   Cg   Cg   H1   1       0    0.209200 3
67  Os   Cg   Cg   Os   1       0    1.673600 2 ; added for trimer (not in
    monomer)
68  C    Cg   Cg   Oh   9       0    0.418400 3
69  C    Cg   Cg   Oh   9       0    0.836800 2
70  C    Cg   Cg   Oh   9       0  -10.460000 1
71  C    Cg   Cg   H1   1       0    0.418400 3
72  C    Cg   Cg   Os   9       0   -0.418400 3 ; added for trimer (not in
    monomer)
73  C    Cg   Cg   Os   9       0    0.418400 2
74  C    Cg   Cg   Os   9       0   -4.184000 1
```

```
75   O2   C    Cg   Os   9       0    0.083680 3
76   O2   C    Cg   Os   9       0   -3.033400 2
77   O2   C    Cg   Os   9       0    0.083680 1
78   O2   C    Cg   H1   1       0    0.000000 1
79   H1   Cg   Cg   H2   1       0    0.711280 3
80   H1   Cg   Cg   H1   1       0    0.711280 3
81   Ho   Oh   Cg   H1   1       0    0.753120 3
82   Ho   Oh   Cg   H2   1       0    0.753120 3
83   C    O2   O2   Cg   4     180   43.932000 2 ; improper
```

Appendix J

Umbrella sampling for investigating the interactions between alginate chains

A trial run of umbrella sampling was performed using two 10-monomer long G chains with concentrations of 0.09 M Na^+ and Ca^{2+} and 0.016 g/g alginate in water. This system was set up and minimised as described in Chapter 7 before being run for 20 ns of NPT simulation. One chain was then pulled 3.5 nm away from the other using steered molecular dynamics (SMD). System configurations were extracted based on the increasing distance between the chains, in 0.2 nm intervals. These configurations were run for 10 ns NPT. A weighted histogram analysis method (WHAM) was then performed on these configurations using GROMACS. Examples of the SMD system, and the WHAM output is given in Figure J.1 and J.2, respectively.

Following this test, this method would need optimised, in which configurations, SMD parameters, and WHAM parameters were tweaked for use with this specific systems. This methodology could then be repeated for different chains in G-G, M-M, and G-M pairings, and for different water and ion environments. It is expected that, from this methodology, and analysis of ion and water positioning, detail of the chain-chain interactions would be extracted and the reason for G-M interactions consistently presenting as more favourable than G-G or M-M interactions in the simulated systems of this work could be gleaned.

Appendix J. Umbrella sampling for investigating the interactions between alginate chains

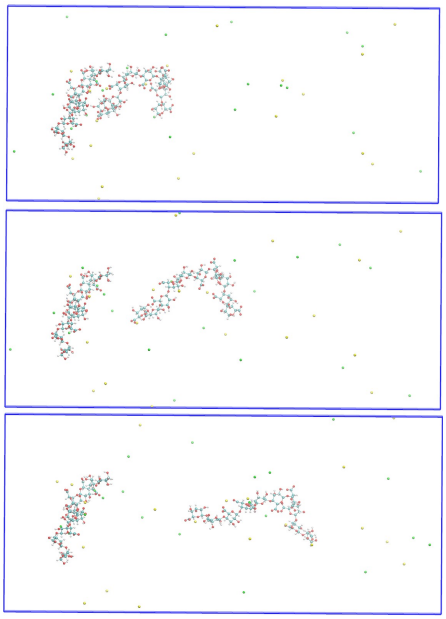


Figure J.1: Images for steered molecular dynamics simulation of two 10-monomer G chains where the right chain is being away from the left chain at a constant speed.

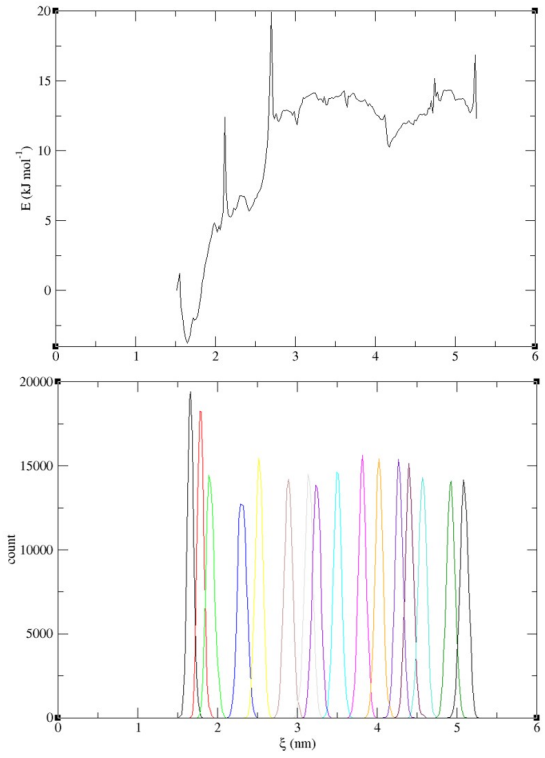


Figure J.2: Energy and sampling output from WHAM analysis of the system shown in Figure J.1.