Optimising the design of building blocks for self-assembly of discrete clusters

MADGE, JIM

OPTIMISING THE DESIGN OF BUILDING BLOCKS FOR SELF-ASSEMBLY OF DISCRETE CLUSTERS

JIM MADGE

THESIS PRESENTED FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Department of Chemistry
Durham University

2017

OPTIMISING THE DESIGN OF BUILDING BLOCKS FOR SELF-ASSEMBLY OF DISCRETE CLUSTERS

Jim Madge

Self-assembly is the spontaneous organisation of matter into an ordered state. Significant progress has been made in the fabrication of synthetic components for self-assembly, opening up routes to building blocks for the production of functional materials and nanomachines. The information required to assemble a target structure can be encoded into the building blocks. For assembly of an equilibrium state, the target must be thermodynamically stable and the pathway must avoid kinetic traps. The design of building blocks must address both these requirements.

In this work a generic model is introduced which, through an explicit representation of interactions, is able to express many approaches to self-assembly. The model consists of hard cubic particles, whose faces are patterned with attractive patches. A hybrid, dynamical Monte Carlo protocol is developed to simulate self-assembly of such inhomogeneous systems efficiently, accounting for both internal rearrangements and relative diffusion rates of aggregates.

Using this single model, different self-assembly strategies are assessed, ranging from simple approaches with only one type of building block, to more complex strategies using multiple components and hierarchical paths. The important case of fully addressable targets, where all components of the structure are unique and have a specific location, is then examined in more detail. Firstly, a new metric is introduced to quantify the problem of competition between partly assembled fragments, which is a prominent source of kinetic traps in addressable clusters. Principles are established for minimising this problem. Secondly, a scheme for globally optimising the interactions amongst a set of particles is developed to maximise the performance of building blocks of a given complexity. This also makes it possible to determine the level of complexity required for a given target to assemble reliably.

The computational tools and general principles established in this work should be applicable in a wide range of self-assembly problems.

CONTENTS

ACRONYMS

AFM atomic force microscopy 10

BD Brownian dynamics 23

LJ Lennard-Jones 14

LP linear programming 93

MC Monte Carlo 20

MD molecular dynamics 14

SATOBB seperating axis theorem for oriented bounding boxes 57

SEM scanning electron microscopy 10

ss-DNA single stranded DNA 88

SVMMC symmetrised virtual move Monte Carlo 34

TEM transmission electron microscopy 13

VMMC virtual move Monte Carlo 30

WG Whitelam and Geissler 30

No part of this thesis has been submitted for a degree at Durham or any other institution. The research presented here is is the authors work unless indicated otherwise.

ACKNOWLEDGEMENTS

First, I must thank Mark Miller for supervising my Ph.D. He has been a constant source of guidance, assistance and knowledge. Particular thanks are required for his tireless correcting of my grammar and (hopefully) changing my writing for the better. He has instructed me on the appropriate placement of commas, the correct use of dashes—why Lennard-Jones is one person whereas Lennard–Jones is two—and the difference between that and which. Any mistakes that remain are my own.

Mark Wilson also deserves significant thanks for being a friend and mentor since I was an undergraduate. My first real experience of simulation and research was in his group, using GROMACS to simulate the stacking of disodium cromoglycate. Since then he has always provided me with opportunities to further my knowledge, not least by applying for funding on more than one occasion.

I would also like to thank all people, past and present, in offices CG200 and 200X, who I have had the pleasure of knowing through my time at Durham. I doubt there is anywhere in the university quite like it, and I will take with me fond memories of tea, board games and pub visits. Of these people, Martin (the ever reliable purveyor of strange foods) deserves special mention. He has definitely made a positive impact on the group, both in work and all "after-school" activities.

Thanks to all members of Breakfast Club, who enjoyed Magnum, bingo and discussion of bowls. Also anyone who indulged me by letting me impose my taste in films on them in occasional after-hours screenings.

Finally, it would be naive of me not to acknowledge the huge amount of free and open-source software that I have used. Very little of my work would have been possible without Linux, the GNU utilities and compilers, git, python, vim, povray and TEX Live.

CHAPTER 1

INTRODUCTION

1.1 SELF-ASSEMBLY

The term self-assembly describes a very broad range of processes.[1] The common features of these phenomena are that an ordered structure forms spontaneously, without detailed external intervention, and to a high fidelity, beginning from components in an initial state of disorder.

This definition encompasses contrasting types of self-assembly. One broad class of self-assembly process is driven by the thermodynamic stability of the target. In this case, spontaneity implies that the free energy change associated with the transition from the disordered to the ordered state must be negative. This free energy is a combination of the energetic interactions[2] between the building blocks, and the entropy of the entire system.[3–5] Alternatively, self-assembly can be directed[6] through the use of external fields,[7,8] flow,[9] or the presence of a template or interface.[10,11] The external influence modifies the free energy landscape, establishing a new equilibrium, or may allow the system to overcome a transition barrier to the ordered state. In the latter case the assembled structure remains stable when the field is removed.[12] Application of these external factors therefore allows us, in some instances, to control or enhance self-assembly.

There are also systems which self-assemble out of thermodynamic equilibrium and which require the continual input of energy to remain ordered.[13] The best known and understood examples of nonequilibrium self-assembly are biological.[14] For example, the many processes required to produce and regulate cells require the continual input of chemical energy, and result in structures that would otherwise tend to decay. Nonbiological examples of self-assembly out of equilibrium are less common. The energy driving organisation may come from a constant supply of chemical fuel,[15,16] or may come in the

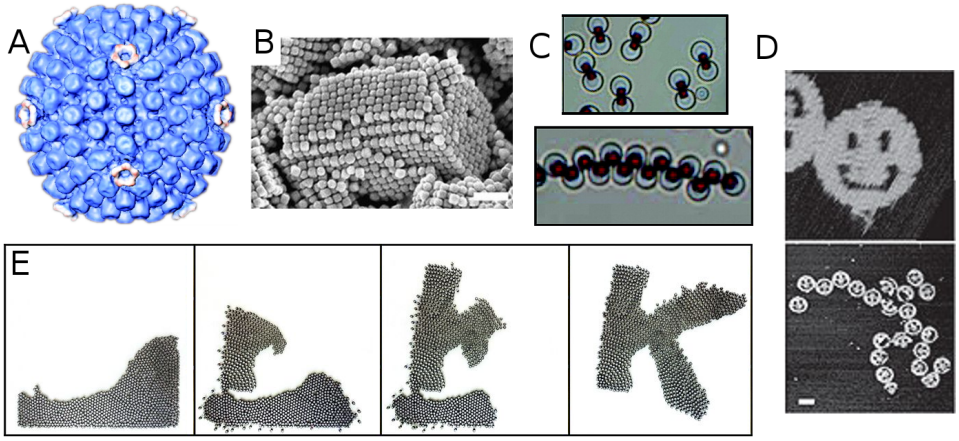

Figure 1.1: Contrasting examples of self-assembly. (A) rendering of the icosahedral capsid of the herpes simplex virus [reproduced from Grünewald *et al.*[19]]. (B) scanning electron microscopy (SEM) micrograph of a binary lattice of alternating cubic and spherical colloidal particles, assembled with the aid of DNA-mediated interactions [reproduced from Lu *et al.*[20]]. (C) an example of reconfigurable assembly: The magnetic patchy particles form dimer clusters (top picture), but may be made to assemble into chains (bottom picture) upon the application of a magnetic field [adapted from Sacanna *et al.*[21]]. (D) atomic force microscopy (AFM) micrographs of two-dimensional clusters in the shape of a smiley face. The clusters are formed from a single strand of DNA, shaped using DNA "staples", a technique called DNA origami [adapted from Rothemund[22]]. (E) an example of active self-assembly: centimetre-scale robots in a swarm cooperatively assembling into the letter K [reproduced from Rubenstein *et al.*[23]].

form of continual perturbation such as a rotating field.[17] Nonequilibrium self-assembly is therefore inherently dynamic as the system is pushed away from the equilibrium state of disorder. The dynamics may come in the form of active particles such as active colloids or bacteria,[18] but may also be due to a constantly evolving environment. A selection of contrasting examples of self-assembly are presented in figure 1.1.

In this thesis, we will concentrate on the first category described above, *i.e.* cases where the target structure is thermodynamically stable. As we shall see, there can be many challenges both to ensuring that the target is indeed stable and to making it reliably kinetically accessible.

The range of products that can be obtained by self-assembly is wide. One important contrast is between periodic, unbounded structures, such as chains, crystals and lattices on the one hand, and finite, discrete structures on the other. The latter may range in the number of building blocks from dimers to structures consisting of hundreds or thousands of building blocks, and in physical dimensions from a cluster of small molecules to a biological cell.

Nature presents a rich collection of examples of self-assembly, in many cases forming the basis of life. Often in nature evolution has encouraged a remarkable efficiency and economy of resources. For example, many capsid shells of viruses form spontaneously to a high monodispersity from a specific number of protein subunits, and often only a single type of protein is used.[24–27] Nature can take efficiency one step further; for some icosahedral capsids identical proteins occupy more than one distinct local environment. This phenomenon is termed quasi-equivalence,[28–30] and demonstrates that assembly from single components is not only possible, but that a well designed building block may even play multiple roles in the assembled structure.

The structure of a capsid is encoded in the structure of the proteins that form it, similarly to how the structure of a protein is determined by the sequence of amino acids that constitute its primary structure. We could therefore ask what it is that makes one protein form an icosahedral capsid instead of a cylindrical capsid, and how this knowledge can be exploited to direct assembly. With an understanding of how structural features control assembly and a method to produce building blocks to a design specification, the path would be open for the production of particles to spontaneously form desirable materials. The following section will outline the extent of progress in creating self-assembling building blocks from colloidal particles.

1.2 Colloidal Self-Assembly

When producing building blocks, colloidal particles are an attractive choice because of techniques to control both their shape and interactions. These give control over both energetic and entropic driving forces for self-assembly. A colloidal system is a suspension of insoluble particles of one medium dispersed in a second medium. The dispersed particles typically range in size from 100 nm to 1 µm. The term most commonly refers to suspensions in a liquid, and in this work concerns solid particles dispersed in a liquid. This section provides an overview of aspects of colloidal particle design and their impact on self-assembly.

1.2.1 Particle Shape

The shape of colloidal particles plays an important role in their crystallisation, by influencing the entropic forces which drive the process.[3] At high densities colloidal particles minimise their excluded volume in order to increase their translational and rotational en-

tropy. This occurs when particles are aligned or arranged in a regular array.[4] The shape of the particles determines the crystal structure that is adopted, and non-spherically symmetric particles may also show orientational ordering in dense phases.[5]

Theory and simulation of completely hard spheres, dating back to the 1950s, showed crystallisation into hexagonally close-packed lattices at high densities.[31–33] The experimental study of nearly-hard colloidal particles confirmed that these simulations, despite their simple potentials, not only reproduce the correct phases, but are also able to make quantitative predictions.[34] This confirms that hard bodies can be a good model for colloidal particles. The simulation of hard particles extends far beyond spheres, and there exists large body of work demonstrating the relation between shape and crystal structures.[4,35,36] Perhaps the best example is the work of Damasceno *et al.*[37] in which the phases of a set of 145 convex polyhedra were characterised. The particles were sorted into groups, based on their geometry and coordination in the fluid phase, which predicted the nature of their organisation (whether it is crystalline, liquid crystalline, disordered *etc.*). In this case simulation has advanced ahead of experiment, due to the ease of controlling the shape of particles *in silico*, demonstrating how simulation can be a powerful predictive tool in self-assembly.

The theoretical studies of hard bodies of different shapes are becoming ever more relevant as further control over the shapes of colloidal particles is achieved.[38] For example, prolate colloidal particles, commonly called nanorods,[39] may be synthesised. These particles have a highly aligned crystal state and display some liquid crystal phases. Protrusions may be created on the surfaces of particles to produce dumbbell shaped particles,[40,41] and "Mickey Mouse" particles,[42,43] so called because of their two ear-like lobes. A selection of some of the shapes currently possible are presented in figure 1.2.

Another entropic contribution is the depletion force,[46,47] where smaller solute molecules, which prefer not to be confined between colloidal particles, push the colloids together. The interplay between the shape of the colloid and size of depletants affects phase behaviour and the crystal structures adopted.[4,48] The Mickey Mouse and dumbbell shaped particles displayed in figure 1.2 both feature bumpy lobes. The depletion interaction affects the rough and smooth lobes unevenly, and with the addition of a depletant, only the smooth lobes become attractive. This results in micelle like structures for dumbbells[49] and discrete clusters or extended tube-like structures for the Mickey Mouse particles.[42,43] Cavities may also be created on the surfaces of nanoparticles leading to lock-and-key type assembly,[50,51] where entropy drives assembly between only specifically matched partners.

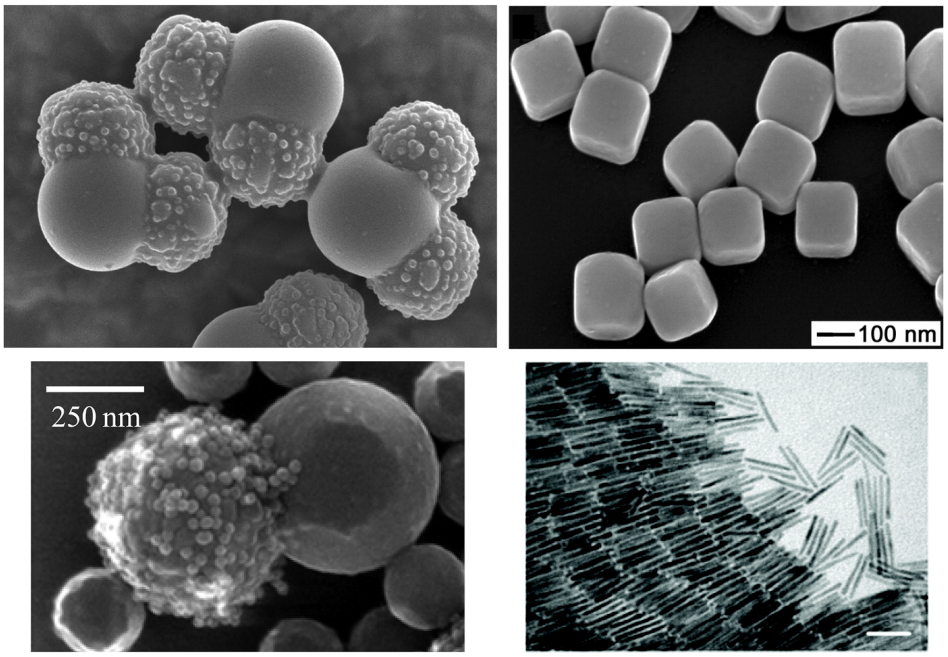

Figure 1.2: Images of a range of shapes of colloidal particles available to us. The top left panel shows an SEM micrograph of Mickey Mouse particles with bumpy lobes [reproduced from Wolters *et al.*[42]]. Top right is a SEM image of silver nanocubes [adapted from Sun and Xia[44]]. A dumbbell particle with one smooth and one bumpy lobe is shown in SEM image bottom left [adapted from Nagao *et al.*[41]]. Finally the bottom right panel shows a transmission electron microscopy (TEM) image of a system of nanorods [adapted from Carbone *et al.*[45]].

A prominent class of non-spherical nanoparticle are colloidal cubes, which may be formed with a great deal of control over their size.[44,52,53] Colloidal cubes display a transition from a fluid phase to a cubic crystalline phase with increasing density.[54,55] This structure is understandable when considering excluded volume, but simulations have revealed a more surprising contribution to the stability of a cubic crystal. Smallenburg *et al.* found an unusually large number of vacancies in the cubic crystals,[56,57] which allow neighbouring particles to move into adjacent lattice sites. In some cases it is possible for entire rows to become dislocated. The translational freedom that vacancies create therefore entropically stabilise the structure.

1.2.2 Patchy Particles

Besides shape, interactions between colloidal particles may be modified by functionalising them with attractive regions; giving them "patchy" interactions.[58–60] The regions of differing character on the surface of the particles are similar to the hydrophobic and

hydrophilic regions on the surface of proteins, which play a role in their assembly and binding to substrates. The anisotropic interactions, provided by patches pointing in particular directions, can be exploited to encourage the colloids to self-assemble into discrete clusters, rather than the periodic or amorphous structure that would be formed without them.

Patchy interactions may also be used to assemble crystal structures not possible with entropic interactions alone, such as the diamond lattice.[61] Another example is the kagome lattice, which has been successfully assembled (with the aid of sedimentation) from a single colloidal component.[62] The particles feature two hydrophobic patches on opposite poles of the spherical core. Careful consideration of the patch size ensures that the kagome lattice is the most stable configuration. An example section of the kagome lattice, showing the bonding between hydrophobic patches, is shown in figure 1.3 top right.

Simulation is useful as a predictive tool in the self-assembly of patchy particles, and numerous models exist to represent them. The Kern–Frenkel model is an early attempt to represent patchy particles. In the model particles feature a hard spherical core and patches that interact through a square-well potential when aligned.[65] This model is popular and has been widely used[61,66–69] because, despite its simplicity, it captures the essential features of patchy interactions. More detailed models also exist, like the one introduced by Doye *et al.*, which features a Lennard-Jones (LJ) attraction between patches and a smooth angular attenuation when the orientation of patches deviates from optimal alignment. This model has proved versatile for exploring the formation of icosahedral clusters,[70,71] the accessibility of various polyhedra,[72] and (with some small modifications) to provide insight on protein clusters.[73]

Alternatively, patchy particles can also been modelled as rigidly connected spheres. Here attractive spheres represent patches, while those acting as the body of a nanoparticle are repulsive. The advantage of this approach is that molecular dynamics (MD) packages, which capture the dynamics of assembly, may be used for simulation. Zhang and Glotzer constructed a large range of spherical and plate-like patchy particles using this approach.[74] Their simulations demonstrated how particle design may be used to assemble different targets. Viral capsids have also been modelled as a union of spheres, providing a great deal of insight into assembly pathways. In particular it was shown that the reversibility of binding plays a key role in achieving high yields, as it allows the system to escape kinetic traps.[25,75]

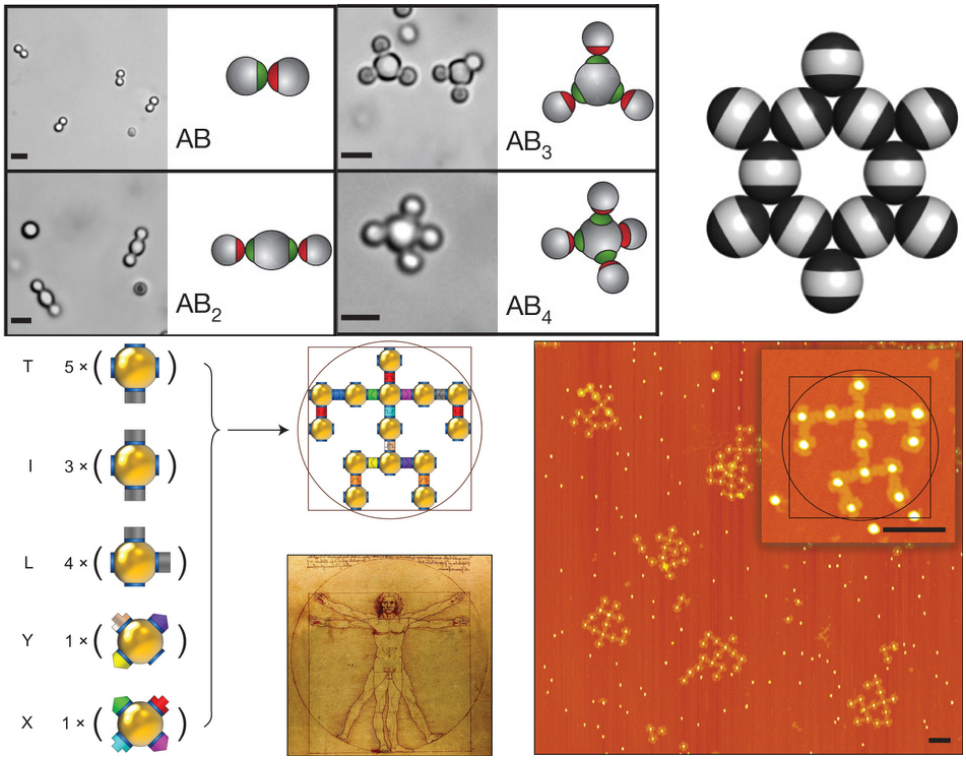

Figure 1.3: A selection of self-assembled systems and their colloidal building blocks. Bright-field images of "colloidal molecules" along with schematics showing the structure of the particles and locations of their DNA patches are shown top left [adapted from Wang *et al.*[63]]. The top right panel shows a section of the kagome lattice formed by the doubly patched nano particles [adapted from Chen *et al.*[62]]. The set of building blocks designed to form the Vitruvian man clusters, along with a picture of the inspiration and an AFM image showing assembled clusters is displayed bottom [reproduced from Liu *et al.*[64]].

1.2.3 DNA-Grafted Colloids

Precise control over the binding of particles can be achieved with DNA-mediated interactions.[76] Initially, interest was driven by the desire to form reversibly crystallising colloids controlled by temperature.[77–79] This has recently been extended to the controllable formation of binary lattices of otherwise incompatible, differently-shaped components (see figure 1.1 B).[20,80,81] The power of DNA interactions goes beyond reversible binding; it is also possible to create selective interactions using distinct DNA sequences. This was exploited by Wang *et al.* to create "colloidal molecules" with bonding arrangements similar to sp, sp^2 and sp^3 hybridised carbon atoms. Small clusters of spherical colloids were swelled with polymer, leaving exposed bumps, which were subsequently functionalised with DNA strands. A schematic of this process along with some example colloids are

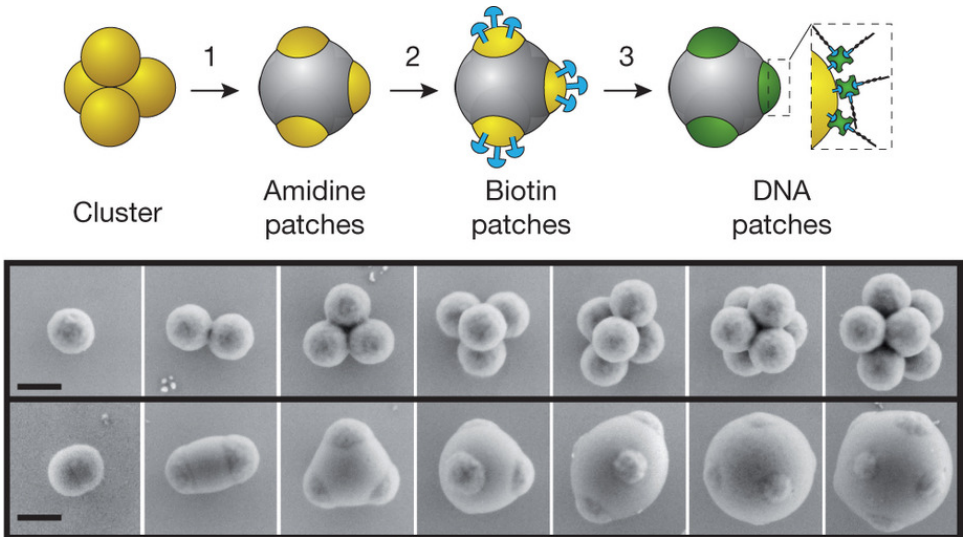

Figure 1.4: A schematic of the production of "colloidal molecules" by swelling small clusters with polymers, followed by fabrication with DNA patches, and a series of examples the valencies and geometries achieved [reproduced from Wang *et al.*[63]].

shown in figure 1.4. A pair of complementary DNA sequences ensure that the core of each colloidal molecule only binds, as intended, to monovalent "colloidal atoms". Examples are shown in figure 1.3 top left, where the red and green regions represent the two complementary DNA patches.

Although it is not yet possible to pattern nanoparticles arbitrarily, in a short space of time progress in designing colloidal particles has been great. It is reasonable to suppose that further developments will produce a still richer variety of shapes and interactions. With an increasingly diverse and complex set of building blocks, self-assembly will be able to reach more targets with greater accuracy.[82]

1.3 Addressable Assembly

Nanoscale machines and devices consist of many unique parts that must be precisely placed in order to ensure correct function.[5] Achieving such a structure would be difficult from a small number of components, as this approach favours producing highly symmetric structures. The solution to this problem, and the assembly of other targets with low symmetry and modularity, may be to use many unique components whose role in the structure are predetermined. This approach is called addressable self-assembly.

In a fully addressable cluster, each component is unique and occupies a particular position in the structure.[5,76,83] In order to ensure that each particle plays its intended role,

Figure 1.5: The top image shows a series of AFM images of clusters formed from DNA tiles [reproduced from Wei *et al.*[85]]. The bottom image shows DNA brick structures with the Arabic digits carved into one face, the top row are representative diagrams where are the bottom contains TEM images of clusters [adapted from Ke *et al.*[86]].

specific interactions are used so that each component will only bind strongly with its intended neighbours. DNA was shown in the previous section to be useful for controlling the specific binding of colloidal particles. The ability to create large numbers of specific interactions makes DNA the tool of choice for addressable assembly.

Self-assembly of a target may be programmed with information about its structure, connectivity, and the pathway of its formation.[84] The encoding of the necessary information into addressable building blocks is an example of programmable assembly. Another novel example is the assembly of a swarm of robots, where each individual robot has knowledge of the target.[23] The swarm rearranges in a cooperative manner to form the desired structure (figure 1.1). The information may also be stored separately from the building blocks, and be enacted by an assembler. For example, in the case of proteins, information is stored in DNA and transcribed into mRNA before being acted upon by the ribosome.[84] Programmable assembly allows for the assembly of arbitrarily shaped structures, unlike the relatively simple clusters and lattices most commonly seen in colloidal self-assembly. Addressability may be one route to programmable colloidal assembly.

The most prominent examples of addressable assembly are DNA bricks[86,87] and DNA tiles.[85] In both cases, folded DNA building blocks self-assemble with great accuracy into addressable clusters of almost arbitrary structure, some of which are shown in figure 1.5.

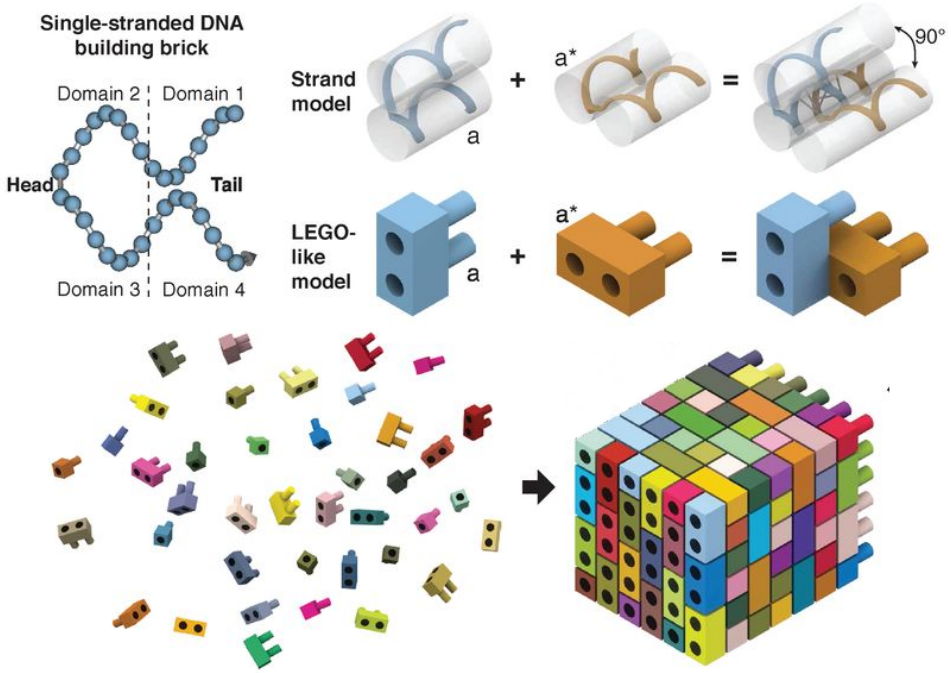

Figure 1.6: A schematic showing the structure of a DNA brick, formed from a single strand of DNA with its four interactive domains. The strands are abstracted to a LEGO-like model, showing how individual bricks are connected with a 90° angle. Below the LEGO-like formalism is used to show how a set of bricks may assemble into a three-dimensional structure [all images adapted from Ke *et al.*[86]].

Each DNA brick is formed from a single strand of DNA whose sequence is designed so that it will fold into a staple shape, leaving four interaction sites exposed. These sites are encoded so that they will be complementary to, and hence bind selectively to, the corresponding site on the intended neighbour. A schematic of the structure of each brick and their assembly into a cluster is shown in figure 1.6

The myriad structures assembled with DNA bricks are all made from subsets of the same complete set of bricks. First, all the bricks required to create a three-dimensional "canvas" are designed. Structures that fit within this canvas may then be assembled by omitting from the mixture the bricks that are not required, in effect carving a structure from the cubic canvas. This approach is, therefore, less design intensive than it may first seem, as a single set of building blocks may be used to assemble many targets.

The success of the DNA brick approach was initially a surprising result, as it was thought that assembly from so many unique components would be too susceptible to error to work in practice. Theory and simulation, conducted in the Frenkel group, has made a valuable contribution to the study of DNA brick assembly, revealing how a slow

nucleation step favours correct assembly.[88,89] The group's methods include both on-[88] and off-lattice[90,91] simulations, as well as a theoretical model for predicting the free-energy landscapes of addressable clusters.[92] Beyond explaining how addressable assembly is possible, this body of work has also made a number of significant predictions about how to improve addressable assembly in terms of coordination number, bond strengths and temperature protocol.[83] These results will be drawn upon significantly throughout this thesis.

As DNA may be grafted to colloidal particles, they are candidate building blocks for addressable assembly. Early theoretical investigations of the assembly of DNA-coated colloids identified the minimum-energy configurations for clusters of spheres with short-ranged interactions,[93,94] which are distinct from the well established minima of LJ clusters.[95] Subsequent work aimed to optimise the energy of the target structure through the use of a matrix of interactions.[96] The optimal interaction sets led to studying the assembly of large addressable clusters, which placed the size limit at around one thousand components.[97] For greater numbers of particles, the large number of possible defect states reduces the yield of the desired target. This work has important implications for the size of self-assembled structures we may achieve with addressable components.

In contrast to building blocks formed entirely from DNA, which spontaneously fold into shape after synthesis, addressable colloids need to be fabricated over a greater number of steps. This could make the production of large numbers of addressable colloidal components slow and cumbersome when compared to DNA bricks. A promising approach was demonstrated by Liu *et al.*[64] who produced programmable colloids by attaching a DNA frame, which carries binding strands, to the particles. The group successfully assembled small clusters, chains, periodic arrays and a cluster resembling the Vitruvian man. The building blocks for the latter cluster are shown schematically in figure 1.3 bottom, along with an AFM image of assembled targets. This is a viable approach for the production of large numbers of unique building blocks as many different DNA frames may be synthesised, similarly to the production of DNA bricks, before attaching them to identical colloidal particles. Addressable building blocks may be created in this scheme, but in this instance the authors chose to take advantage of the symmetry of their targets and use a smaller set of particles.

1.4 Structure and Aims of this Thesis

This introduction has surveyed the broad class of self-assembling systems, from simple colloids to fully addressable DNA bricks. It has also presented the state-of-the-art in self-assembly in practice and theory. The general aims of this thesis are as follows.

1. To develop a generic model for patchy particles, which has the versatility to study contrasting approaches to and types of self-assembly, including addressable assembly.

2. To assess the state of simulation methodology in self-assembly, and develop new approaches if necessary.

3. To compare different assembly strategies within a single framework, identifying the advantages and drawbacks of each approach and learn in what circumstances each may be appropriate.

4. To investigate the effect of structure and connectivity on the assembly pathway of discrete addressable targets, and use this information to optimise yield.

5. To address contrasting evidence on the optimal distribution of bond strengths in addressable structures. Some work has suggested that homogeneous interactions are optimal,[96,98] yet the assembly of DNA bricks in practice and theory contradicts this.[83,86,89]

6. To design optimal interactions for self-assembly, within our patchy model, through an automated procedure.

7. To establish the limits of target complexity that may be assembled with a restricted set of information encoded into building blocks.

The next chapter will introduce the relevant simulation techniques in the computational study of self-assembly. Monte Carlo (MC) will be discussed in particular depth, and two existing MC algorithms, featuring cluster moves, will be described in detail. Cluster moves are important for the efficient simulation of strongly bound systems, as often found in self-assembly.

The patchy model used throughout this work will be defined in chapter 3. The model allows for surfaces of particles to be patterned with attractive patches. This approach gives

an explicit representation of the interaction between particles and also allows the interaction design space to be explored to find optimal designs. Following this, the characteristics of the two MC algorithms introduced in chapter 2 will be compared and the justification for a new hybrid algorithm presented. This pseudo-dynamical algorithm (also introduced in chapter 3) moves aggregates in unison, and allows them to relax internally or combine with other aggregates. These features allow for efficient assembly when compared to other MC schemes. The algorithm also ensures that the diffusion of aggregates occurs in a realistic manner based on the size of the aggregates. This is a feature neglected by other MC schemes featuring collective motion.

Chapter 4 will contrast different strategies for self-assembly, addressing aim 3 above. The versatility of the patchy model will be exploited to create particle designs from intuition. Control over the number, arrangement and interactions of patches will be used to influence the pathway of assembly. Through developing strategies for assembling the same simple target, the merits of each may be compared. The explored strategies range from a minimal one-component system inspired by the natural example of certain viral capsids, to a fully addressable strategy similar to DNA bricks. These simulations will feature multiple copies of each building block, and so can probe sources of error not evident in other simulations with only a single copy.

Addressable assembly was introduced in section 1.3, and this will be the focus of chapters 5 and 6. In chapter 5 the effect of connectivity and distribution of bond strengths on assembly will be studied. One of the key aims of this investigation is to try and find ways to avoid addressable systems becoming stuck in frustrated configurations where there is no possibility of completing a full set of targets. The emergence of these frustrated states is an important result from chapter 4. This frustration is found to be a major limiting factor on the success of addressable assembly. A method is introduced to quantify the amount of frustration in any configuration of an addressable system, and is used to explore how this problem may be avoided. This chapter will address aims 4 and 5.

Specific interactions are necessary for addressable assembly to ensure that building blocks only bind to their intended neighbours. Chapter 6 will explore designing particles with optimal addressable interactions. The faces of the patchy particles are treated like canvases, on which a set of patches can be arranged to create an interface. By ensuring that a set of faces are as distinct as possible, interfaces with addressable interactions are created. In particular, the limits of addressability using a restricted set of patches and unique interaction types will be explored, thereby addressing aims 6 and 7.

CHAPTER 2

COMPUTATIONAL METHODS IN SELF-ASSEMBLY

2.1 SIMULATING SELF-ASSEMBLY

Simulation techniques are very important to the study of self-assembly. Numerous examples of simulation studies were shown in chapter 1. Simulation allows us to probe aspects of self-assembly which may not be studied physically, perhaps due to limitations in the building blocks which can be made. Furthermore, the trajectories generated by a simulation technique give us access to many static and dynamic properties of the system simultaneously. Simulation is particularly influential when it is able to lead experiment. For example, configurable models for building blocks allow us to easily tweak designs and identify features beneficial or detrimental to assembly.[70,83,99] Even very simple and generic models can be illuminating, identifying general features of assembly which are applicable to many self-assembling systems.[25,98,100]

Self-assembly is an inherently dynamic process, and so to fully understand it dynamics must be modelled in a realistic manner. The most direct way to do this is by generating a dynamic trajectory, but dynamic information may also be obtained from static methods. For example, the energy landscape of a system may be characterised, in particular the locations of energetic minima, maxima and transition states.[101] While not inherently dynamic, these methods may provide information about self-assembly. Transition state theory can be used to calculate rates of assembly, or a Markov chain model may be created. Metastable states on the energy landscape may indicate kinetic traps, which can significantly slow self-assembly. Hence, energy landscape methods can also be used to say something about the reliability of assembly. However, diffusion of building blocks and their ability to fit into a target cluster can be important limiting factors in assembly. These features can only be accounted for by an appropriate treatment of dynamics.

For particularly coarse models a simple approximation of dynamics may be suitable and some lattice MC simulations fall into this category. Here, building blocks may move with limited freedom in translational and rotational space. The fine detail of the motion of particles is lost, but the essential aspects of particles' diffusion, combination and separation remains. A protocol capturing just the essential features of dynamics was used by Hedges *et al.* to simulate a set of addressable tiles.[98] Here, tiles were added to or removed from randomly picked sites on a square lattice, with the exception of sites for which all neighbouring sites were occupied. This was in order to model the slow relaxation within a solid, as a site needs to be either on an edge, or next to a vacancy for a change in its state to occur.

MD is a popular dynamical simulation technique that is important to the study of self-assembly. In MD the coordinates and orientations of particles are evolved in time by iteratively solving Newton's equations of motion. With a sufficiently small time step this method produces dynamically realistic trajectories. However, there are barriers to using MD for simulating colloidal self-assembly. As colloidal particles are very large relative to solvent molecules it is infeasible to simulate colloidal particles with an appropriate amount of solvent at atomic detail. In the absence of a solvent beads, colloidal particles in a MD simulation would exist in a vacuum and display unrealistic ballistic motion. One solution to this problem is to use a coarse representation of the system where colloidal particles may be represented by one or more large beads and solvent by smaller beads.

An alternative approach is to account for the effect of solvent implicitly. One popular way of achieving this is Brownian dynamics (BD). A BD simulation may be viewed as a MD simulation with the addition of a random force to the iterative integration of the equations of motion. This random force models the collisions of solvent with colloidal particles. This saves considerable computational expense relative to modelling solvent explicitly as the majority of molecules in the system have been removed, and so the forces acting on them need not be calculated.

As MD and BD integration with a fixed time step requires taking derivatives of the potential they are not suited to simulating hard bodies. However, particles of this type are commonly used in the modelling of patchy particles and colloids. One way to circumvent this problem, shown in section 1.2, is to build colloidal particles as a union of rigidly connected spheres. However, this approach has its own shortcomings, as the rough surfaces that this technique creates can leave unwanted artefacts contaminating results.[102,103] Furthermore, modelling colloids in this way is computationally expensive as a single particle

consists of many spheres which must have the forces acting on them calculated, and their positions tracked.

On the other hand, MC schemes, by virtue of being able to reject moves, are easily able to handle hard bodies. Also, as derivatives of the potential are not needed, anisotropic potentials are simple to implement. Therefore, it is clear how the desire for a dynamically realistic MC algorithm would arise. In the limit of a small step size, a simple, single move MC algorithm can produce Brownian-like dynamics.[104,105] The dynamics occur on a timescale that depends on the acceptance rate of moves. The overall dynamics of the Brownian MC method agree with BD even in glass formation and nucleation, where we might expect somewhat inhomogeneous configurations to disrupt the kinetics. However, it may not be applied when there are large heterogeneities in density, such as crystallisation from a gas. The same method may also be applied to anisotropic particles, such as patchy nanoparticles, by tuning the relative size of translational and rotational displacements.[105]

While a Brownian-like scheme is appropriate for the dynamics of nanoscale particles, single-particle moves are not ideal for simulating colloidal self-assembly. This is because when only one particle at a time is moved aggregates will have to move through a series of cooperative steps, temporarily breaking links between particles before recombining them. As aggregates become larger and more densely linked these types of processes become more unlikely. The size of moves must then be made very small to offset the energetic penalty of temporarily breaking links. As a result, more moves need to be made to simulate the same effective time and simulations can become unreasonably expensive. Even when moves are small, collective motion of large aggregates is still unlikely, as it requires many moves of the same geometry to be proposed, and in an appropriate order. An alternative to single moves are cluster moves, where groups of particles are moved collectively. The remainder of this chapter will look at MC simulation in depth and how cluster moves are implemented. The technique of basin-hopping, used in a later chapter, will also be introduced.

2.2 Monte Carlo

2.2.1 Statistical Mechanics

The partition function of the canonical ensemble (where the system has a fixed number of particles N, volume V and temperature T) may be expressed as an integral over the

momenta $\mathbf{p}$ and positions $\mathbf{r}$ of particles as

$$Z_{NVT} = \frac{1}{h^{3N} N!} \int \int \exp\left[-\beta H(\mathbf{r}, \mathbf{p})\right] \, \mathrm{d}\mathbf{r} \, \mathrm{d}\mathbf{p} \,, \tag{2.1}$$

where h is the Planck constant, $H(\mathbf{r}, \mathbf{p})$ is the classical Hamiltonian and β is the reciprocal thermal energy $1/k_{\mathrm{B}}T$ (k_{B} is the Boltzmann constant). In many classical systems only the kinetic part of the Hamiltonian depends on momenta, and in these cases the integrals over positions and momenta may be separated. Integrating over the momenta results in a partition function which only depends on positions

$$Z_{NVT} = \frac{1}{N! \Lambda^{3N}} \int \exp\left[-\beta U(\mathbf{r})\right] \, \mathrm{d}\mathbf{r} \,, \tag{2.2}$$

where Λ is the thermal de Broglie wavelength, $\Lambda = \sqrt{h^2 / 2\pi m k_{\mathrm{B}} T}$ (where m is the mass of a particles), and $U(\mathbf{r})$ is the potential energy of the system.

The momentum integral is constant and independent of positions. This factor is identical for all configurations (sets of positions). It is therefore sufficient to consider only the configuration integral Z when exploring configuration space. Z is given by

$$Z = \int \exp\left[-\beta U(\mathbf{r})\right] \, \mathrm{d}\mathbf{r} \,. \tag{2.3}$$

The configuration integral may be used to calculate properties of the system similarly to the full partition function. For example, the ensemble average of a property A (which depends only on positions) is given by

$$\langle A \rangle = \frac{\int A(\mathbf{r}) \exp\left[-\beta U(\mathbf{r})\right] \, \mathrm{d}\mathbf{r}}{\int \exp\left[-\beta U(\mathbf{r})\right] \, \mathrm{d}\mathbf{r}} \,. \tag{2.4}$$

While it is not possible to evaluate expressions like equation 2.4 through numerical integration, we may still find ensemble averages through MC. The next section will show how to use Metropolis MC to find such averages.

2.2.2 METROPOLIS MONTE CARLO

Throughout this work MC simulations are used to study the self-assembly. This section will summarise the Metropolis MC algorithm along with its notation and key concepts. These will be expanded in the more advanced cluster move algorithms that will follow.

In its most general definition, MC refers to a family of algorithms for evaluating integral of functions numerically through sampling the function at random points picked from a uniform distribution.[106,107]

The expressions for thermodynamic properties of classical systems are often of the form of equation 2.4. Estimating integrals of this type can be challenging because many of the configurations make only a very small contribution to the result. The solution is to use importance sampling, where configurations are not inspected with equal probability, and instead the sampling of phase space is biased to prefer regions that make significant contributions to the integral. The most common example of importance sampling in chemical MC simulations is the Metropolis method.[108–110] In fact in physical sciences MC almost always refers to Metropolis MC.

For a classical, canonical system at equilibrium the trialling of states should be biased according to their Boltzmann population, given by

$$N(\mathbf{r}) = \frac{\exp\left[-\beta U(\mathbf{r})\right]}{\int \exp\left[-\beta U(\mathbf{r})\right]\, \mathrm{d}\mathbf{r}}\,. \tag{2.5}$$

The integral in the denominator of this expression is the canonical partition function (equation 2.3), which we have already deemed too costly to evaluate. However, as the partition function is constant, states may simply be sampled according to their relative Boltzmann populations (the numerator in equation 2.5). Sampling state according to this bias is Metropolis MC.

To generate trial configurations with frequencies proportional to their Boltzmann populations, a Markov chain of states is established. Typically this is achieved by ensuring that the system may transition from any state μ to any other ν through a finite number of MC moves. This is the condition of ergodicity. In order to correctly sample the relative equilibrium populations of each state, once equilibrium has been reached the total rate of leaving a state, through all routes, must be equal to the total rate of entering a state, through all routes. This requirement is called balance. However, it is in general easier to impose the more strict (but not necessary) condition of detailed balance.

Detailed balance is achieved when the rate of transition for a state μ to any other state ν is equal to the reverse rate, *i.e.*

$$N(\mu)M^{\mu\to\nu}W_{\mathrm{acc}}^{\mu\to\nu} = N(\nu)M^{\nu\to\mu}W_{\mathrm{acc}}^{\nu\to\mu}\,. \tag{2.6}$$

Here $M^{\mu\to\nu}$ is the probability of proposing a move from state μ to state ν and $W_{\mathrm{acc}}^{\mu\to\nu}$

is the probability of accepting that move. Typically in MC the geometry of moves are constructed to be symmetric so that $M^{\mu\to\nu} = M^{\nu\to\mu}$. Therefore, detailed balance ensures that the rate of leaving a state through any particular route is the same as entering a state by the same route, which means that balance is satisfied.

If the M terms are symmetric, using the definition of N from equation 2.5 the ratio of acceptance rates may be expressed by

$$\frac{W_{\text{acc}}^{\mu\to\nu}}{W_{\text{acc}}^{\nu\to\mu}} = \frac{N(\nu)}{N(\mu)} = \exp\left[-\beta\left(U(\nu) - U(\mu)\right)\right] . \tag{2.7}$$

Metropolis proposed the following form for W_{acc} consistent with equation 2.7

$$W_{\text{acc}}^{\mu\to\nu} = \min\left\{1,\, \exp\left[-\beta\left(U(\nu) - U(\mu)\right)\right]\right\} , \tag{2.8}$$

which is 1 if the change in energy from μ to ν is negative, and < 1 otherwise. Therefore, by proposing moves that are accepted according to equation 2.8 the equilibrium distribution of states is generated. The ensemble average $\langle A \rangle$ of a system may be calculated by an unweighted average of A over the states produced in the MC simulation.

2.2.3 Cluster Moves and the Troisi Algorithm

Conventional single move MC struggles when particles become strongly associated, such as during self-assembly. Using only single moves, aggregates must be moved through a series of cooperative moves, with the continual breaking and reforming of links. This makes diffusion and assembly of aggregates slow and uncontrollable. Cluster moves allow for collective motion of a group of particles, which makes simulation of systems with strong association much more efficient.

Early examples of cluster move algorithms include the Swendsen–Wang algorithm[111] and the Wolff algorithm,[112] both for simulation of spin glasses, a classic MC problem. In both algorithms the interactions between particles and their neighbours are used to divide the system into clusters. The ideas introduced in these algorithms have been extended by Troisi *et al.*, who aimed to improve the efficiency of thermodynamic sampling in strongly associating systems.[113] The Troisi algorithm represents significant progress in the simulation of strongly associating systems. It makes no assumptions about the nature of the pair potential and may be applied to systems on or off lattice. It can be applied to pair potentials featuring hard repulsions and angularly dependent terms. Furthermore, the al-

gorithm is based on an intuitive definition of a cluster based on the interaction energy of pairs of particles. Therefore, the algorithm evolves the system by moving strongly bound fragments which we would naturally consider clusters.

In any cluster move scheme the manner in which clusters are constructed and chosen to be moved must be considered when ensuring that detailed balance is observed. The new condition of detailed balance is

$$N(\mu)M^{\mu\to\nu}(C)P(\mu,C)W_{\mathrm{acc}}^{\mu\to\nu} = N(\nu)M^{\nu\to\mu}(C)P(\nu,C)W_{\mathrm{acc}}^{\nu\to\mu}\,, \tag{2.9}$$

where $P(\mu, C)$ is the probability of constructing cluster C in state μ and $M^{\mu\to\nu}(C)$ is the probability of picking proposing the move from μ to ν involving C. Moves are accepted with the probability

$$W_{\mathrm{acc}}^{\mu\to\nu} = \min\left\{1, \frac{P(\nu,C)}{P(\mu,C)}\frac{M^{\nu\to\mu}(C)}{M^{\mu\to\nu}(C)}\exp\left[-\beta\left(U(\nu)-U(\mu)\right)\right]\right\}. \tag{2.10}$$

As the terms relating to the probability of building and moving cluster C appear in the acceptance criterion, it is necessary to evaluate these probabilities when deciding whether to accept a move or not.

In the scheme proposed by Troisi *et al.*, clusters are built by creating links between a pair of particles i and j with the probability

$$\pi_{ij} = 1 - \exp(\beta_{\mathrm{f}} u_{ij}) \tag{2.11}$$

where β_{f} is a fictitious reciprocal temperature, and u_{ij} is the interaction energy between particles i and j.

As particles within a cluster are moved collectively, the pairwise energy of particles within the cluster remain the same in states μ and ν. The probability of linking these particles therefore remains the same. Similarly the probability of linking all particles external to the cluster remains the same. Therefore, when considering the ratio of probabilities of forming the cluster C in equation 2.10 it is only required to consider the interaction energy between pairs of particles where one member is within the cluster, and one outside. This set of pairs defines the boundary B. A schematic of an example of a cluster move in the Troisi algorithm showing the boundary is given in figure 2.1.

In each step in the Troisi algorithm a particle is picked at random. This particle and all those bound to it are moved together. Therefore, the probability of choosing a particular

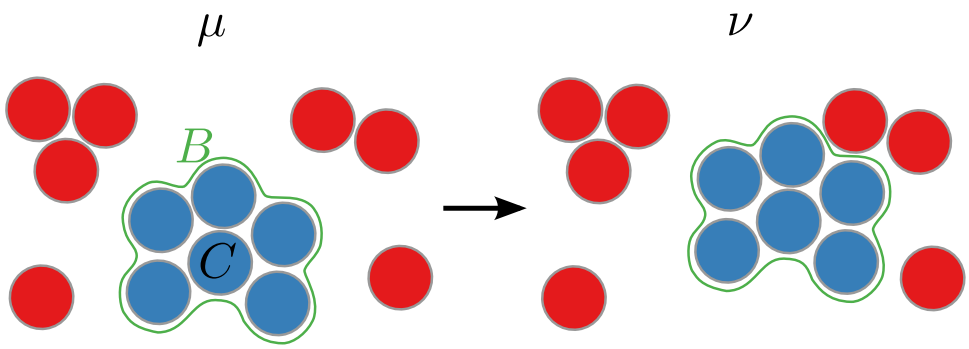

Figure 2.1: An example of a typical cluster move in the Troisi algorithm. The cluster C is formed from the blue particles. These particles are likely to link due to their strong interactions. Red particles, not belonging to C are unlikely to link due to their weak interactions with the members of C. The green line defines the boundary B. Between configurations μ and ν only interactions across B change.

cluster to move is proportional to its size, which is unchanged after a move. Hence, if the geometry of moves are symmetric then $M^{\mu\to\nu}(C) = M^{\nu\to\mu}(C)$ and the ratio of these terms in the acceptance criterion is equal to one.

Combining the linking probability (equation 2.11) and the symmetry of the M terms, the acceptance rate in the Troisi algorithm is given by

$$\begin{aligned} W_{\text{acc}}^{\mu\to\nu} &= \min\left\{1, \prod_{(i,j)\in B} \exp\left[\beta_{\text{f}}(u_{ij}^{(\nu)} - u_{ij}^{(\mu)})\right] \exp\left[-\beta\left(U(\nu) - U(\mu)\right)\right]\right\} \\ &= \min\left\{1, \exp\left[\left(\beta_{\text{f}} - \beta\right)\left(U(\nu) - U(\mu)\right)\right]\right\} . \end{aligned} \tag{2.12}$$

The simplification in the above expression may be made as the change in energy between pairs belonging to the boundary accounts for the total energy change in moving from μ to ν.

The linking probability in the Troisi algorithm is an elegant choice as it results in a simple acceptance criterion. However, this algorithm makes no attempt to produce realistic diffusion of clusters, which is important for the simulation of self-assembly. Furthermore, as the linking procedure considers only a static energy, particles tend to move together once they are interacting. For example, if two particles become loosely bound, they will tend to move collectively. This is true even when an individual move of one of the pair could relax the structure and reduce the energy of the system. Individual moves are not constructed to be dynamically reasonable. In the next section we shall look at a MC cluster move algorithm that does attempt to propose dynamically realistic moves.

2.2.4 VIRTUAL MOVE MONTE CARLO

The virtual move Monte Carlo (VMMC) algorithm of Whitelam and Geissler (WG)[114–116] is a cluster algorithm that seeks to overcome the problems of using single particle moves while also producing physically meaningful trajectories. In contrast to the Troisi algorithm, which chooses a move independently of the cluster, VMMC builds a cluster appropriate to the geometry of the move. The algorithm recruits particles to the cluster with a probability that depends on the difference in energy of the move with and without each successive particle in the cluster. Particles will tend to move in concert when this will create a more stable configuration. VMMC thereby implicitly accounts for forces and torques generated by the potential without the need for derivatives to be calculated. The VMMC algorithm therefore produces natural collective motion that allows proper internal relaxation of aggregates.[114]

The ability of the algorithm to efficiently build and relax aggregates, as well as the dynamic-like trajectories it generates, has ensured the popularity of VMMC. In particular the algorithm is commonly applied to the simulation of patchy particles where angularly dependent potentials make the use of other dynamic techniques such as MD difficult. Wilber *et al.* used the VMMC algorithm to simulate their model of patchy colloidal spheres.[71,72] The dynamic-like trajectories allowed them to identify different pathways to target assembly, including a low-temperature budding mechanism where correct targets broke off from disordered aggregates. VMMC has also been shown to produce pathways and rates that are comparable with those from Langevin dynamics in a coarse-grained model of DNA.[117]

The original version of the VMMC algorithm[114,115] begins by selecting a seed particle randomly and generating a trial move. Particles are then recruited to the cluster in an iterative testing procedure, beginning with the neighbours of the seed and ending when all neighbours of all particles in the cluster have been tested. A link between a pair of particles i and j in the initial configuration μ is formed with the probability

$$p_{ij} = I_{ij}^{(\mu)} \max\left\{0,\, 1 - \exp\left[\beta u_{ij}^{(\mu)} - \beta u_{i'j}^{(\mu)}\right]\right\} . \tag{2.13}$$

Here $I_{ij}^{(\mu)} = 1$ if particles i and j are interacting in configuration μ and 0 otherwise, $u_{ij}^{(\mu)}$ is the pairwise energy of i and j in configuration μ, and $u_{i'j}^{(\mu)}$ is the pair energy after the individual move of i. The expression within the exponential therefore accounts for the difference in energy between an individual move of particle i, and a collective move of

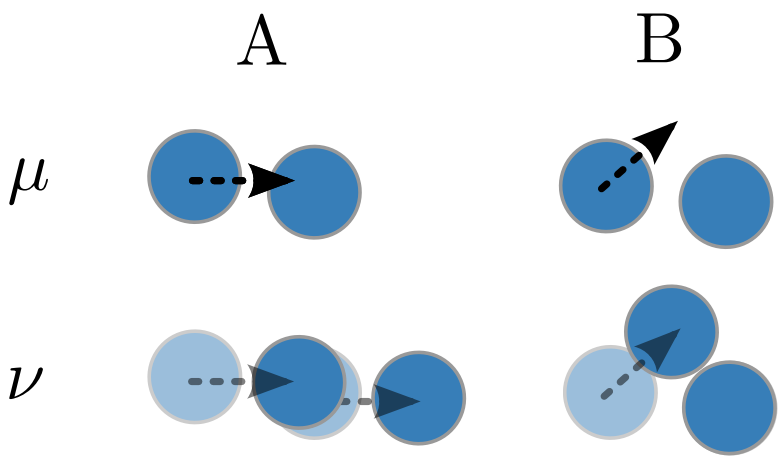

Figure 2.2: Examples of the VMMC algorithm building clusters appropriate to the move. Examples A and B both start in the same configuration μ but have different displacements resulting in collective motion for A and an individual move for B. In case A the proposed move would result in the seed particle overlapping with its neighbour and so the two particles move collectively resulting in configuration ν. In configuration ν the translucent circles and arrows show the original configuration and the motion taken respectively. In contrast example the move proposed in example B results in a stronger interaction between the two particles, so an individual move of the seed is conducted.

both i and j.

The linking probability, equation 2.13, ensures that a link may only form if a collective move of i and j is energetically beneficial relative to an individual move of i (when $u_{ij}^{(\mu)} < u_{i'j}^{(\nu)}$). The more favourable a collective move is the more likely the link is to form. However, a link will never form when a move of i relative to j would be energetically beneficial (when $u_{ij}^{(\mu)} > u_{i'j}^{(\nu)}$). It is this dependence of the linking probability on energy gradients that allows VMMC to propose sensible relaxation moves. A particularly attractive feature of equation 2.13 is that if a proposed move would cause the hard cores of particles to overlap then the particles will always move collectively. In this manner particles may "push" each other. Examples of these features are shown in figure 2.2. In the Troisi algorithm there is no consideration of whether a move will cause particles to overlap until considering whether to accept a move or not. Therefore, if a move does cause an overlap the move must be rejected with no opportunity to also move the overlapping particles.

Proposed moves in the VMMC algorithm are accepted with probability

$$W_{\mathrm{acc}}^{\mu\to\nu|R} = \Theta(n_{\mathrm{c}})D(C)\min\left\{1,\ \prod_{(i,j)n\leftrightarrow o}\exp\left[-\beta(u_{ij}^{(\nu)}-u_{ij}^{(\mu)})\right]\right.$$
$$\left.\prod_{(i,j)\in F}\frac{q_{ij}(\nu\to\mu)}{q_{ij}(\mu\to\nu)}\prod_{(i,j)\in L}\frac{p_{ij}(\nu\to\mu)}{p_{ij}(\mu\to\nu)}\right\}. \tag{2.14}$$

Here, $(i,j)n \leftrightarrow o$ indicates all pairs (i,j) that are not interacting in configuration μ but have a positive energy of interaction in configuration ν or *vice versa*. The sets F and L are the pairs of links that failed and succeeded to form, respectively. $q_{ij}(\nu \to \mu)$ is the link failure factor ($q_{ij} = 1 - p_{ij}$) between particles i and j for the move that takes configuration ν to configuration μ, that is, the reverse of the proposed move.

The probability of forming links between a pair of particles are not equal for forwards and backwards moves. Because of this the probability of constructing a cluster and proposing its move (P and M terms respectively in equation 2.9) are not equal in the forwards and backwards directions in VMMC. The acceptance rate accounts for these differences with the p_{ij} and q_{ij} factors.

The R in the acceptance probability $W_{\mathrm{acc}}^{\mu\to\nu|R}$ of equation 2.14 indicates that this is the acceptance rate for a particular realisation R of the cluster C. A realisation of a cluster represents not only the particles belonging to the cluster, but also the specific links that have formed between particles in order to connect them (figure 2.3). In general the same set of particles may be connected by a different network of links. The extra specification of ensuring that the transition rate is equal for forward and reverse moves between any two configurations, through every pathway, and every realisation of the cluster is termed super detailed balance by WG. The condition of super detailed balance is given by

$$\frac{W_{\mathrm{acc}}^{\mu\to\nu|R}}{W_{\mathrm{acc}}^{\nu\to\mu|R}} = \frac{N(\nu)}{N(\mu)}. \tag{2.15}$$

The acceptance rate in equation 2.14 is only for a particular realisation of cluster C because it considers the linking probabilities in forward and reverse moves for just one realisation of formed links (the set L) and failed links (the set F). In order to account for all realisations, it would be necessary to enumerate all the possible realisations of clusters C. It is more convenient, and efficient, to consider only the realisation that has been

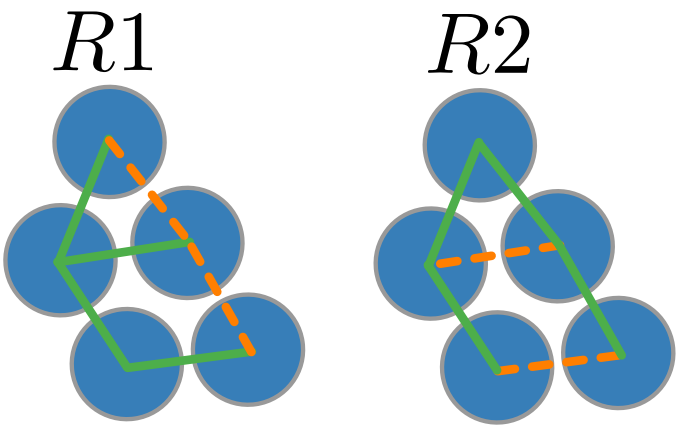

Figure 2.3: Two different realisations ($R1$ and $R2$) of the same cluster C. Formed links are represented by solid green lines, and failed links by dashed orange lines. In both cases the cluster is the same as all particles are joined by a network of formed links. However the realisations are different because sets of formed and failed links (L and F in equation 2.14 respectively) are not identical. These differences account for different p_{ij} and q_{ij} factors in the acceptance criterion, and hence difference acceptance probabilities.

created when determining the acceptance probability.

The two prefactors of the $\min$ function in equation 2.14 are included in order to try and impose reasonable dynamics on the cluster moves. The first of these terms $\Theta(n_c)$ is equal to 0 if n_c, the number of particles in the cluster, is larger than a cut-off and 1 otherwise. WG propose that if the cut off is chosen from an appropriate distribution this factor should correctly suppress the motion of large clusters. Such a term is required for two reasons. First, in Brownian motion large aggregates should diffuse more slowly than smaller ones. Second, when a group of strongly bound particles is present in a simulation they are likely to be linked in the cluster building process. Large aggregates are consequently more likely to be moved as they contain more particles that could be picked as the seed. Therefore, it is necessary to compensate for the inherently higher probability of picking a large aggregate.

The second prefactor, $D(C)$ is designed to approximate realistic dynamics by taking account of the geometry of the cluster with respect to the move. In principle, the form of this prefactor could be modified in order to make the dynamics of the VMMC algorithm match a variety of scenarios. For example, in their original work WG proposed a form that tries to account for drag by considering the cross-sectional area of the clusters being moved.[114]

The VMMC algorithm produces dynamically meaningful moves by considering the

difference between individual and collective moves, unlike the Troisi algorithm, which just tends to link strongly interacting particles. However, this is at the cost of a much more complicated acceptance criterion, featuring potentially very large products that can reduce the acceptance probability. Hence, the behaviour of the algorithm is not very predictable. The proposed way of producing realistic diffusion is not very elegant, relying on rejecting moves that are unrealistically large. This is quite an expensive proposition as as significant amount of calculation must be done to build a cluster, all of which is wasted if the move is rejected. This is the cost of picking a move before building a cluster, which is a prerequisite for building dynamically sensible clusters. The former of these two drawbacks has been addressed by WG in a modification to the VMMC algorithm, which will be discussed in the next section.

2.2.5 SYMMETRISED VIRTUAL MOVE MONTE CARLO

Symmetrised virtual move Monte Carlo (SVMMC)[116] seeks to simplify the acceptance criterion of the original VMMC algorithm. The simplification also has the benefit of removing a number of factors from from the acceptance expression that would often reduce the acceptance probability to near zero, particularly for larger clusters as we shall see in section 3.3

In SVMMC, particles are first "pre-linked" with probability p_{ij}, the same linking probability as in the original VMMC algorithm (equation 2.13). If pre-linking succeeds then full links may be formed with probability

$$\min\left\{1,\ \frac{p_{ij}(\nu \to \mu)}{p_{ij}(\mu \to \nu)}\right\}. \tag{2.16}$$

The overall probability of forming a full link is therefore given by

$$P_{ij}(\mu \to \nu) = p_{ij}(\mu \to \nu)\min\left\{1,\ \frac{p_{ij}(\nu \to \mu)}{p_{ij}(\mu \to \nu)}\right\}. \tag{2.17}$$

Particles that fail to pre-link are said to have failed outright. If a pair of particles pre-link, but fail to form a full link, then the link is said to be frustrated. The linking probability in equation 2.17 is constructed so that the probability of forming a full link in the forward and reverse moves is identical. This means that the linking probability satisfies $P_{ij}(\mu \to \nu)/P_{ij}(\nu \to \mu) = 1$, which is why the algorithm is described as symmetrised.

Whereas in the original VMMC algorithm ratios of linking probabilities could re-

duce the acceptance probability (equation 2.14), in SVMMC the new definition of a link ensures this is not the case. Furthermore, the linking probabilities of unformed links (whether failed outright or frustrated) within the cluster are also symmetric. So neither do these terms reduce the acceptance probability.[116] SVMMC therefore addresses a significant drawback of the VMMC algorithm, where a link that has formed or failed within the cluster could cause a move to be rejected if the same is unlikely in the reverse move. This is in spite of this link being redundant as all particles still belong to the cluster whether it is formed or not. The only linking factors that may reduce the acceptance probability are those for failed and frustrated links where one particle belongs to the cluster and one does not.

The simplified acceptance probability of SVMMC is given by

$$W_{acc}^{\mu\to\nu|R} = \Theta(n_{\mathrm{c}})D(R)\min\left\{1,\ \delta_{\mathrm{f}}\prod_{(i,j)n\leftrightarrow o}\exp\left(-\beta(u_{ij}^{(\nu)}-u_{ij}^{(\mu)})\right)\right\}, \tag{2.18}$$

where all terms have the same definitions as before. δ_{f} is a factor taking into account the frustrated links at the boundary of the cluster. It is 1 if there are no frustrated links between the cluster and the rest of the system, and 0 otherwise. Therefore, this algorithm does not needlessly reject moves where the original algorithm would. If a frustrated link does form on the boundary of a cluster, there is also still a chance that the particle outside of the cluster may become incorporated by another series of links. This frustrated link would, in this case, become internal and not require the rejection of the move.

2.3 BASIN-HOPPING

In chapter 6 it will be necessary to optimise a function of many variables to find its lowest value, a procedure called global optimisation. This section will briefly discuss optimisation and use the example of geometry optimisation to introduce basin-hopping,[95,101] a very successful method for global optimisation.

Finding the global minimum of a potential energy landscape is not a simple problem. As discussed in section 2.2.2 the potential energy of a system has a very large number of dimensions as it depends on the positions (and sometimes orientations) of all particles. Because of this, searching configuration space exhaustively for the minimum would take an unreasonably long time. MC and MD may be used to efficiently sample the landscape according to the Boltzmann population of states (equation 2.5). However, these are not

appropriate for identifying the global minimum because at low temperatures, where the global minimum should be the predominant configuration, the simulations tend to get stuck in local minima as there is insufficient thermal energy to overcome kinetic barriers.

One approach to overcome this problem is simulated annealing. In a simulated annealing experiment the temperature is slowly reduced. The intention is to start at a temperature where kinetic barriers are easily overcome, after which the slow reduction in temperature should guide the system into the deep potential well of the global minimum. This approach may be enhanced by simulating multiple replicas of the system at different temperatures that periodically attempt to swap temperatures using MC-like moves. This approach is called replica exchange.

Despite the effort to try and avoid kinetic traps, both simulated annealing and replica exchange are prone to get stuck. This is especially true when the potential energy surface has multiple funnels (a series of downhill pathways to a minimum) leading to different minima. This is because it is often true that the temperature at which the global minimum becomes the predominant state is lower than the temperature at which kinetic barriers become insurmountable. Basin-hopping works by transforming the energy landscape in a manner that broadens the transitions between minima. The global minimum is therefore more likely to be occupied at temperatures where the (new) barriers between minima are surmountable.[118]

Basin-hopping combines energy minimisation with a MC search in phase space. Each step in a basin-hopping simulation consists of a move in parameter space, followed by the determination of the local minimum. The move to the new configuration is then accepted as in Metropolis MC using the criterion

$$W_{\text{acc}} = \min\left\{1, \exp\left(-\beta \Delta U_{\min}\right)\right\} , \tag{2.19}$$

where $\Delta U_{\min}$ is the difference between the minimised energies of the last configuration and the new configuration. β is a parameter which acts like the reciprocal temperature in a Metropolis MC simulation. β does not relate to a physical temperature, but can be understood as a parameter which controls the probability of accepting a move that raises the energy. Generally β is set to allow efficient exploration of the landscape.

Basin-hopping only moves between minimised configurations of the objective function, and may therefore be thought of as a MC simulation of the minima of the system. Basin-hopping is an efficient method for finding the global minimum as the modified

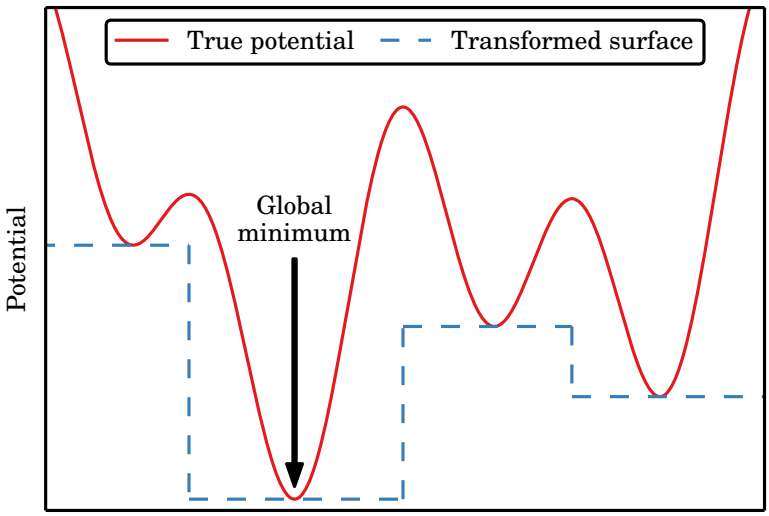

Figure 2.4: A demonstration of how a true potential is mapped to a potential without barriers between minima in a basin-hopping simulation. The entire basin of attraction of each minima is assigned an energy equal to the minima.

landscape it samples removes the kinetic barriers between local minima, as shown in figure 2.4. Moves within the basin of attraction of one minima, or to a lower lying minima will always be accepted. Only moves to the basin of a higher energy minimum may be rejected, and so if the global minimum is surrounded by other low lying minima the algorithm will tend to move towards it.

CHAPTER 3

PATCHY PARTICLE MODEL AND DYNAMICAL MONTE CARLO ALGORITHM

3.1 INTRODUCTION

This work aims to study general features of self-assembly that are applicable to a large range of self-assembling systems, rather than focusing on the detail of any one case. In particular, the effect of the design of building blocks on assembly will be explored. For example, we would like to address how design can be used to optimise assembly or guide assembly through distinct pathways, such as hierarchical or addressable routes. We would also like to explore the limits of reliable assembly for a given complexity of building block.

Key to addressing these questions is the generic and versatile model that will be presented in this chapter. The model consists of hard cubic particles with flat and featureless faces. The faces may be patterned with attractive patches to create binding interfaces. Interactions between particular pairs patches may be tuned using an interaction matrix. Therefore, it is possible within this model to create a huge variety of designs to compare and contrast, from single component clusters through to the fully addressable limit. Using the same model to represent the range of approaches ensures that meaningful comparisons can be made between different designs.

The nature of the model places a number of requirements on the simulation algorithm. Our model uses hard bodies, and patches interact through an angularly attenuated potential. As discussed in chapter 2, MD is not suitable for simulating potentials with these features, and so MC will be used to study our model. In self-assembly, the particles must bind strongly and form aggregates. Therefore, the MC algorithm must deal efficiently

with the collective motion associated with diffusion and assembly of clusters. Furthermore, to study the dynamical aspects of self-assembly the MC algorithm must also move aggregates in the system in a dynamically reasonable manner.

In the previous chapter two existing MC algorithms were introduced, which featured cluster moves for the acceleration of simulation of strongly associating systems. The Troisi algorithm builds clusters based on their pairwise energy using a fictitious temperature to determine the probability of forming links. This scheme is attractive as it offers a simple acceptance criterion and accelerates collective motion. VMMC and its improved variant SVMMC attempt not only to produce collective motion but to also propose dynamically reasonable moves. This is achieved by building clusters based on the change of energy between pairs of particles over the course of individual and collective moves. Particles will therefore tend to move along energy gradients. However, this is at the cost of a more complicated building procedure and acceptance criterion.

In section 3.2 the patchy cube model will be defined. The form of the potential will be presented, including detail of how to detect the overlap of the hard cores of the particles, as well as the form of the attractive potential between surface patches. Subsequently, in section 3.3 the algorithms from the previous chapter will be applied to the model and their behaviour discussed. We are particularly interested in whether the existing algorithms meet, or can be adapted to meet, our requirements of producing collective moves for assembly and relaxation efficiently and moving aggregates of various sizes in a dynamically reasonable manner.

The shortcomings of the existing MC schemes will be addressed. Neither of the algorithms will have all of the features desired. However, the attractive features of both the Troisi and SVMMC algorithms are found to be quite complementary. It was necessary to develop a hybrid algorithm as no single existing algorithm satisfied our requirements. In section 3.4 we will present our new dynamical MC algorithm for simulating self-assembly. The algorithm combines aspects of both the Troisi algorithm and SVMMC to create a hybrid scheme that can handle both assembly and diffusion.

3.2 Patchy Particle Model

In the patchy cube model, complementary interactions may be made by ensuring that the patches overlap when the faces are brought into contact. Similarly, we can discourage particles from binding by ensuring that few patches are able to overlap well when the faces

are touching. The interaction of particles through patterned interfaces rather than via a single site on each particle is similar to protein interactions, where binding involves an interface between the surfaces of the proteins and determines their quaternary structure.[119] The model is also reminiscent of colloids fabricated with patchy interactive sites.[58,60,63]

Although cubic particles may seem somewhat artificial in the context of proteins, there are now many examples of synthetic routes to colloidal cubes.[44,54,120] These developments have stimulated both experimental and computational investigations of the self-assembly of cubic particles.[12,55] Currently, experimental cubes have not been decorated with patches. However, given the interest in spherical colloids with patchy interactions, with theory and simulation encouraging experiment, patchy cubes do not seem like an unrealistic proposition. Furthermore, the self-assembling nanoparticles developed by Liu *et al.*[64] used a rigid DNA frame to give the particles four interactions in a square arrangement. It is feasible that this scheme could be extended to create cubic cages around spherical nanoparticles. Besides their relation to real systems, the flat faces of the cubes allow us to study in detail the effect of patch patterns at a binding interface. With a spherical particle, for example, the curvature of the surface would severely limit the size of the interface possible.

In order to detect overlap of the hard particle cores in simulations, cubes are treated as oriented bounding boxes.[121] This approach is commonly used in computer graphics to detect the overlap of objects. The more general algorithm for detecting the overlap of two orthorhombic boxes is then simplified to the case of cubes with edge d, following the same approach used for simulations of purely repulsive hard cubes.[56,103] The procedure for detecting overlaps of cubes is detailed in appendix 3.A.

An alternative approach to modelling the cubes as a single rigid body is to construct the particles as a union of repulsive spheres. The advantage of such an approach is that the resulting particles may be simulated using MD, similarly to the models of patchy particles built from rigidly connected spheres seen in section 1.2.2. Such models have been used in early studies of cubic particles.[122] However, they produce artefacts due to the corrugated surface of the particles.[103] The smooth faces of the hard cubic particles allow us to easily use the faces as a blank canvas for designing interactions, whereas a rough surface would introduce a bias for certain configurations unrelated to the patch patterns.

The attraction between patches on the faces of the hard cubes is a pairwise Morse potential with an angular attenuation, which helps ensure that particles orientate with faces parallel when binding. This also gives each patchy interaction a width that distinguishes

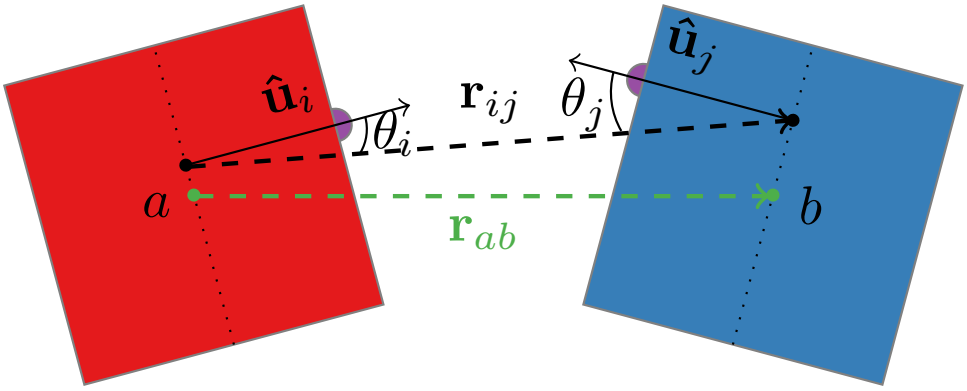

Figure 3.1: Schematic representation of the interaction between patches i and j on two cubes a and b, showing the definition of the angles θ_i and θ_j.

patches and stops interactions blurring, which would happen if an isotropic potential was used. The Morse potential between two sites corresponding to the patches i and j may be written

$$V_{ij}^{\mathrm{M}}(r_{ij}) = \varepsilon_{ij}\left[e^{-2\alpha(r_{ij}-d)} - 2e^{-\alpha(r_{ij}-d)}\right] , \tag{3.1}$$

where r_{ij} is the distance between the sites and α is a parameter controlling the range of the potential. We have chosen $\alpha = 6$, which gives a curvature at the minimum of the potential that matches that of the Lennard-Jones potential. ε_{ij} is the strength of interaction between patches i and j. In general this strength may vary between pairs of patches and so the subscripts ij are included on the function V_{ij}^{M} as well as on its argument r_{ij}. The Morse site representing a given patch is embedded inside the cube at a depth $d/2$ from the surface with which the patch is associated (figure 3.1). The distance dependence of equation 3.1 therefore ensures the minimum of the potential occurs when two particles faces are in contact and the patches overlap perfectly at the interface.

The angular part of the potential takes the form of a Gaussian attenuation

$$V^{\mathrm{ang}}(\hat{\mathbf{r}}_{ij}, \hat{\mathbf{u}}_i, \hat{\mathbf{u}}_j) = \exp\left(-\frac{\theta_i^2 + \theta_j^2}{2\sigma^2}\right) , \tag{3.2}$$

where $\hat{\mathbf{r}}_{ij}$ is the unit vector pointing from patch i to j. $\theta_i = \cos^{-1}(\hat{\mathbf{r}}_{ij} \cdot \hat{\mathbf{u}}_i)$ and $\theta_j = \cos^{-1}(\hat{\mathbf{r}}_{ji} \cdot \hat{\mathbf{u}}_j)$ are the angles between the unit surface normals $\hat{\mathbf{u}}_i$ and $\hat{\mathbf{u}}_j$ of the patches and the inter-patch vector (figure 3.1). The width of the Gaussian, σ, controls the directionality of the patches, by determining the rate at which the potential decays with deviation from the ideal alignment. In this work σ has been fixed at 0.2.

Describing a patch by an embedded attractive site, with an angular attenuation has been implemented before in a Lennard-Jones-based potential for patchy spheres.[70–72] The patches may be understood as conical regions of interaction extending through the surface of the particles. The width of the cone and hence directionality of the patch can be controlled by σ, independently of the range of attraction. Embedding the attractive sites rather than placing them at the surface of the particles ensures that the cones of interaction have finite width at the surface. Furthermore, embedding the patches places the minimum of the attractive potential when particles are in contact. The repulsion of the Morse potential is superseded by the hard bodies of the particles.

The Morse potential is truncated at a distance $r_{ij} = 2d$. To avoid a discontinuity at the cut-off, the potential is shifted by $V_{ij}^{\mathrm{M}}(2d)$ and scaled to recover a well depth of ε_{ij}. The overall form of the attractive potential between two patches i and j is therefore

$$V_{ij}^{\mathrm{patch}}(\mathbf{r}_{ij}, \hat{\mathbf{u}}_i, \hat{\mathbf{u}}_j) = \left[\frac{V_{ij}^{\mathrm{M}}(r_{ij}) - V_{ij}^{\mathrm{M}}(2d)}{\varepsilon_{ij} - V_{ij}^{\mathrm{M}}(2d)}\right] \Theta(2d - r_{ij}) V^{\mathrm{ang}}(\hat{\mathbf{r}}_{ij}, \hat{\mathbf{u}}_i, \hat{\mathbf{u}}_j)\,, \tag{3.3}$$

where $\mathbf{r}_{ij} = r_{ij}\hat{\mathbf{r}}_{ij}$ and Θ is the Heaviside step function. The total interaction between two cubes a and b is given by the sum of the interactions between the patches $i \in a$ and $j \in b$ on each of them,

$$V_{ab}^{\mathrm{cube}}(\mathbf{r}_{ab}, \boldsymbol{\Omega}_a, \boldsymbol{\Omega}_b) = \sum_{i \in a} \sum_{j \in b} V_{ij}^{\mathrm{patch}}(\mathbf{r}_{ij}, \hat{\mathbf{u}}_i, \hat{\mathbf{u}}_j) \Delta_{ij}(\hat{\mathbf{r}}_{ij}, \boldsymbol{\Omega}_a, \boldsymbol{\Omega}_b)\,, \tag{3.4}$$

where $\boldsymbol{\Omega}_a$ represents the orientation of cube a and $\mathbf{r}_{ab}$ is the vector position of the centre of cube b with respect to that of cube a. $\Delta_{ij} = 1$ if the faces associated with patches i and j are the closest pair of most aligned faces of the two cubes, and 0 otherwise. Δ_{ij} therefore acts as an angular truncation of the potential. We do not expect any artefacts from this term as the strength of the interaction is negligible at the point of truncation (typically less than $10^{-6}\varepsilon_{ij}$).

3.3 Dynamics and Relaxation in Cluster Algorithms

A cluster move based purely on interaction energy, such as the Troisi algorithm (section 2.2.3), will help accelerate the simulation of systems with strongly associated particles, as collective motion occurs in a more efficient way. It is, however, not perfect because once particles become bound they will tend to move collectively rather than in-

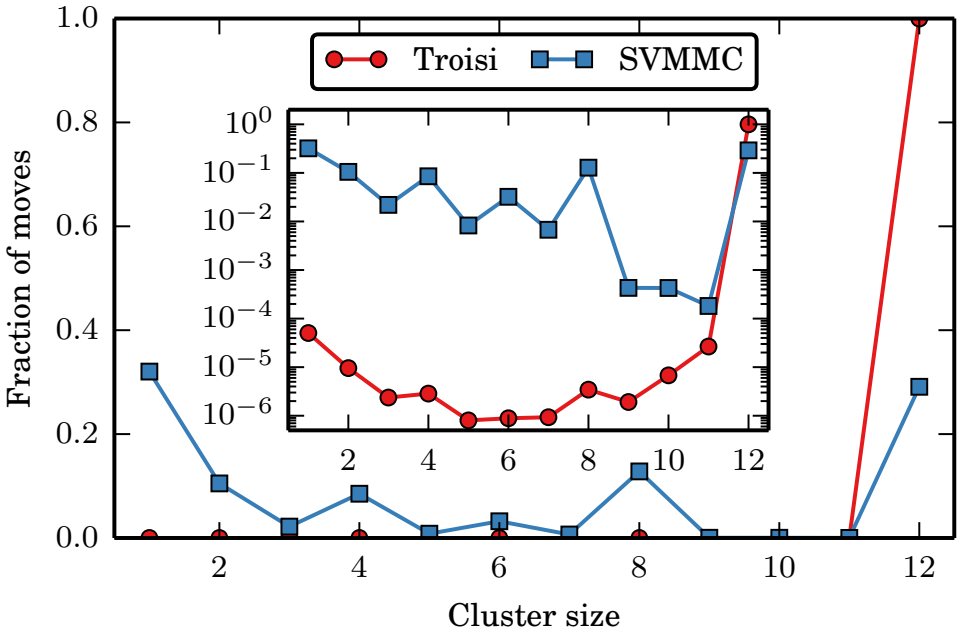

Figure 3.2: The normalised frequency of accepted moves as a function of size of cluster being moved for the Troisi and SVMMC algorithms. Each simulation featured a single, pre-assembled cluster consisting of twelve particles. The simulations were conducted at a low temperature to ensure that the cluster did not fall apart. Inset is the same data on a logarithmic scale. The peaks in frequency at 4, 6 and 8 for the SVMMC algorithm correspond to characteristic fragments of the cluster.

dividually, even if an individual move would be more energetically beneficial. This is due to the dependence of linking probability (equation 2.11) on a static pairwise energy rather than the energy gradient over a move, like the linking probability in VMMC. For this reason the Troisi algorithm moves whole aggregates successfully, but it struggles to relax the internal structure of a cluster, as that requires moving only some of the particles in an aggregate.

The authors of the VMMC algorithm proposed a protocol to improve relaxation in the Troisi algorithm by allowing for variation in the value of β_{f}.[114] If $\beta_{\mathrm{f}} = 0$ no links are formed, and the algorithm only performs single particle moves. In the opposite limit $\beta_{\mathrm{f}} \to \infty$ links between any pair of interacting particles will always form. They suggested varying β_{f} on a distribution $[0, \beta_{\mathrm{f,max}}]$ to ensure moves with a range of cluster sizes.

Figure 3.2 shows the distribution of accepted translation moves of different sized clusters for the Troisi and SVMMC algorithms in simulations featuring a pre-assembled cluster consisting of twelve of our patchy cube particles. The structure of the cluster is a $3 \times 2 \times 2$ array of cubes, shown in figure 3.3. This cluster was chosen as it is large enough

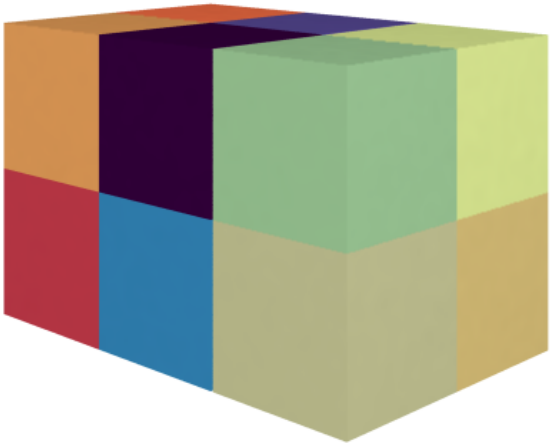

Figure 3.3: The cluster of twelve particles used to test the cluster move algorithms ability to propose collective moves.

to have non-trivial internal motion, for example the collective motion of the 2×2 or 3×2 faces. The size of the cluster also tests the ability of the algorithms to move clusters of a range of sizes, from monomer to the entire cluster. In all simulations the temperature was set low enough to ensure that the cluster remained intact for the duration of the simulation.

The result for the Troisi algorithm with β_{f} picked on the distribution $[0, \beta]$ where β is the reciprocal of the actual simulation temperature (red circles) shows a strongly unimodal profile. Accepted moves are almost always moves of the entire cluster. Moves of single particles are the second most common, and are roughly ten thousand times less frequent. This can be seen more clearly in the logarithmic insert of figure 3.2. Moves of intermediate size are very rare. The results demonstrate the difficulty that the Troisi algorithm has in suggesting collective relaxations, even with the stochastic β_{f} modification.

Under the same conditions, SVMMC showed a more equal distribution of cluster size. In figure 3.2, the blue squares represent the results for the SVMMC algorithm without any dynamical prefactors. The more even distribution demonstrates that SVMMC is indeed able to propose collective motion of fragments of an aggregate. These moves are important for the relaxation, and internal thermal fluctuations in aggregates. In particular, there is an increased probability of moves of fragments with 4, 6 or 8 particles. These sizes correspond to "slices" of the pre-assembled aggregate which we would expect to move quite freely due to a lack of steric constraints. It is therefore encouraging that the SVMMC algorithm does indeed propose the moves of these fragments with a relatively high probability. The SVMMC algorithm is able to generate moves of clusters of intermediate size

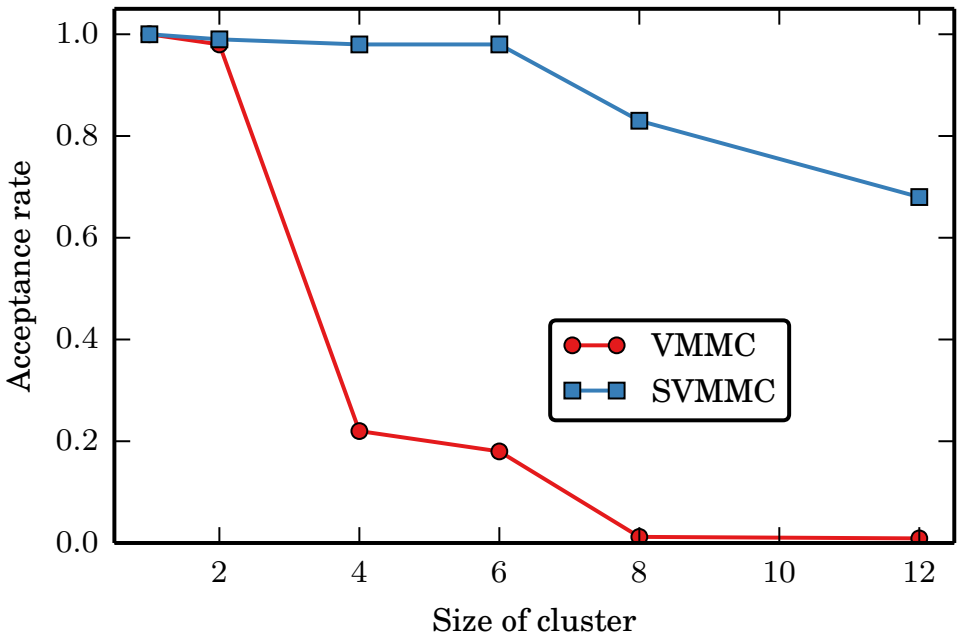

Figure 3.4: The mean acceptance rate over a simulation as a function of aggregate size for the VMMC and SVMMC algorithm's. In each simulation the temperature is low in order to ensure the aggregate does not break up. The move size was also large relative to the size of a particle to try and ensure moves were collective, recruiting all particles.

because of its recruiting procedure. As new members are recruited to the cluster based on the energy gradient over a move, when a proposed move can act as a relaxation it will tend to do so. In contrast, the Troisi algorithm, based on a static energy, will tend to always recruit particles when they interact strongly.

The average acceptance rates of the VMMC and SVMMC algorithms depend quite strongly on the size of cluster being moved. Figure 3.4 shows the mean acceptance rate of translation moves for a range of simulations featuring aggregates of different sizes. The temperature was set low to ensure clusters remained intact over the course of the simulation. A large step was used to encourage recruiting all particles to the cluster when generating a move. Therefore, this simulation tests the acceptance rate when almost all moves should be moves of the entire aggregate. For both algorithms, prefactors are excluded so that moves of larger clusters are not suppressed.

For both algorithms, the acceptance rate for monomer is trivially unity, and for a dimer the acceptance is close to unity. Beyond this, the acceptance rate of the VMMC algorithm falls dramatically, and not in a smooth manner. Despite promoting whole cluster moves, which may always be accepted, there are a significant proportion of rejected moves for

clusters larger than two particles. The acceptance rate (equation 2.14) contains probabilities of forming and failing links of the realisation in the reverse move. As the size of a cluster grows, the number of these factors increases. The likelihood of a forming a link that is very likely to fail in the reverse move (or *vice versa*) becomes higher. Therefore, the products of q_{ij} and p_{ij} factors approach zero, and the acceptance rate is greatly reduced. The VMMC algorithm therefore tends to unrealistically suppress the motion of larger clusters, even without using the prefactors that would lead to more moves of large clusters being rejected. We also see in figure 3.4 that moves of even moderately sized clusters (perhaps ten or so particles) are unlikely to be accepted.

The SVMMC algorithm shows better behaviour, with a more consistent acceptance rate. The change to the linking procedure in SVMMC means that for an isolated aggregate the move is always accepted as long as all particles are recruited to the cluster. Only frustrated links on the boundary of the cluster (or a move that increases the energy of the system) can cause a rejection (equation 2.18). The deviation from unity in figure 3.4 therefore demonstrates that the algorithm has difficulty in recruiting all particles to the cluster.

As we cannot achieve an even acceptance (or uneven but predictable acceptance) for a range of cluster sizes, it is not reasonable to attempt to control the dynamics of aggregates by prefactors as proposed by WG. Furthermore, the nature of moves proposed, and their acceptance, in the SVMMC algorithm vary dramatically with move size and temperature. For example, large moves tend to recruit all particles as it is likely particles would otherwise end up in an overlapping state, or in a state with a less favourable pairwise energy. In contrast, very small moves will generate mostly single particle moves. Therefore, while in the extreme limit of a large step size and low temperature reasonable translational dynamics may be achieved by rejecting unrealistically large moves, this scheme is not transferable across temperature and move size. This is particularly troubling as the limit in which the dynamics may be more reasonably controlled is also where there are far fewer interesting relaxation and assembly moves, and instead mostly whole cluster moves.

When we consider rotations as well as translations the situation becomes worse. Ideally it would be possible to control both translational and rotational diffusion and ensure that the quotient of the respective rates of diffusion is physical. In VMMC and SVMMC we should expect to encounter the same problems as with translation, where an unpredictable building procedure makes controlling rotational diffusion difficult. Furthermore,

as a move must be picked before building a cluster, when rotational moves are performed it is not possible to ensure that the rotation occurs about the centre of mass of the cluster. Because of this, rotational moves also contribute towards the translational diffusion of clusters in an unaccountable manner.

An alternative to simply rejecting moves that are deemed unrealistic, would be to scale the size of moves based on the size (and shape) of the cluster being moved. For example, the displacement of a large cluster could be reduced to account for the increased drag it would experience (and also the higher probability of it being picked for a move). However, such a scheme is difficult to implement in the VMMC and SVMMC algorithms as the geometry of a move must be chosen before building a cluster, due to the dependence of forming links on single and collective moves. WG proposed a method to rescale moves.[114] However, this modification, which involves considering distinct real and virtual moves, makes the algorithm significantly more complicated and computationally expensive and adds more elements, making the acceptance unpredictable. We therefore did not consider this a suitable way to control dynamics.

Although the Troisi algorithm was not intended to be dynamic, it does have a number of attractive features which allow us to use it to control dynamics. Unlike the VMMC algorithms, the cluster-building procedure is controllable through the parameter β_f. If $\beta_\mathrm{f} = \beta$ then the acceptance rate becomes unity (see equation 2.12) and so the algorithm is rejection-free (in the absence of an overlap of particles). A rejection-free algorithm may also be achieved in the limit $\beta_\mathrm{f} \to \infty$. Here, all links between interacting particles are accepted. In both of these cases moves are only rejected when it would move the system into a state where particles overlap. Also, as the linking procedure is independent of the move, a move appropriate to the size (and, if desired, geometry) of the cluster may be picked. This approach has two principal advantages over the VMMC and SVMMC schemes. First, it is computationally cheaper as moves are not rejected, so the expense of building a cluster is not wasted when a cluster is large. Second, rotations can be made about the centre of mass of the cluster. This means that rotational diffusion may be controlled independently and will not contribute to the translational diffusion of the system.

To conclude, this section has explored the implications of the cluster building procedures and acceptance criteria of Troisi, VMMC and SVMMC cluster moves on the relaxation and diffusion of an example cluster using the patchy cube model. We found that the Troisi algorithm, even with the modification proposed by Whitelam *et al.*, had

a very strong tendency to perform either moves of single particles or entire aggregates. This demonstration confirmed observations that the algorithm tended to produce poor relaxations of clusters. Once particles become strongly linked in the Troisi algorithm they will tend to move collectively. The relaxation and thermal motion of clusters is therefore severely neglected. The SVMMC algorithm produced a much more even distribution of move sizes. If the acceptance criterion (equation 2.18) is modified to not reject the moves of large clusters (the Θ prefactor) the distribution is more even. It is, therefore, possible to understand why the SVMMC produced the efficient collective relaxations of clusters, important to self-assembly.

However, where the SVMMC algorithm ultimately falls short is in producing realistic diffusive behaviour of aggregates. The failure is from two important sources. First, the acceptance rate for moves of aggregates of different sizes is unpredictable. This is because of the complicated nature of the linking procedure. Even for a given temperature and step size the acceptance is hard to predict. Moreover, the nature of moves proposed, and hence realised depends strongly on the temperature and step size. This means that the algorithm's behaviour is not consistent across a range of these parameters. For diffusion, the Troisi algorithm has advantages. Although the original algorithm does not contain any terms accounting for dynamics, it has a more predictable acceptance rate which can be made rejection free, and as moves are picked independently of clusters their size may be easily scaled.

Neither the Troisi nor the SVMMC algorithm therefore satisfy our requirements for a MC algorithm that efficiently assembles and relaxes aggregates using collective motion and imposes a reasonable dynamics upon the system. However, the relative strengths and weaknesses of the Troisi and SVMMC algorithms are quite complementary. In the next section we will therefore propose our algorithm which combines Troisi-like and SVMMC moves to achieve a robust algorithm for simulating the dynamics of self-assembly in an inhomogeneous environment.

3.4 Dynamical Algorithm

The diffusional moves in the hybrid MC algorithm, presented in section 3.4.1, are a special case of the moves in the Troisi algorithm that ensure predictable behaviour. Relaxation moves are based on the SVMMC algorithm, which successfully proposes collective relaxation moves in aggregates of a range of sizes. These moves are described in section 3.4.2.

Subsequently the way in which realistic dynamics are imposed is discussed.

3.4.1 BULK DIFFUSION MOVES

In our dynamical MC algorithm the diffusion of particles is handled by a cluster move that translates and rotates a set of interacting particles. The size of the move is scaled such that clusters of different sizes diffuse at appropriate relative rates (this will be discussed in depth in section 3.4.3).

An important difference between the cluster moves presented here and similar algorithms we have already discussed is that here the linking stage is completely deterministic. In each of the previously mentioned algorithms, links between particles are made probabilistically. Generally, particles that interact more strongly are more likely to be linked. By contrast, in our diffusion algorithm the system is unambiguously divided into clusters, which are groups of particles connected by a network of non-zero interactions. Two particles i and j in our system are hence linked with the probability

$$\pi_{ij} = \begin{cases} 1 & \text{if } u_{ij} < 0 \\ 0 & \text{otherwise}\,, \end{cases} \tag{3.5}$$

where u_{ij} is the interaction energy between particles i and j. Interacting particles are always linked, which is the limit of the Troisi algorithm with $\beta_{\mathrm{f}} \to \infty$. Two particles belong to the same cluster if there exists an unbroken series of non-zero interactions between them. A schematic of the partitioning of a system into clusters, and an example of a diffusion move, are shown in figure 3.5

Note that this approach assumes that interactions between particles are always negative or zero in any accessible state. This algorithm may not, therefore, be used with soft repulsive potentials where a finite, positive interaction energy is possible. However, it is suitable for the patchy cube model, and others featuring hard bodies, where overlaps result in a positive infinite potential as these states are not accessible.

When a move is conducted, a single particle is picked from a uniform distribution, and the entire cluster to which this particle belongs is moved. In order to ensure detailed balance (equation 2.9), a move from state μ to ν is accepted according to

$$W_{\mathrm{acc}}^{\mu\to\nu} = \begin{cases} 1 & \text{if } U_\nu - U_\mu = 0 \\ 0 & \text{otherwise}\,, \end{cases} \tag{3.6}$$

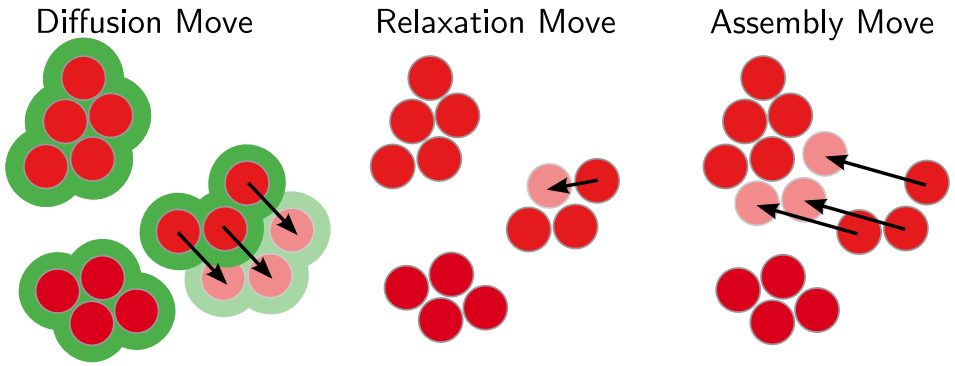

Figure 3.5: Three schematics of types of move possible in the dynamical algorithm. Left is an example of a bulk diffusional move. Here, the green boundaries around the particles show the range of the attractive potential. These boundaries show how the system is unambiguously divided into clusters. A diffusion move may pick one of these clusters and move it, so long as its boundary does not overlap with the boundary of another cluster. The remaining two figures are examples of moves that are handled by the relaxation part of the dynamical algorithm, which will be introduced in section 3.4.2. Centre is an example of an internal relation, where a rearrangement occurs within one cluster to reduce its energy. Right is an example of an assembly move where one cluster is brought to interact with another to form a larger aggregate.

where U_μ is the total potential energy of state μ. This means that moves are accepted only when no new interactions are created as a result of the move (including moves which result in hard core overlaps of particles). In other words, the algorithm begins by selecting an isolated cluster, and then moves it with the condition that it must still be isolated after the move. As there is no change in the total energy of the system the statistical weight of the μ and ν states are the same. If the geometry of moves is symmetric (the probability of proposing the move $\mu \to \nu$ is the same as $\nu \to \mu$), then moves may always be accepted, providing no overlap or change in potential occurs.

The rejection-free nature of the diffusion moves ensures that dynamics can be controlled. By not allowing any change in interaction energy, this type of move is not able to produce assembly or relaxation of clusters. The following section will present the moves used in the hybrid algorithm to allow assembly and relaxation of clusters.

3.4.2 Relaxation Moves

Our choice of assembly move is based on the SVMMC algorithm described in section 2.2.5. This decision was made because of this method's excellent ability to propose appropriate assembly and relaxation moves. The symmetrised variant was chosen over the original VMMC algorithm as we found it had a better acceptance rate in general, and

struggled far less in moving and relaxing large aggregates.

In order to ensure that the relaxation moves do not interfere significantly with the diffusion of particles, the extra condition is added that an assembly move must result in a change in the energy of the system. This is opposite to the condition for the diffusion moves. The extra condition was added so that the SVMMC component of the algorithm can only contribute moves which result in some sort of assembly, disassembly or relaxation. Moves of this nature tend to be small, and so do not contribute significantly to diffusion.

Another modification is that the prefactors $\Theta(n_c)$ and $D(R)$ are both omitted as they cannot achieve realistic dynamics, and would otherwise only cause the unnecessary rejection of moves. As the relaxation moves are restricted to non-isolated clusters, the removal of these prefactors will not allow whole clusters to be moved in an unrealistic manner at this stage. No attempt is made to compensate for the fact that particles in large clusters will be chosen more often as a seed for a relaxation move. This feature is not undesirable, as large clusters will in general have more internal degrees of freedom and so will require more moves to relax.

It is also important to consider the hand-over between the two types of step. For example, consider two isolated building blocks coming together to form a larger cluster. At some point the building blocks will interact, at which time relaxation moves must be used to bring the particles closer together. Therefore, it must be ensured that the move sizes for translations are not so drastically different between the two algorithms that the building blocks suddenly begin to move much slower or faster than during the approach. For this reason, translations for the SVMMC-like moves are picked from the same distribution as for a monomer in the cluster moves.

3.4.3 DIFFUSION

Nano and microscale particles in solvent exhibit Brownian motion, where particles move in a random walk with the mean squared displacement increasing linearly with time (or, by analogy with a random walk, the number of steps). The rate of increase of the mean squared displacement is proportional to the diffusion constant for that particle. This section will consider the nature of the diffusion constant and construct a suitable representation for the patchy cube model.

The mean squared displacement of a particle undergoing Brownian motion in three

dimensions increases linearly in time according to

$$\left\langle |\Delta \mathbf{r}|^2 \right\rangle = 6 D_\mathrm{t} t \,, \tag{3.7}$$

where $\mathbf{r}$ is the position vector of the particle, D_t is the translational diffusion constant and t is time. The motion of a particle may be discretised into N steps of size L. In this case

$$\left\langle |\Delta \mathbf{r}|^2 \right\rangle = 3 N L^2 \,, \tag{3.8}$$

where $N = t/\Delta t$ and

$$L^2 = 2 D_\mathrm{t} \Delta t \,. \tag{3.9}$$

Equation 3.8 remains true whether each step is equal in size, or of a random size picked from the same distribution.

Particles diffuse in both translational and rotational space. The equivalent expression to the mean squared displacement (equation 3.7) in rotational space is

$$\left\langle |\Delta \mathbf{w}|^2 \right\rangle = 6 D_\mathrm{r} t \,, \tag{3.10}$$

where $\Delta \mathbf{w}$ is the rotational displacement of a particle in the angle-axis notation and D_r is the rotational diffusion constant. It is important to note that successive rotations are not commutative and the above expression is only true for infinitesimal rotations. Here, it is assumed that the hard body rotations are small enough to make equation 3.10 a reasonable approximation.

For a spherical particle, the translational and rotational diffusion constants vary with the particles radius R according to the Stokes–Einstein relations

$$D_\mathrm{t} = \frac{k_\mathrm{B} T}{6 \pi \eta R} \tag{3.11a}$$

and

$$D_\mathrm{r} = \frac{k_\mathrm{B} T}{8 \pi \eta R^3} \,, \tag{3.11b}$$

where k_B is Boltzmann's constant, T is the temperature and η is the viscosity of the medium.

Our main concern in simulating self-assembly is that, between collisions with other aggregates, clusters of different numbers of particles should diffuse at physically reason-

able relative rates. It is, therefore, sufficient to ensure that the diffusion of any cluster relative to that of a monomer ($R = 1$) is correct, and that the quotient of translational and rotational diffusion constants satisfies

$$\frac{D_\mathrm{t}}{D_\mathrm{r}} = \frac{4}{3}R^2\,, \tag{3.12}$$

in accordance with equation 3.11.

For simplicity, we approximate clusters of n particles in our simulations as spheres of radius $dn^{1/3}$, where d is the edge length of a cube. The quotient of diffusion constants can then also be expressed as

$$\frac{D_\mathrm{t}}{D_\mathrm{r}} = \frac{4}{3}\left(dn^{1/3}\right)^2\,. \tag{3.13}$$

The justification of this scaling will become apparent in chapter 4, where our octamer target has twice the radius of a monomer. As long as clusters are relatively compact we consider the radius approximation a sensible compromise between realistic dynamics and computational efficiency. It would be possible, if necessary, to improve the accuracy of the algorithm by considering drag produced by the shape of the cluster perpendicular to the direction of travel in a similar way to the dynamical factors in the original VMMC algorithm.[114]

Since the diffusion constant of a random walk is proportional to the square of the mean step size (equation 3.9), the mean translation and rotation steps should then vary with cluster size according to

$$\delta_\mathrm{t}(n) \propto n^{-1/6} \tag{3.14a}$$

and

$$\delta_\mathrm{r}(n) \propto \sqrt{\frac{3}{4}}n^{-1/2} \tag{3.14b}$$

respectively.

In practice, diffusive steps are implemented by selecting a particle at random, determining the other particles that belong to the same cluster, and then performing the trial move of the cluster. This means that the probability of choosing a particular cluster is proportional to the number of particles it contains, since any of its members could have been the particle initially selected. In diffusive motion, the mean square displacement is proportional to the number of steps (equation 3.7). Therefore, the effect of choosing a cluster with probability proportional to n may be cancelled by multiplying the step size

by $n^{-1/2}$. Hence, to produce the correct relative mean square displacement of aggregates our final choice of step sizes must be scaled according to

$$\delta_{\mathrm{t}}(n) \propto n^{-2/3} \tag{3.15a}$$

and

$$\delta_{\mathrm{r}}(n) \propto \sqrt{\frac{3}{4}} n^{-1}. \tag{3.15b}$$

3.4.4 Coupling Diffusion and Assembly

The scaling of move sizes presented at the end of the previous section (equation 3.15) will ensure that the translational and rotational diffusion constants scale correctly as a function of cluster size. However, for the relative rates of diffusion to be correct it is necessary that the diffusion constants for a monomer satisfy equation 3.13.

The previous section made no assumptions about the nature of the distribution from which moves are picked, but it is still desirable that the distribution of individual moves is dynamically realistic. A popular method of generating translational moves in MC simulations is to pick a random deviate uniform in the range $[-\Delta, \Delta]$ for each spatial dimension. Using this scheme, moves are uniformly distributed within the volume of a cube of edge length 2Δ centred on the particle being moved. This scheme is not dynamically realistic as it is not isotropic; it allows particles to move further in some directions than others.

In order to achieve more realistic motion, we pick a displacement in each spatial dimension from a Gaussian distribution,

$$p(x) = \frac{1}{\Delta_{\mathrm{t}}\sqrt{2\pi}} \exp\left(-\frac{x^2}{2\Delta_{\mathrm{t}}^2}\right), \tag{3.16}$$

where $p(x)$ is the probability of picking a displacement (in any of the Cartesian axes) of size x and Δ_{t} is the standard deviation of the Gaussian, which controls the move size. Picking displacements in this manner means that translations are made with equal probability in all directions, with the magnitude of displacement taken from the Maxwell–Boltzmann distribution.

In a rotational move the angular displacement is given by

$$\Delta\mathbf{w} = \theta\hat{\mathbf{u}}\,, \tag{3.17}$$

where θ is the angle of rotation and $\hat{\mathbf{u}}$ is the vector about which the particle is rotated. As $\hat{\mathbf{u}}$

is a unit vector, the magnitude of the angular displacement is simply given by $|\Delta\mathbf{w}| = \theta$. Similarly to a translation, the angle of displacement is picked from a Maxwell–Boltzmann distribution with standard deviation Δ_r. A rotational move is conducted by taking an angle picked from this distribution and an axis of rotation picked uniformly on the surface of a sphere. Picking the angle in this way ensures that the likelihood of proposing the forward and reverse move is identical.

If $\Delta_\mathrm{t}/\Delta_\mathrm{r} = d$ then the required relation between diffusion constants (equation 3.13) can be achieved by picking move sizes in translational and rotational space according to the following equations

$$\delta_\mathrm{t}(n, \Delta_\mathrm{t}) = r_\mathrm{M}(\Delta_\mathrm{t}) n^{-2/3} \tag{3.18a}$$

and

$$\delta_\mathrm{r}(n, \Delta_\mathrm{r}) = \theta_\mathrm{M}(\Delta_\mathrm{r}) \sqrt{\frac{3}{4}} n^{-1}\,, \tag{3.18b}$$

where r_M and θ_M are a distance and an angle picked from Maxwell–Boltzmann distributions with standard deviations Δ_t and Δ_r respectively.

In our algorithm the geometry of SVMMC moves is picked in the same way as the diffusion moves. We found that the mean cluster size generated by the SVMMC algorithm depends on the size of move. Large moves tend to encourage the recruitment of more particles to the cluster and generally end up incorporating all interacting particles. Small moves, on the other hand, often result in only single particles being moved. In order to ensure a good balance of single and cluster moves, and to avoid the assembly moves proposing many whole-cluster moves which will always be rejected, we adjusted the standard deviations Δ_t and Δ_r (subject to $\Delta_\mathrm{t}/\Delta_\mathrm{r} = d$) until a reasonable balance of cluster sizes was achieved. The same Δ values were then used for diffusion moves, so that there is a smooth handover between the moves at a single-particle level in the two move types.

Every step in the dynamical algorithm begins by deciding whether to conduct a diffusion or assembly move. These move types are chosen with equal probability. Subsequently it is determined whether the move will be translational or rotational; again these are selected with equal probability. For diffusion moves, picking translations and rotations with equal probability is nessecary to achieve the correct quotient of diffusion constants (equation 3.12. For the assembly moves this is not necessary as they should not contribute significantly to diffusion.

The success of the dynamical algorithm in producing trajectories with Stokes–Einstein diffusion can be demonstrated by determining the translational and rotational diffusion constants as a function of cluster size, as plotted in figure 3.6. All data points were obtained by simulating a single cluster, which was very strongly bound to ensure that there was little risk of fission over the course of the simulation. Translational diffusion constants were determined from the mean squared displacement of particles, using equation 3.7, with t being measured in MC cycles. Rotational diffusion constants are estimated using a similar method to translational constants, where angular displacements are represented as a vector and summed to determine the total angular distance travelled.[105] The observed diffusion constants in figure 3.6 lie very close to the required trend. Small deviations from ideal behaviour are due to the fact that the VMMC moves do contribute a small amount to the bulk diffusion of the system. However, this effect is not large enough to seriously disrupt the intended diffusion rates.

The diffusion constant of a monomer may be used as a basis for interpreting the number of MC cycles completed as a measurement of time. If the step size is picked from the same distribution for a series of simulations, the diffusion constants will be the same, and hence the time scales in steps are identical. To simply use the number of MC cycles as a measure of time would result in temperature independent diffusion constants. However, the diffusion constants must vary linearly with temperature, according to equation 3.11. In order to compensate for this, the step size could be changed in order to appropriately increase or decrease the diffusion constant. However, this approach is poor because the step sizes should be set to produce a good acceptance of internal relaxations, which is largely independent of temperature.

In order to keep an appropriate step size, and have consistent time scales over a range of temperatures, we introduce a reduced time t^* that depends on both the simulation temperature and the number of MC cycles. Consider, for instance, doubling the temperature and hence diffusion constant. Equation 3.7 shows that if the mean squared displacement remains constant (due to a fixed time step) then the time associated with each move must be halved. Therefore, we propose the mapping

$$t^* = n_{\text{cycles}}/T^* \,, \tag{3.19}$$

where n_{cycles} is the number of MC cycles completed and T^* is the reduced simulation temperature, which will be defined for different models in their appropriate chapters.

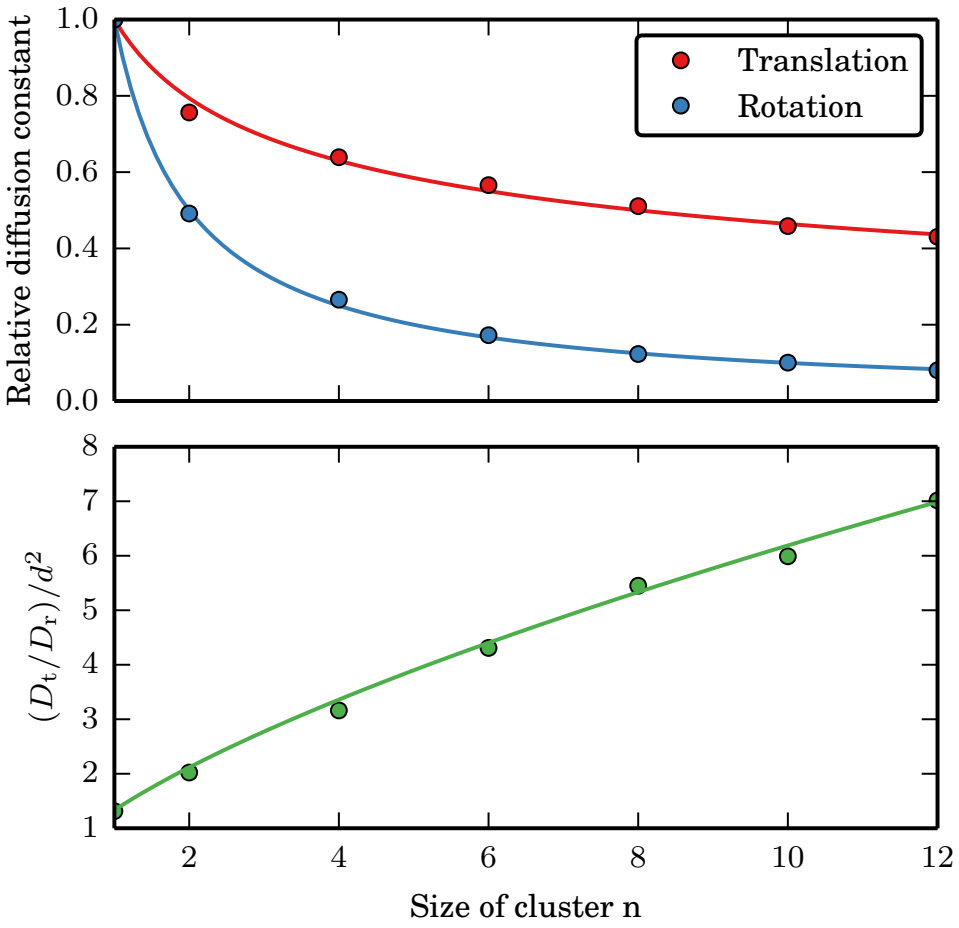

Figure 3.6: The top plot shows diffusion constants in the low-density limit for clusters of up to 12 particles, relative to the value for a monomer. Circles are diffusion constants from simulations with the dynamical MC algorithm while the solid lines show the desired trend from the Stokes–Einstein equations for spherical particles equation 3.11. The bottom plot shows the quotient of the translational and rotational diffusion constants. Here, the solid line shows the correct trend from equation 3.13

APPENDIX 3.A ORIENTED BOUNDING BOXES

The presence of an overlap between the hard cubic particles is tested using the seperating axis theorem for oriented bounding boxes (SATOBB).[121] If two cubes do not overlap then there must be a plane which separates the two. The separating axis is normal to the plane that separates the cubes. The projections of the cubes onto this axis must not overlap. In other words, the sum of the projections of the half dimensions of the boxes onto the separating axis must be less than the projection of the inter-particle vector onto the same axis. This concept is easier to visualise in two dimensions as demonstrated in figure 3.7.

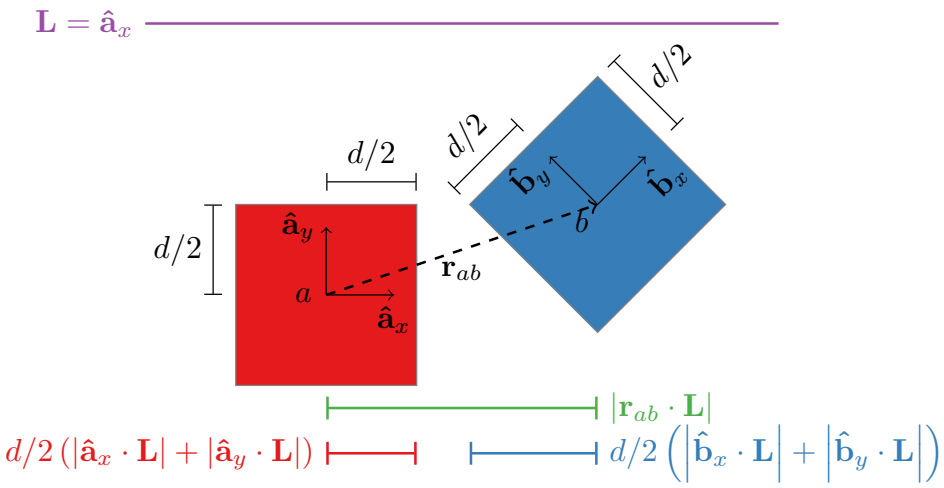

Figure 3.7: An example of a separating axis $\mathbf{L}$ for two squares a and b, with edge length d. Here $\mathbf{L} = \hat{\mathbf{a}}_x$ and we can see that the sum of the projections of the squares half dimensions onto $\mathbf{L}$ (red and blue lines) is less than the projection of the vector between the squares onto $\mathbf{L}$ (green line). Thus, the squares do not overlap. Notice that in the case $\mathbf{L} = \hat{\mathbf{b}}_y$ this is not the case, as the projections of the squares into that dimension do overlap.

For three-dimensional boxes there are fifteen axes to test. First there are the six axes normal to the faces of the two cubes, which are equivalent to the six vectors defining the edges. Second there are nine potential separating planes spanned by the 3×3 combinations edge vectors. The corresponding separating axes of these planes are given by the nine combinations of cross products of the edge vectors. The separating axes $\mathbf{l}$ to be checked for two cubes a and b are therefore $\hat{\mathbf{a}}_x$, $\hat{\mathbf{a}}_y$, $\hat{\mathbf{a}}_z$, $\hat{\mathbf{b}}_x$, $\hat{\mathbf{b}}_y$, $\hat{\mathbf{b}}_z$, $\hat{\mathbf{a}}_x \times \hat{\mathbf{b}}_x$, $\hat{\mathbf{a}}_x \times \hat{\mathbf{b}}_y$, $\hat{\mathbf{a}}_x \times \hat{\mathbf{b}}_z$, $\hat{\mathbf{a}}_y \times \hat{\mathbf{b}}_x$, $\hat{\mathbf{a}}_y \times \hat{\mathbf{b}}_y$, $\hat{\mathbf{a}}_y \times \hat{\mathbf{b}}_z$, $\hat{\mathbf{a}}_z \times \hat{\mathbf{b}}_x$, $\hat{\mathbf{a}}_z \times \hat{\mathbf{b}}_y$ and $\hat{\mathbf{a}}_z \times \hat{\mathbf{b}}_z$, where $\hat{\mathbf{a}}_x$ is the unit vector defining the orientation of the local axis x of particles a.

To save the computational expense of calculating the cross products the following relation may be used

$$\mathbf{r}_{ab} \cdot \left(\hat{\mathbf{a}}_x \times \hat{\mathbf{b}}_x\right) = (\mathbf{r}_{ab} \cdot \hat{\mathbf{a}}_z)\left(\hat{\mathbf{a}}_y \cdot \hat{\mathbf{b}}_x\right) - (\mathbf{r}_{ab} \cdot \hat{\mathbf{a}}_y)\left(\hat{\mathbf{a}}_z \cdot \hat{\mathbf{b}}_x\right) , \tag{3.20}$$

along with the eight related equations by permutation of dimensions x, y and z. The fifteen inequalities, for cubes with edge length d, for each separating axis $\mathbf{l}$ are

axis $\mathbf{l}$	inequality
$\hat{\mathbf{a}}_x$	$\lvert\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_x\rvert > \frac{1}{2}d\left(1+\lvert S_{xx}\rvert+\lvert S_{xy}\rvert+\lvert S_{xz}\rvert\right)$
$\hat{\mathbf{a}}_y$	$\lvert\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_y\rvert > \frac{1}{2}d\left(1+\lvert S_{yx}\rvert+\lvert S_{yy}\rvert+\lvert S_{yz}\rvert\right)$
$\hat{\mathbf{a}}_z$	$\lvert\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_z\rvert > \frac{1}{2}d\left(1+\lvert S_{zx}\rvert+\lvert S_{zy}\rvert+\lvert S_{zz}\rvert\right)$
$\hat{\mathbf{b}}_x$	$\left\lvert\mathbf{r}_{ab}\cdot\hat{\mathbf{b}}_x\right\rvert > \frac{1}{2}d\left(1+\lvert S_{xx}\rvert+\lvert S_{yx}\rvert+\lvert S_{zx}\rvert\right)$
$\hat{\mathbf{b}}_y$	$\left\lvert\mathbf{r}_{ab}\cdot\hat{\mathbf{b}}_y\right\rvert > \frac{1}{2}d\left(1+\lvert S_{xy}\rvert+\lvert S_{yy}\rvert+\lvert S_{zy}\rvert\right)$
$\hat{\mathbf{b}}_z$	$\left\lvert\mathbf{r}_{ab}\cdot\hat{\mathbf{b}}_z\right\rvert > \frac{1}{2}d\left(1+\lvert S_{xz}\rvert+\lvert S_{yz}\rvert+\lvert S_{zz}\rvert\right)$
$\hat{\mathbf{a}}_x\times\hat{\mathbf{b}}_x$	$\lvert(\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_z)\,S_{yx}-(\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_y)\,S_{zx}\rvert > \frac{1}{2}d\left(\lvert S_{zx}\rvert+\lvert S_{yx}\rvert+\lvert S_{xz}\rvert+\lvert S_{xy}\rvert\right)$
$\hat{\mathbf{a}}_x\times\hat{\mathbf{b}}_y$	$\lvert(\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_z)\,S_{yy}-(\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_y)\,S_{zy}\rvert > \frac{1}{2}d\left(\lvert S_{zy}\rvert+\lvert S_{yy}\rvert+\lvert S_{xz}\rvert+\lvert S_{xx}\rvert\right)$
$\hat{\mathbf{a}}_x\times\hat{\mathbf{b}}_z$	$\lvert(\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_z)\,S_{yz}-(\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_y)\,S_{zz}\rvert > \frac{1}{2}d\left(\lvert S_{zz}\rvert+\lvert S_{yz}\rvert+\lvert S_{xy}\rvert+\lvert S_{xx}\rvert\right)$
$\hat{\mathbf{a}}_y\times\hat{\mathbf{b}}_x$	$\lvert(\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_x)\,S_{zx}-(\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_z)\,S_{xx}\rvert > \frac{1}{2}d\left(\lvert S_{zx}\rvert+\lvert S_{xx}\rvert+\lvert S_{yz}\rvert+\lvert S_{yy}\rvert\right)$
$\hat{\mathbf{a}}_y\times\hat{\mathbf{b}}_y$	$\lvert(\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_x)\,S_{zy}-(\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_z)\,S_{xy}\rvert > \frac{1}{2}d\left(\lvert S_{zy}\rvert+\lvert S_{xy}\rvert+\lvert S_{yz}\rvert+\lvert S_{yx}\rvert\right)$
$\hat{\mathbf{a}}_y\times\hat{\mathbf{b}}_z$	$\lvert(\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_x)\,S_{zz}-(\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_z)\,S_{xz}\rvert > \frac{1}{2}d\left(\lvert S_{zz}\rvert+\lvert S_{xz}\rvert+\lvert S_{yy}\rvert+\lvert S_{yx}\rvert\right)$
$\hat{\mathbf{a}}_z\times\hat{\mathbf{b}}_x$	$\lvert(\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_y)\,S_{xx}-(\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_x)\,S_{yx}\rvert > \frac{1}{2}d\left(\lvert S_{yx}\rvert+\lvert S_{xx}\rvert+\lvert S_{zz}\rvert+\lvert S_{zy}\rvert\right)$
$\hat{\mathbf{a}}_z\times\hat{\mathbf{b}}_y$	$\lvert(\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_y)\,S_{xy}-(\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_x)\,S_{yy}\rvert > \frac{1}{2}d\left(\lvert S_{yy}\rvert+\lvert S_{xy}\rvert+\lvert S_{zz}\rvert+\lvert S_{zx}\rvert\right)$
$\hat{\mathbf{a}}_z\times\hat{\mathbf{b}}_z$	$\lvert(\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_y)\,S_{xz}-(\mathbf{r}_{ab}\cdot\hat{\mathbf{a}}_x)\,S_{yz}\rvert > \frac{1}{2}d\left(\lvert S_{yz}\rvert+\lvert S_{xz}\rvert+\lvert S_{zy}\rvert+\lvert S_{zx}\rvert\right)$

where $S_{xy}=\hat{\mathbf{a}}_x\cdot\hat{\mathbf{b}}_y$.

As soon as one of the inequalities is found to be true, it has been shown that there is no overlap, and so the remaining inequalities need not be checked to save computer time. Furthermore, early acceptance/rejection may be introduced to avoid the expense of the SATOBB procedure altogether. For cubes with edge length d, the closest approach without overlap possible is $|\mathbf{r}_{ab}|=d$. Therefore if $|\mathbf{r}_{ab}|<d$ there definitely is an overlap. The minimum distance at which two cubes cannot overlap is $\sqrt{3}d$ and so if $|\mathbf{r}_{ab}|>\sqrt{3}d$, there is definitely no overlap.

CHAPTER 4

DESIGN STRATEGIES FOR SELF-ASSEMBLY

4.1 INTRODUCTION

The outcome of self-assembly is strongly influenced by the design of the building blocks. This refers not only to the structure of the final product of assembly, but also to the pathway towards that product. Encoding the desired outcome in the design of self-assembling particles in this way is an example of programmable assembly.[76,82,84] The patchy cube model introduced in the previous chapter can be used to explore how different building block designs affect self-assembly. Using the dynamical MC algorithm makes it possible to analyse the rates, pathways and success of assembly for a range of designs. In this chapter, the patchy model will be used to compare the merits of three strategies to achieve efficient and reliable self-assembly.

A strategy is a plan for the pathway of self-assembly, which aims to ensure that assembly is robust by avoiding potential pitfalls. Even at optimal conditions, a thermodynamic yield of the target may take a very long time to achieve due to kinetic traps on the assembly pathway.[123] A kinetic trap occurs when the system spends a long time in metastable states along the path to the energetic ground state of the target. A successful design should not only stabilise the target, but also influence the pathway to avoid kinetic traps.

One strategy is sequential assembly from identical components, where the target builds one particle at a time from a reservoir of monomers. The advantage of this approach lies in its simplicity. With only a single component, the target may be formed from any combination of the right number of building blocks. This provides an entropic benefit as the target has a large multiplicity of permutations, $N!$ where N is the number of building blocks in the target. Furthermore there are many pathways to the target structure.

An important example of this strategy is the assembly of certain icosahedral virus

capsids that form from a precise number of identical proteins. Capsids typically feature a large number of weak interactions between the building blocks, and this has been demonstrated to help to avoid kinetic traps.[123–125] Strong bonds, while stabilising the target, encourage incorrect structures to form. Correct assembly instead proceeds when bonds are formed transiently, which allows the system to more easily escape traps and remedy incorrect aggregates.[75] Nature has therefore found a robust assembly strategy for large structures from the bare minimum of building blocks. The high symmetry of the capsids is undoubtedly key in allowing for one-component assembly.

Recently successful assembly has been demonstrated using quite the opposite approach, where every building block in the target is unique. This is termed addressable assembly (section 1.3) and each component occupies a specific position and orientation in the structure. The correct arrangement of building blocks is ensured with strong, specific interactions between neighbours in the target structure. This approach loses the entropic advantages of a single-component assembly; there is only one permutation of particles which is correct, compared to the $N!$ permutations for a single-component system. Therefore, the more severe entropy penalty must be compensated for with a strong binding enthalpy between building blocks. As addressable structures grow larger the entropic penalty becomes worse and so binding free energies must further increase. However, this will probably also involve increasing the strength of nonspecific interactions between building blocks that are not bound in the target, which increases the likelihood of kinetic traps on the pathway to assembly.[5]

The pathway to the target may be also be more precisely controlled through particle design. In hierarchical schemes the assembly of the final target is broken down into two or more stages. Smaller subunits of the target are assembled first, before themselves combining.[126] The potential advantage of this approach is that a large assembly problem is broken down into smaller stages which can be controlled more precisely.[127–129] Capsids have already been introduced as an example of sequential assembly from a single component. However, some capsids may assemble in a hierarchical scheme where proteins first form small clusters called capsomeres, which then assemble to form the capsid itself. Simulations have suggested that the HIV-1 virus assembles in a multistep procedure where protein dimers form capsomeres consisting of six proteins.[130,131] This work suggests the hierarchical approach mitigates competition from the many competing structures which could be formed from a large set of monomers.

However, more recent simulations have cast doubt over whether a hierarchical scheme

can ever perform better than the sequential addition of monomers. In the model of Haxton *et al.*[99] a lattice target formed from rectangular tiles has an inherently hierarchical structure. Through control of the binding energies of two distinct binding sites on the tiles, assembly can be encouraged to occur through two separate routes, one via a dimer intermediate, the other via tetramers. The authors found that if particles only interacted through these binding sites, hierarchical assembly was no worse than sequential assembly. The pathway to assembly could be controlled by adjusting the relative strengths of the two interaction sites. For example, assembly through a tetramer intermediate could be promoted by making the interactions between tetramer units stronger. However, when a general interaction between entire particles (rather than interaction sites) was introduced, the hierarchical schemes rapidly became worse than assembly through sequential addition. This would suggest that the success of a hierarchical route relies heavily on being able to exclude more general interactions between particles, which may be difficult in practice.

The large difference between the dimer and tetramer binding energies required to ensure hierarchical assembly encouraged the formation of kinetic traps. One form of trap was caused by the weaker of the two specific sites becoming similar in strength to the general interaction, leading to amorphous aggregates. A second type of trap occurred when the specific interactions between tetramers were made particularly strong. Here, links between monomers were made very quickly, consuming all monomer and often leaving many trimers that are unable to combine to form the target structure.

Given the range of possible strategies, we should ask how best to go about assembling a given target. The simplicity of a one-component solution is attractive, requiring much less investment in producing the components. However, such an approach is limited by the symmetry of the target. On the other hand, the fully addressable solution, while applicable to any target, requires a large amount of design input which may be unnecessary for small targets, or targets with symmetry or which can be broken into smaller fragments.

The patchy cube model is well suited to studying the different self-assembly strategies because the patterning of faces is an explicit representation of the interactions between particles. Features such as symmetry, a torsional potential and exclusivity may be created through the placement of patches. This is in contrast to many previous studies, where the description of interactions is more abstract. Examples include single interaction sites,[132] implicit torsional potentials,[73] and all-or-nothing interactions based on lattice positions.[98,133] Using the hybrid MC algorithm also means that the dynamics of assembly

will be more realistic than some idealised models, such as the assembly protocol used by Hedges *et al.* described in section 2.1.

In this chapter, a range of contrasting strategies for the self-assembly of a small target will be tested and compared using intuitively designed building blocks. The designs used will cover the gamut of strategies employed in nanoparticle self-assembly. By determining the success of each strategy for the same target, it is possible to compare the strengths and weaknesses of each strategy and provide some insight into when each may be an appropriate choice.

4.2 Fragment Detection

In order to quantify the progress of assembly in a simulation, it is necessary to detect successfully completed target structures, as well as plausible intermediates and fragments. All such clusters will deviate from idealised geometries due to thermal fluctuations, making it necessary to introduce some tolerance in matching instantaneous configurations to a library of structures being tracked. The method for identifying fragments must be computationally efficient, since it will be applied repeatedly during the simulations.

One class of metrics that satisfy many of the desirable properties of fragment detection algorithms is based on matrices of a pairwise property such as distance or energy. The eigenvalues of such matrices are unaffected by bulk translation, rotation and particle permutations, thereby providing a "fingerprint" by which structures can be recognised.[134] Nevertheless, a tolerance for comparison of the eigenvalues must be chosen, and the most appropriate tolerance may depend on the size of the structure and the number of variables in its fingerprint. Moreover, an eigenvalue is determined by all elements of a matrix collectively, so there is no direct structural interpretation of the deviation of an eigenvalue from the ideal case.

We therefore propose a scheme for fragment recognition based on pairwise links between particles. Here, the definition of a link includes the geometry of the arrangement and the particular types of particle involved, not just interaction energy or distance. The tolerances have clear physical meaning and, as the accuracy of links made is invariant with respect to the size of a fragment, these tolerances may be fixed for all fragments. This versatile approach means it is possible to detect the target, and fragments of the target, in a consistent manner with a small number of parameters. Examples of the kinds of fragments the algorithm can handle are shown in figure 4.1.

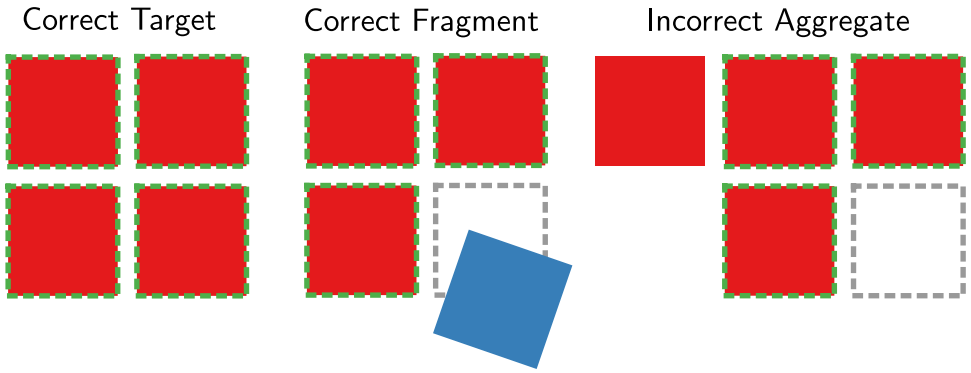

Figure 4.1: A selection of cases that the fragment detection algorithm may encounter and successfully handle. Left is a correct target with all particles in their correct location and orientation (indicated by the green borders). In the centre is a correct fragment of the target with three particles in their correct location. There is also a fourth particle which is interacting with the fragment, but is not strongly bound to it. The algorithm should detect the correct fragment and be able to discount the casual interaction with the fourth particle as we would not consider this particle to be part of the aggregate. Right is an incorrect aggregate where four particles have bound strongly into a geometry not compatible with the target structure, which can be seen as this structure does not fit within the borders.

The first step is to identify aggregates that are potential candidates for recognition as a structural fragment. For this purpose we use a similar definition of aggregates to that used for isolated clusters in diffusion MC moves (section 3.4.1), *i.e.*, that cubes a and b belong to the same cluster if they are interacting ($V_{ab}^{\text{cube}} < 0$). However, the additional requirement that a and b are closer than $r_{ab} = 1.207d$ is imposed. When the orientation of two cubes differs by $45°$, this is the shortest distance between their centres when the edge of one is in contact with the face of the other. This requirement on proximity ensures particles that are only very loosely associated with a fragment—and therefore should not be considered part of an aggregate—are ignored. An aggregate that we would consider a correct fragment is therefore not disregarded because of a casual interaction with nearby particles (see figure 4.1 centre), as would be the case if all interacting particles were admitted to the aggregate. A quick test can now be performed on the aggregate to see whether its size and (in cases where more than one type of building block is in use) its composition match those of recognised fragments. Any aggregate which fails to meet either the size or composition requirements may be rejected at this stage without the need for any further analysis.

After identifying aggregates the links between members of the aggregate are examined in more detail. All links between particles are enumerated. A link between a pair

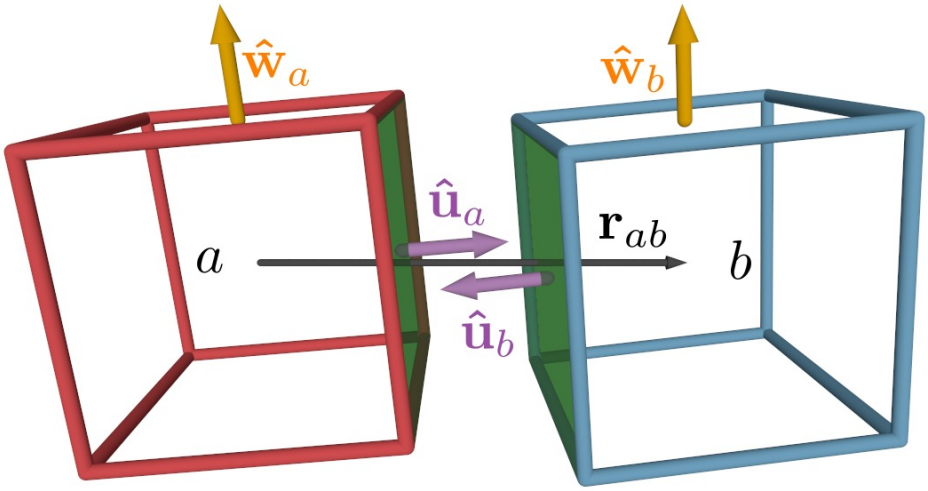

Figure 4.2: Testing of links between particles in a fragment involves the unit normals $\hat{\mathbf{u}}_a$ and $\hat{\mathbf{u}}_b$ to the faces that have come together, and the auxiliary vectors $\hat{\mathbf{w}}_a$ and $\hat{\mathbf{w}}_b$ associated with the linked faces.

of particles (a and b) is characterised by the species of the two particles, and through which face on each particle the link is made. An interaction between two particles is only counted as a link when it meets the following criteria.

1. The centres of the building blocks must be closer than $r_{ab} = 1.140d$, which is more strict than the requirement for simply being admitted to an aggregate.

2. The faces through which the link is made must be well aligned.

3. The building blocks must be correctly orientated within a torsional tolerance.

For two faces to be considered well aligned as in criterion 2 above, the following conditions must be met, $\hat{\mathbf{r}}_{ab} \cdot \hat{\mathbf{u}}_a > 0.8$ and $\hat{\mathbf{r}}_{ba} \cdot \hat{\mathbf{u}}_b > 0.8$, where $\hat{\mathbf{u}}_a$ is the unit vector normal the face on particle a through which the link is made (figure 4.2). The orientation criterion in item 3 is $\hat{\mathbf{w}}_a \cdot \hat{\mathbf{w}}_b > 0.95$, where $\hat{\mathbf{w}}_a$ is a unit vector normal to one of the faces on particles a adjacent to the linking face. $\hat{\mathbf{w}}_b$ is a companion vector on particle b, and so the criterion is met when $\hat{\mathbf{w}}_a$ and $\hat{\mathbf{w}}_b$ lie parallel (figure 4.2). When a link is defined, these two vectors are therefore chosen to be parallel in the correct fragment geometry.

The criteria described above allow for deviation from the ideal structure in order to allow for thermal fluctuations without discounting correct structures. Aggregates that we would not consider correct fragments typically fail this requirement by a significant amount.

For any fragment or complete target which should be tracked in a simulation, the list of links that define its geometry is created. Recognition of clusters of interest is then simply a matter of matching a sorted list of links of fragments present in the simulation against the lists of clusters being tracked.

4.3 RESULTS

A simple octamer target will be used to contrast the self-assembly strategies. The target consists of eight particles joined into a $2 \times 2 \times 2$ cube. The high symmetry of the target allows us to compare the full spectrum of strategies, from assembly from a single component, to the fully addressable limit where each of the eight particles has a unique identity and position within the target.

All simulations in this work consist of 64 particles, enough to make eight full copies of the target, at a number density of $0.05d^{-3}$. Not all studies of nanoparticle self-assembly consider multiple replicas of the target in a single simulation. However, it is an important consideration as there are kinetic traps involving large aggregates or many partially assembly clusters which can only occur when multiple copies of each particle are present. Each simulation begins from a high-temperature fluid of monomer, which is immediately quenched to the assembly temperature at the beginning of the simulation. To obtain statistics of the progress of assembly as a function of time, an ensemble of 50 replicas, each starting from a different disordered state, were averaged for any given set of conditions.

In order to make comparisons between sets of building blocks with different numbers of patches the temperature scale is related to the energy of an optimised target E_{target}. The reduced temperature is therefore defined by

$$T^* = k_{\text{B}}T / \left|E_{\text{target}}\right| \ . \tag{4.1}$$

The reduced temperature in turn defines the reduced time according to equation 3.19.

4.3.1 SINGLE-COMPONENT ASSEMBLY

A summary of our designs is presented in figure 4.3. The three models (A–C) are all examples of single-component strategies. In each of the designs A–C, the patches are placed such that the particles have C_{3v} symmetry. The diagonal line of symmetry on each face then ensures that the patches may always match up with the neighbouring particle,

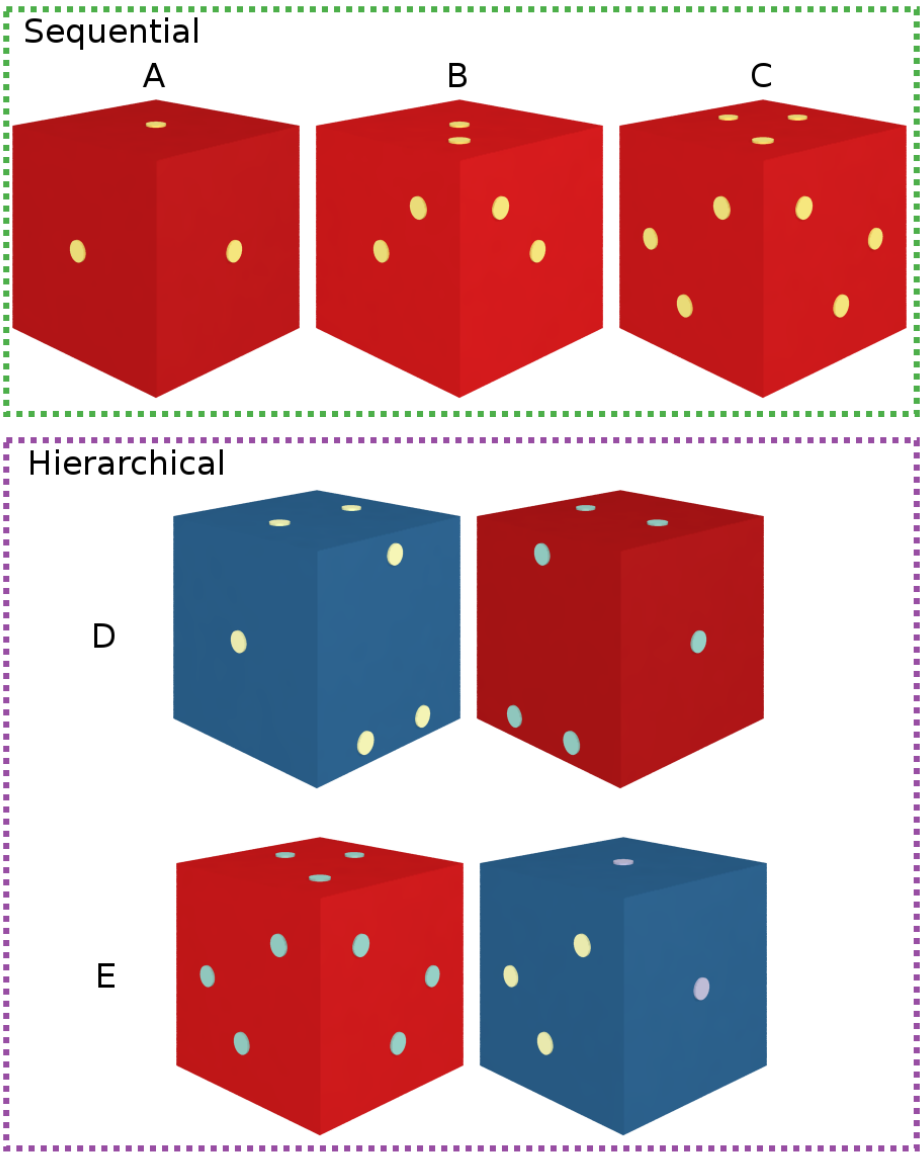

Figure 4.3: Designs of patchy particles used in this work. The sequential A, B and C models self-assemble from systems of only one component and were designed to test the minimal requirements for self-assembly. Both hierarchical models, D and E, consist of two particle types and use different numbers of patches per face, in conjunction with interaction alphabets, to assemble in multiple steps.

whichever position a particle takes in the target. We intend for these designs to assemble the target by sequential addition of monomer.

Moving across the top row of figure 4.3 from model A to C there is an increase in the number of patches per face, and hence the complexity of the design. In all cases

patches interact with equal strength $\varepsilon_{ij} = \varepsilon$. In model A each patterned face only has a single patch, positioned in the centre of the face. This is our minimal design capable of assembling the target octamer.

While the patch pattern of model A does ensure that the target cluster does optimise the energy of the system, there exist many large, disordered networks of particles which satisfy most bonds. These states compete strongly with the intended target, and significantly hinder assembly. Model A therefore proves to be a poor design, producing few correct clusters (figure 4.4). Furthermore, the temperature window in which clusters may be observed is very narrow, approximately $0.06 \lesssim T^* \lesssim 0.07$. Below this range assembly of very large disordered aggregates is rapid and irreversible, and above it the system exists mostly as a fluid of monomers.

Furthermore, even in the region where correct assembly may occur it is very common to see partially assembled clusters where one or more particles are present in an incorrect orientation (figure 4.4 right). Aggregates of this type cannot form the target correctly without first breaking up. The incorrectly exposed patches may bind further, leading to the large aggregates seen at lower temperatures. Large, amorphous aggregates of this type have been previously observed at sub-optimal temperatures in a model of patchy spheres.[70]

Model B features one extra patch per face compared to model A, while retaining C_{3v} symmetry. The extra patch creates an orientational preference to the binding of two particles, which is completely absent with model A's single patch, where all orientations are of equal energy. The second patch results in two minimum energy configurations where the two patches overlap perfectly. One of these orientations is correct, where two particles bind such that their faces overlap perfectly and leave all faces in an orientation compatible with the target. Because the two patches are not located symmetrically along the diagonal of the face, in the other configuration the faces are staggered. When particles bind incorrectly in this way they do not leave a flat binding interface. The intention is to discourage further aggregation of building blocks on incorrectly bound dimers.

The additional patch creates a dependence of the binding energy of two particles on the dihedral angle between them and so can be interpreted as an explicit representation of a torsional potential. In less detailed models of patchy particles implicit torsional potentials have been explored by imposing an extra energy term depending on the dihedral angle between two particles.[72,73] In these studies a torsional potential was found to an important factor to ensure correct assembly. It is therefore consistent that model B assem-

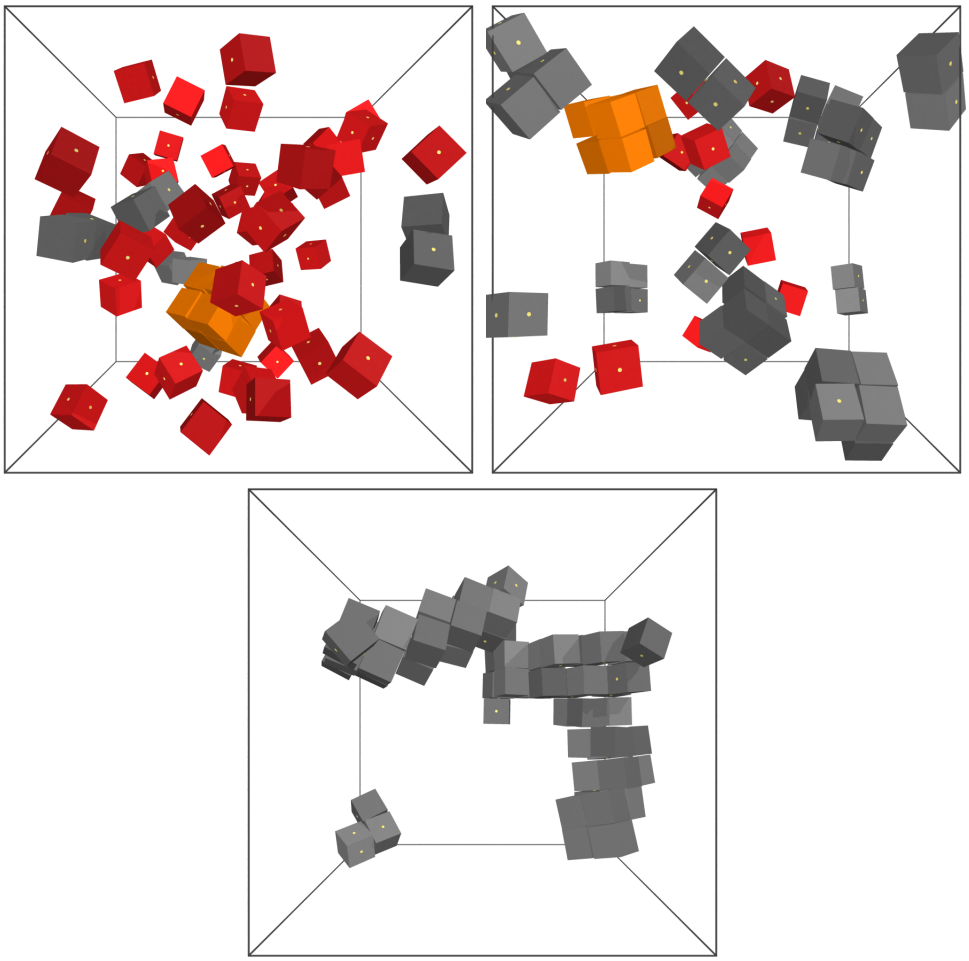

Figure 4.4: Snapshots from simulations of model A at $T^* = 0.07$, 0.06 and 0.04 (left, right and bottom respectively). In the left snapshot a single correctly assembled octamer (orange) can be seen, but most particles existing as monomers (red). At a lower temperature (right), isolated aggregates (grey) with incorrect orientation of building blocks form, and at very low temperatures (bottom) the particles form large, extended aggregates.

bles much more reliably than model A. The additional information, one extra patch per face compared to model A, has produced what is probably the minimal viable design for self-assembly of the octamer target.

In order to quantify the efficiency of assembly for each model the time taken to achieve 50% yield of the octamer target, $t^*_{1/2}$, is determined. This time is plotted as function of assembly temperature in figure 4.5 where model B is represented by red circles. This plot shows the optimal temperature for rapid assembly is around $T^* = 0.055$, although there is a broad range of temperatures either side of this temperature where assembly is compara-

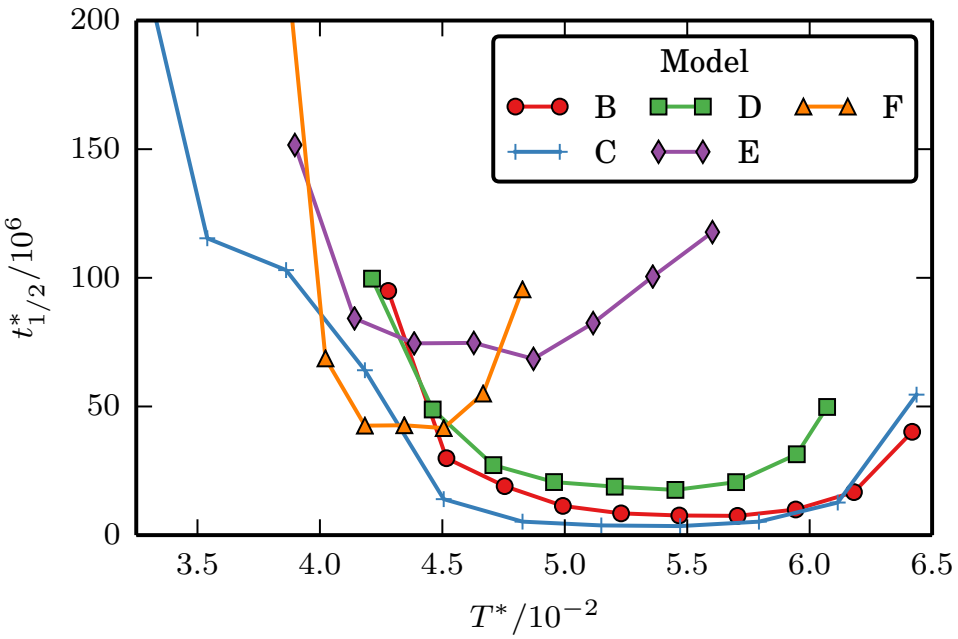

Figure 4.5: The time $t^*_{1/2}$ taken to achieve 50% yield of the octamer target as a function of reduced temperature for models B–E.

bly swift. Assembly begins to slow and eventually fail at both high and low temperatures. The failure at high temperature is due to the target becoming thermodynamically unstable with respect to a system of monomers or smaller clusters. At low temperatures large amorphous aggregates begin to hinder assembly as bonds break less frequently, meaning that incorrect aggregates have a longer lifetime before being corrected.

The third patch introduced on the faces of model C further strengthens the orientational bias introduced in model B. In model C, three patches arranged in an isosceles triangle ensure that there is only one minimum configuration, where all three patches overlap. An equilateral triangle was intentionally avoided as such an arrangement would result in three configurations with minimum energy, only one of which is correct. Although a degenerate ground state has been eliminated, model C does introduce competing bindings where two out of three patches align. These configurations are higher in energy, and hence less stable than the intended orientation. Furthermore, in the incorrect orientation the particles are staggered as the faces no not overlap completely. This arrangement of particles should limit further assembly as it prevents the close-packed arrangements of misaligned particles seen in model A (figure 4.4). For both of these reasons we do not expect the incorrect bindings of model C to severely disrupt assembly.

$t^*_{1/2}$ as a function of temperature for model C is shown by the blue crosses in figure 4.5.

The plot shows that model C assembles with greater speed and over a wider range of temperatures than model B. The most striking difference between the two models is towards lower temperatures where, while slow to assemble, model C proves more resilient than B. This is an indication that the extra patch introduced in model C has to some extent reduced the occurrences of kinetic traps. The introduction of a third patch can therefore be seen as an example of negative design[135] where instead of seeking to promote the target, a design element has been added to make unwanted structures less stable. Avoiding undesired binding between building blocks through negative design will be a major focus of chapter 6

It should be noted that the optimal temperature for assembly of the successful single-component models is quite high, with $k_{\mathrm{B}}T$ around 70% of the total binding energy between two particles. At such temperatures we should expect the bonds formed between individual particles to be quite fragile and easily reversible. This suggests that being able to reverse potentially incorrect bindings is important to the assembly, a feature already recognised as important in capsid assembly.[75,123]

Evidence that assembly mostly follows pathways featuring the sequential addition of monomer can be obtained by examining the histories of successfully completed targets. Figure 4.6 shows two examples of cluster histories for model C. In these diagrams the horizontal axis represents the reduced time since the beginning of the assembly simulation. The horizontal lines represent clusters, with thickness proportional to the number of particles in the cluster. Red lines indicate isolated monomers, while orange lines correspond to the octamer target. Dark lines represent all intermediate structures. Vertical dashed lines indicate the joining or fission of clusters. A cluster is only included in the diagram if one of its particles appears in the final target. Hence, tracing the diagram from right to left reveals the history of how particles came to be incorporated in the product. Other particles that interact only transiently with those that are finally incorporated appear as a temporary thickening of the corresponding line.

The top history in figure 4.6 shows a typical, sequential assembly pathway. Here, almost all assembly steps towards the target feature the combination of a monomer and another cluster. The two exceptions are the formation of a dimer by the combination of two monomer, and what appears as a combination of tetramer and two monomers. It should be noted that the latter of these two may in fact be two sequential steps which occur faster than the resolution at which fragments were identified. Assembly steps involving two clusters (rather than a cluster and a monomer) are also seen, as shown in the

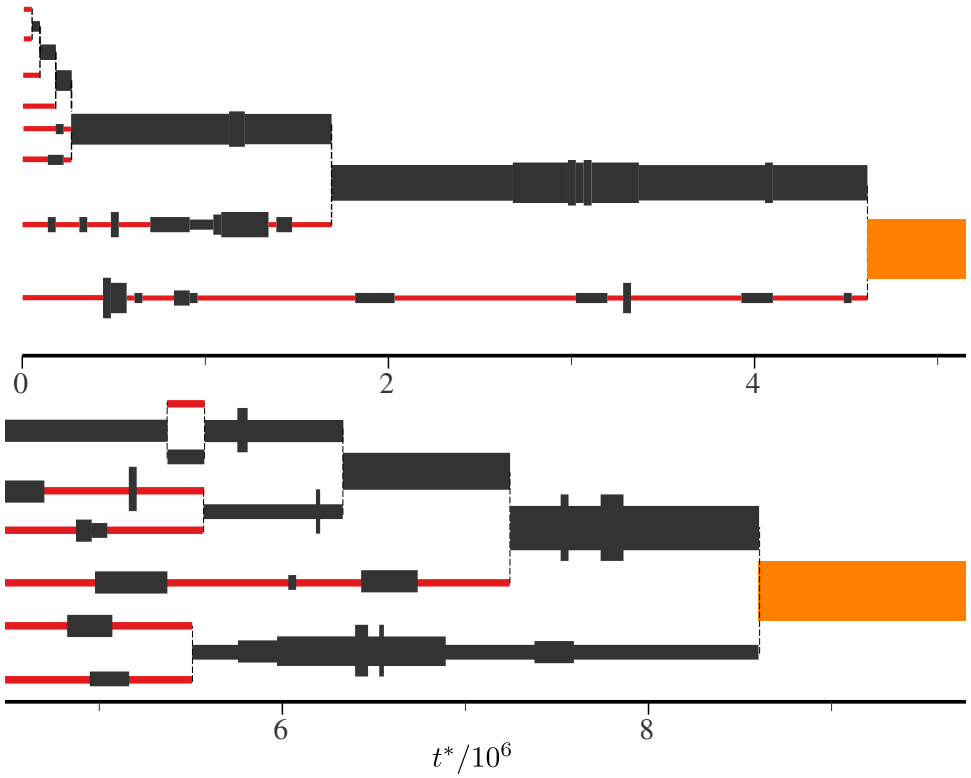

Figure 4.6: Histories of two target clusters of the single component model C obtained from the trajectory of a single simulation at $T^* = 0.05$. Red lines correspond to monomer particles, and orange lines correspond to the octamer target. Dark lines are unrecognised clusters and the width of these lines is proportional to the cluster size. The upper of the two histories shows a sequential assembly pathway. The lower history shows another route to the target where a six-membered cluster binds with a dimer.

lower history in figure 4.6. Here, the target is finally formed through the combination of a six-membered cluster and a dimer. Earlier in the history a five-membered cluster is formed by the fusion of a trimer and a dimer. Pathways featuring assembly steps involving two clusters become more prominent at low temperatures and at later times when the concentration of monomers is low.

Analysis of the trajectories of simulations therefore confirm that assembly does occur predominantly through the sequential addition of monomer. However, the combination of larger aggregates does play a role, particularly in conditions where monomer is less available. These results also confirm that the dynamical algorithm is able to simulate collective assembly moves involving many particles.

4.3.2 HIERARCHICAL ASSEMBLY

In addition to the three single-component models, two multi-component, hierarchical models were developed, where subunits of the target are intended to first assemble before being themselves combined to form the target. The two schemes presented here (D and E in figure 4.3) both consist of two particle types, and use an interaction alphabet larger than one to enforce exclusive interactions.

In model D (figure 4.3 middle row) the two species, D1 and D2, are mirror images of each other. Patches on opposite particle types all interact with equal strength $\varepsilon_{ij} = \varepsilon$, whereas patches on the same type of particle do not interact $\varepsilon_{ij} = 0$. In the target, D1 particles only neighbour D2 particles and *vice versa*. These interaction rules prevent incorrect structures that would result from the binding of two particles of the same type.

The patterned faces of the particles in model D each have a different number of patches, 3, 2 or 1. Links formed between faces with more patches will therefore have a stronger binding energy. This design feature is intended to promote a particular order of binding on the pathway to assembly. The rate of link formation between different faces will be mostly determined by the encounter rate, and so should be equal. However, interfaces with more patches will tend to be bound for longer, so we intend for aggregates deviating from our chosen pathway to be relatively short lived. In this manner we hope to promote a three-stage assembly as follows. First a D1 and D2 particle will bind through their three-patch faces to form a D1-D2 dimer. The two-patch faces of the D1 and D2 particles will now form a face of size $d \times 2d$ with four patches on the dimer. The remaining extended patterned face of the dimer has two patches. The dimer should then bind with another dimer preferentially through their four-patch faces, forming a square tetramer with four exposed patches on one face. Finally, two tetramers bind to form the target octamer. This process is illustrated in figure 4.7.

Evidence that assembly takes place by the intended hierarchical route is provided by figure 4.8, which shows a rapid decay of the population of isolated monomers as they become incorporated into larger structures. There are successive peaks in the populations of dimers and tetramers preceding the growth of the completed target. The order and peak in populations of our intended intermediates suggest that assembly of the final target does indeed proceed via the expected route. Crucially, the population of trimers is never high, implying that very few tetramers form by sequential addition of monomers to a dimer. Similarly, the populations of 5-, 6- and 7-member clusters are too low to be visible in figure 4.8. This suggests that the octamer target does generally form from the union of

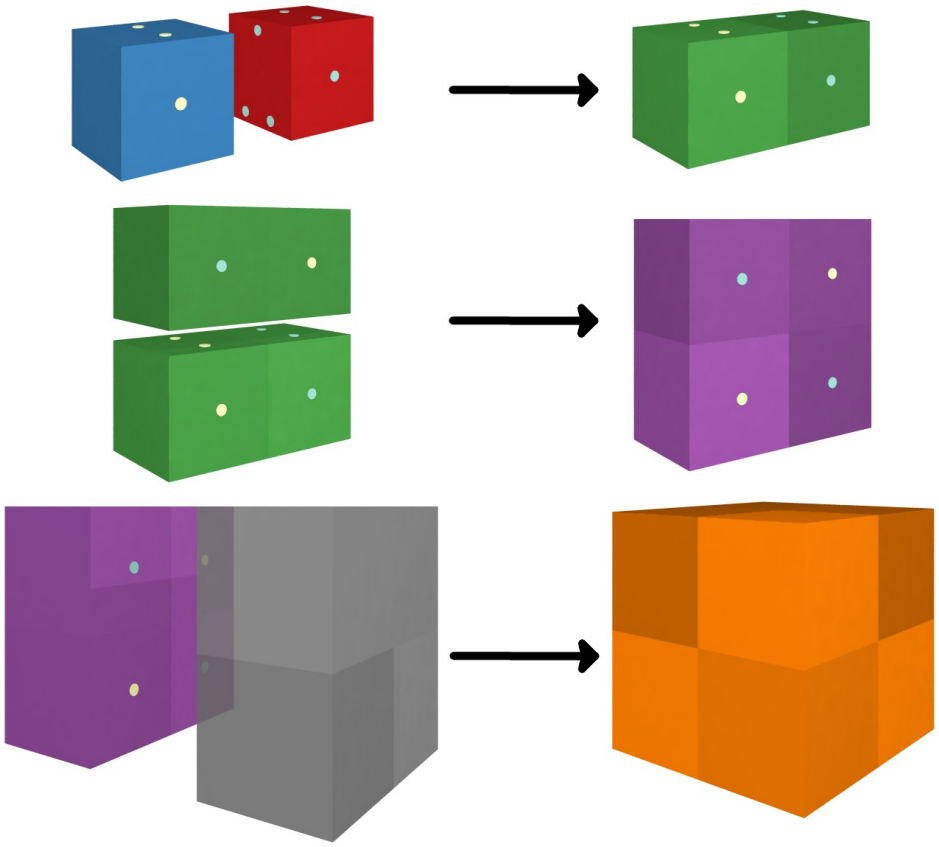

Figure 4.7: The intended hierarchical assembly pathway of model D. A pair of one D1 (red) monomer and one D2 (blue) monomer bind through their three-patch faces to form a dimer (green). Next two dimers bind through their four-patch faces to form a tetramer (purple). A pair of tetramers then bind through their only patterned faces to form the octamer target (orange).

tetramers, rather than either addition of monomer or other combinations of intermediates, such as a tetramer plus two dimers.

The pathway of assembly from individual trajectories like those obtained for model C were also found for model D. Figure 4.9 contains three cluster histories. The figures now contain some extra detail compared with before, with the colour of horizontal lines identifying recognised fragments using the same convention as figure 4.8. Dark segments now indicate aggregates that are not part of the intended assembly pathway.

The first two histories in figure 4.9 show decisively hierarchical paths to the target, with lines representing monomers, dimers and finally tetramers joining in turn. Brief black segments on the coloured lines indicate that additional building blocks temporarily attach to the developing structure, but these excursions always revert to the underlying

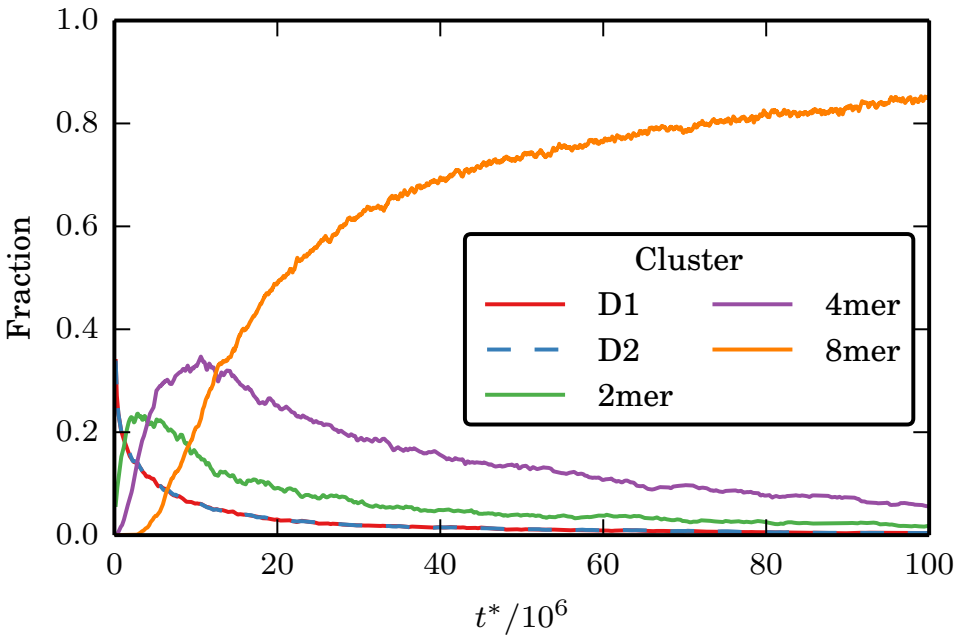

Figure 4.8: The fraction of building blocks in the hierarchical model D existing as monomers, as part of a complete target, or in correct fragments of the target, as a function of time averaged over 50 simulations at $T^* = 0.05$. The population of 5-, 6- and 7-member clusters is too low to be visible on the scale of this plot, and the lines corresponding to the two monomer types coincide almost exactly.

fragment in the examples shown. The third history in figure 4.9 shows that variations of the intended path are not ruled out in model D; in this example at $t^* \approx 8.5 \times 10^6$ a tetramer, a dimer and a D2 monomer combine to make a heptamer with the final D1 monomer being added some time later.

The predominant mechanism of assembly can change with conditions. For example, while the top two examples in figure 4.9 are representative of the hierarchical pathway typical at $T^* = 0.05$, at the lower temperature of $T^* = 0.04$ there is a late but steady production of the target cluster in model D. At this temperature the assembly is not hierarchical. Large, disordered aggregates form rapidly and, through internal rearrangement, correct subclusters may be released from a larger aggregate. These released fragments then proceed to form the target structure. In the example shown in figure 4.10 two tetramers emerge from separate larger aggregates and then combine directly to make the target. This path to assembly is similar to the budding mechanism identified in a one-component system of patchy spheres.[70] It also bears some resemblance to a mechanism of amyloid fibril formation, where polypeptides first condense into a disordered structure, before relaxing into an ordered amyloid structure.[136,137]

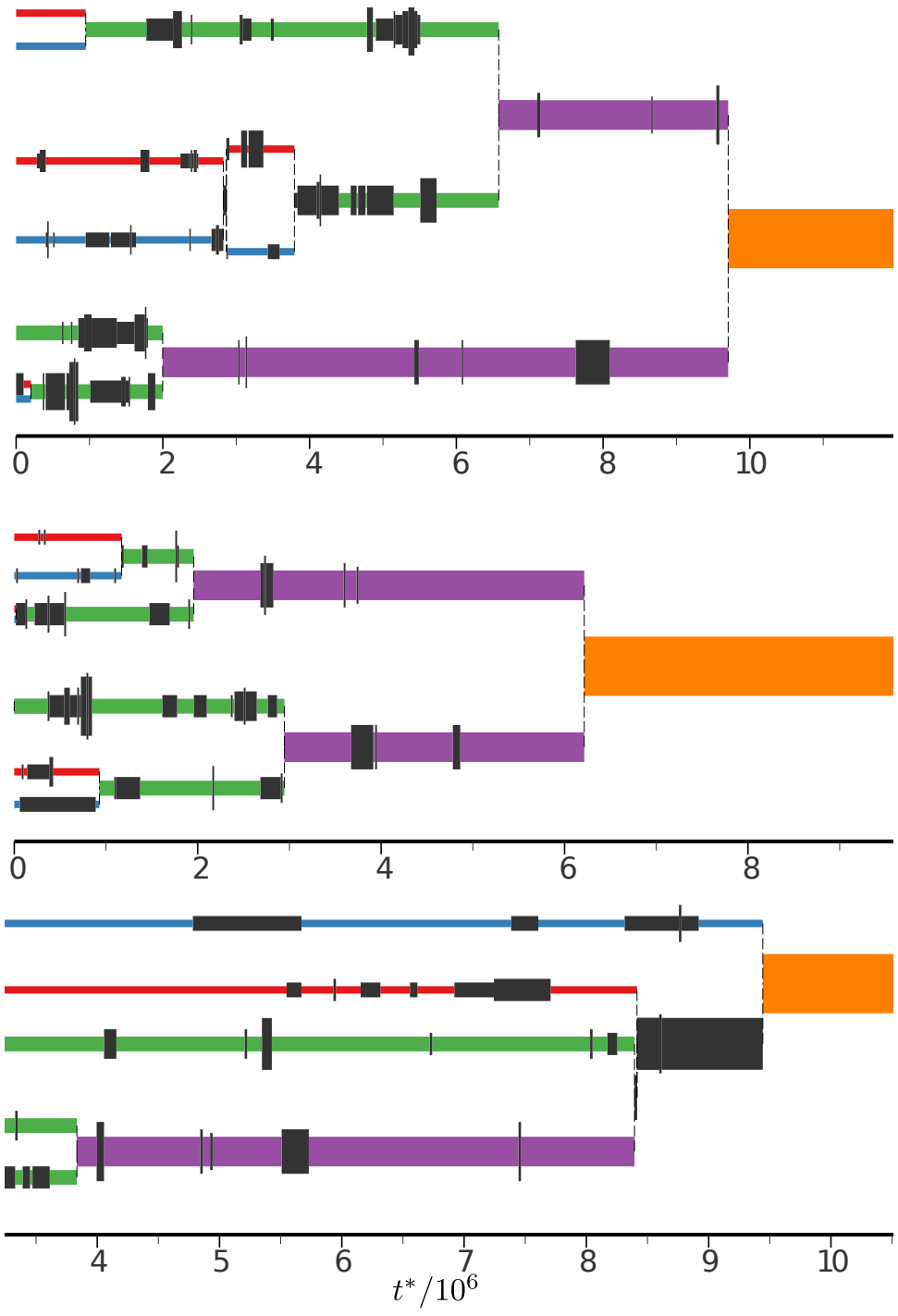

Figure 4.9: Extracts from the history of target clusters in the hierarchical model D, taken from one simulation at $T^* = 0.05$. Coloured lines correspond to the scheme in figure 4.7. The first two histories are examples of clear-cut hierarchical assembly, while the third history shows a rarer variation of the route where a cluster of seven particles is formed and a monomer then joins to complete the octamer.

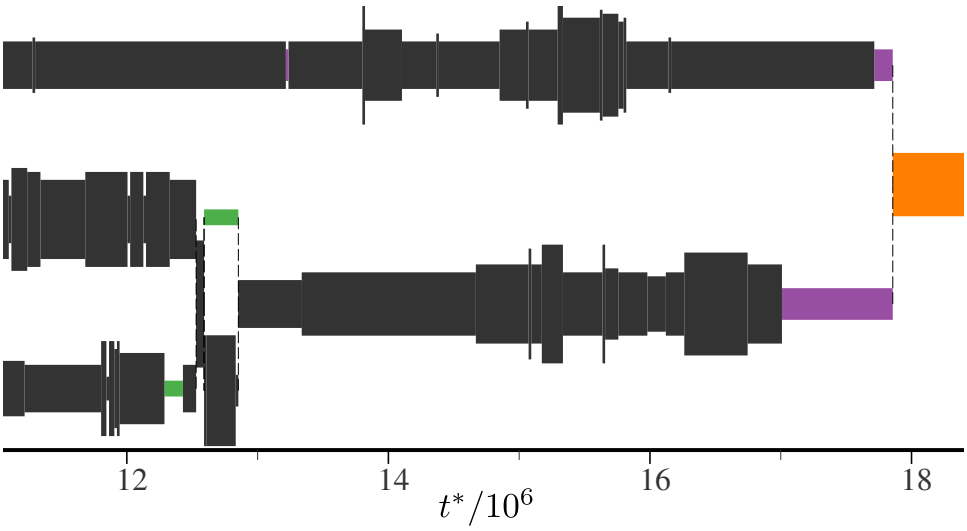

Figure 4.10: An extract from the history of an octamer forming at $T^* = 0.04$ in model D. Two correct tetramer fragments break apart from larger, unrecognised aggregates and then bind to form the octamer target.

As with the sequential models, $t^*_{1/2}$ at a range of temperatures was determined for model D. The green line in figure 4.5 shows that model D is quite robust with respect to changes in temperature, but the simpler sequential models B and C both do better in terms of speed and temperature range. The strategy of assembling sub-components of the target simultaneously does not seem to have provided model D with an advantage over sequential assemblers. This is in agreement with previous simulations of assembly of a hierarchical structure.[99]

An alternative hierarchical model, E (figure 4.3, bottom row), employs a two-step process in which two tripod-shaped tetrameter fragments form and then interlock to make the target as illustrated in figure 4.11. Two particle types are again required but in this model, unlike model D, they must be combined in a 1:3 ratio. A three-letter alphabet of patches is required to steer the model towards the anticipated pathway. Patches on the E1 monomers, which form the apex of a tripod, bind exclusively to the patches arranged in a triangle on one face of the E2 monomer. The patches on E1 particles do not interact, since this would make them identical to the one-component model C, and neither do the corresponding patches on the E2 particles. The third type of patch appears in the centre of two adjacent faces of the E2 monomer. These patches bind exclusively amongst themselves. Six pairs of these patches come together when two tripods bind correctly.

The two steps in the assembly of model E occur on well separated time scales. figure 4.12 shows that a large intermediate population of tetramer tripods rapidly assembles

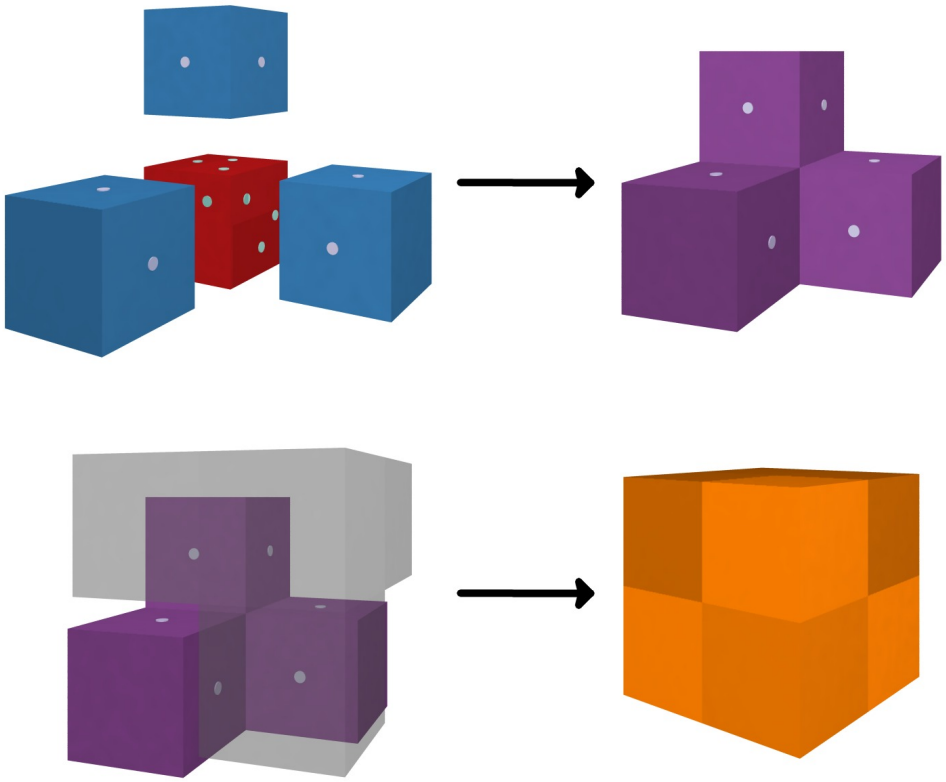

Figure 4.11: The intended hierarchical assembly pathway of model E. Particle E1 (red) binds to the three E2 monomers (blue) to form a tetramer tripod (purple). Two tetramers may then link to form the octamer target (orange).

from monomers, but complete targets only emerge slowly. The cluster history shown in figure 4.13 is typical for model E and illustrates the sequential addition of monomers to make a tripod, which then undergoes a long and uneventful trajectory before finally pairing up to complete the target. Despite the substantial energetic reward for the second step of assembly, there is a stringent steric requirement on the orientations in which two tripods must approach each other in order to bind. The purple line in figure 4.5 shows that model E is almost always slower than the other hierarchical scheme D. However, model E does have an advantage at low temperatures, where successful assembly persists into the region in which models B and D have become very slow. Success at low temperature indicates the ability to avoid kinetic traps.

The assembly of model E may benefit from taking place in only two steps. This allows us create a wider energetic separation between those faces which we want and do not want to bind in the first step of assembly. At this stage, bonding may occur between a three-patch face, or a face with only a single patch. This is in contrast to model D, where there

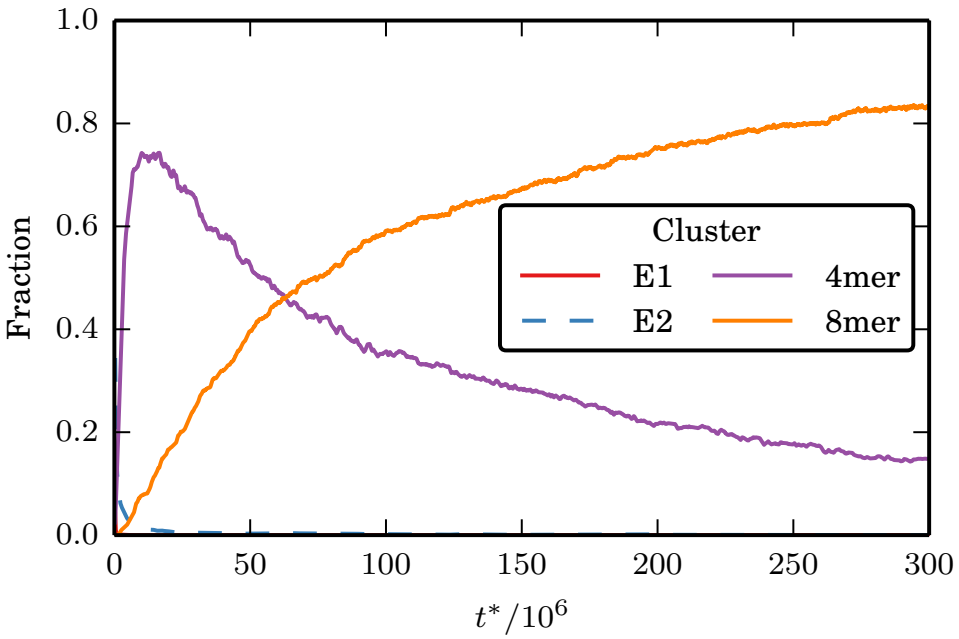

Figure 4.12: The population fraction of the target and correct fragments as a function of time for the hierarchical model E at $T^* = 0.044$. E1 monomers are rapidly incorporated into clusters and the corresponding line on the plot is barely visible. Note the difference in the time scale compared with the other models, in particular the long time taken for tetramers to assemble into the octamer target.

is also a face with two patches. Bindings that are off the intended assembly pathway are therefore more likely to be short lived. The tripod intermediate is also relatively inert, and misaligned encounters are unlikely to lead to long-lasting aggregates, as it is only possible to satisfy two patches in anything but the correct conformation. Hence, there is an absence of kinetic traps which proves advantageous to model E at low temperature.

4.3.3 Addressable Assembly

The final assembly strategy presented here is the fully addressable limit, comparable to DNA bricks, where all components of the target are distinct. Model F therefore consists of eight different particles for the case of our octamer target. In order to establish a basis for comparison of model F with the other models, the same pattern of patches as in the most efficient design so far is used, that is the sequential model C (figure 4.3). However, each patterned face of each particle type in model F is now given a different patch type, making a 24-letter alphabet. Building block F1, for example, has three faces each of which binds exclusively to a particular face on one of the building blocks F2, F3 and F4.

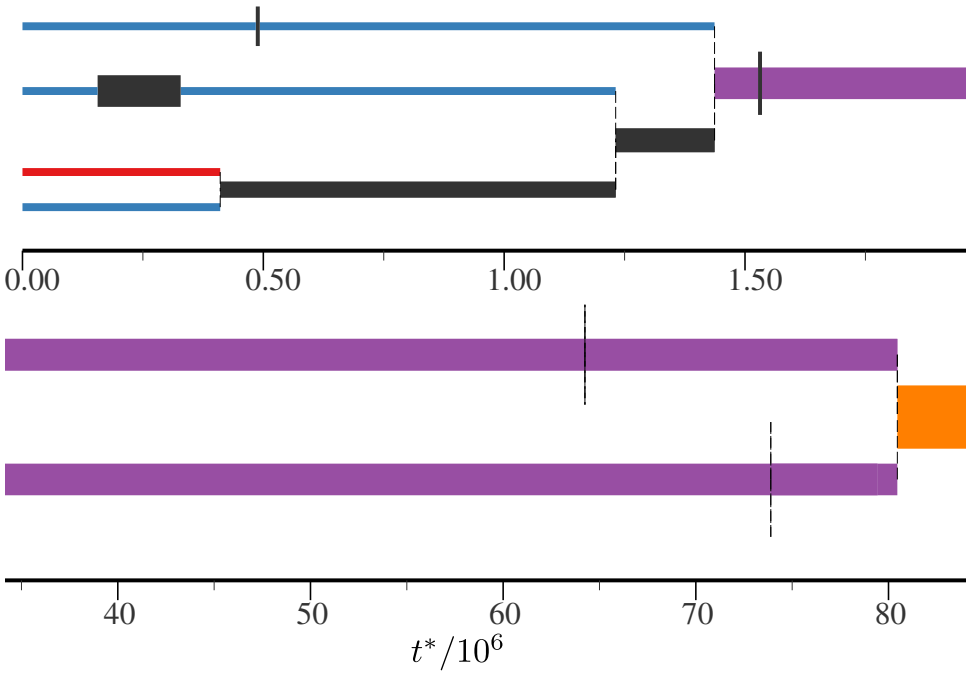

Figure 4.13: Example cluster histories for a single tetramer (top) and the formation of an octamer from tetramers on a much longer time scale (bottom) in hierarchical model E.

In this way all interactions that are not present in the target structure are removed. All interacting pairs of patches have the same strength ($\varepsilon_{ij} = \varepsilon$), and the energy of the target is therefore the same as in model C.

The specificity in model F rules out many of the fragments that would be incompatible with the final structure and which could act as kinetic traps. However, the requirement of neighbouring particles to be of a particular species also removes a large number of fragments and targets that have the right geometry but involve a combination of particle types that are not complementary in the addressable design. In the equivalent one-component case, model C, any combination of particles in the right geometry is automatically a correct fragments of the target cluster. Therefore, the labelling of particles greatly reduces the number of paths by which the target may be built and amounts to an entropic disadvantage that shifts optimal assembly of model F to lower temperatures than model C, as shown by the orange line in figure 4.5.

The reduced number of paths generally also makes model F slower to assemble. In the one-component model C, an encounter between any two monomers can always initiate assembly and there are nine combinations of pairs of faces on the two particles that may bind correctly. In contrast, a building block in the addressable case F must diffuse until it

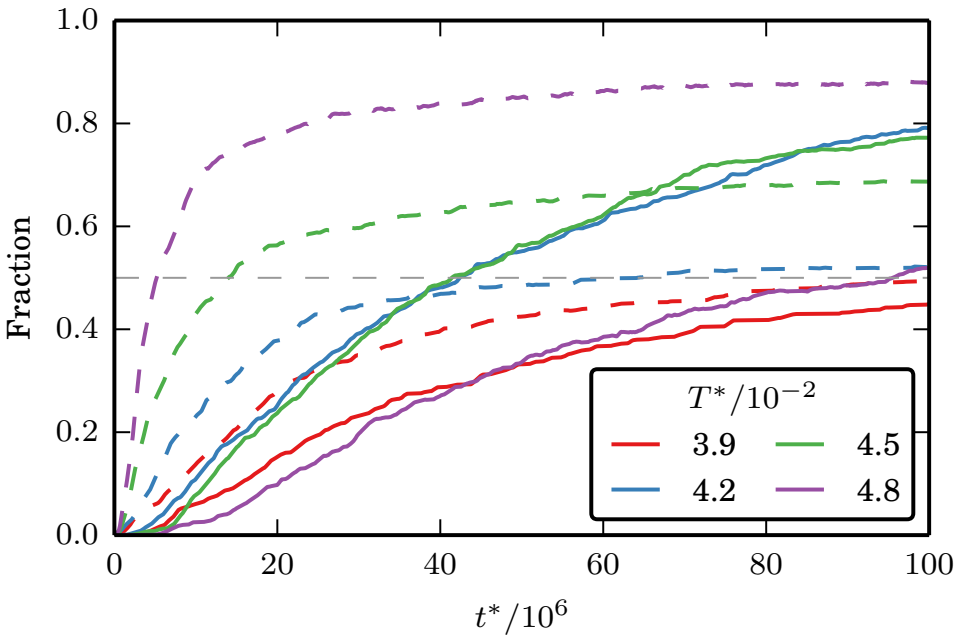

Figure 4.14: Comparison of the population fraction of the target structure as a function of time at four temperatures for both the addressable model F (solid lines) and the single-component model C (dashed lines). The dashed horizontal line is the 50% threshold at which $t^*_{1/2}$ is defined.

meets one of the three other species with which it can bind, and binding may only occur through one of the 36 possible combinations of faces. By simulating model F with the same total density as the single-component models, the density of the individual species of building blocks is lower.

The addressable model is less robust with respect to temperature changes than the other models examined so far. A narrow range of temperature in which assembly is possible is a general feature of addressable systems observed in other theoretical[83,88] and experimental[86] work. Despite the exclusion of incorrect fragments, a thermodynamic yield of target at low temperatures can still be prevented by the formation of multiple partially completed structures that starve the system of the building blocks required to complete any one target. At high temperature, assembly suffers from the need for the right combination of building blocks to encounter each other within a time frame shorter than the lifetime of a transient partially completed structure. This need for a rare fluctuation is consistent with a recent analysis showing that addressable assembly proceeds by a non-classical nucleation process.[89]

A more detailed comparison between the addressable model F and the one-component

model C shows that while the addressable case is slower, it may nevertheless have superior performance in terms of yield at long times. Figure 4.14 shows that, at some temperatures, the yield of target in model F rises slowly but steadily beyond the point where the rate of assembly in model C slows down dramatically. Hence, if the threshold at which the assembly time is recorded for figure 4.5 were raised above 50% yield (horizontal line in figure 4.14) the comparison would put the addressable strategy in a more favourable light.

The assembly pathways of model C may also be contrasted with the fully addressable case. Figure 4.15 shows the contrasting evolution of fragment populations in the two cases at an intermediate temperature of $T^* = 0.045$. The upper panel shows that monomers in model C are almost instantly incorporated into six- and seven-membered clusters that then struggle to be completed due to the lack of monomers. For the addressable case however (lower panel of figure 4.15) the decay of monomers is much more gradual. Although a low background population of sizeable fragments arise, these partially completed structures can systematically be completed, causing the yield of target structures to continue rising at times when model C is virtually stuck. At this temperature model C has entered the monomer starvation regime, where incompatible fragments and a lack of monomer slow assembly, as can be seen by the rising values of $t^*_{1/2}$ at and below this temperature (figure 4.5). By contrast, model F is within the optimal temperature window for assembly. Monomer starvation will occur for model F at lower temperatures.

So far, the addressable model has always used perfectly exclusive interactions. This is where the strength of interactions between patches is ε when these patches are on adjacent faces in the target structure and 0 otherwise. However, the patchy model gives full control over the interactions between patches and so it is possible to introduce nonspecific interactions between addressable patches. This is an important factor to consider as DNA mediated interactions are not entirely exclusive. Nonspecific interactions between all types of patch that previously did not interact were added to model F. All nonspecific interactions are of equal strength, which is defined as a fraction of the specific interaction strength $\varepsilon_{\mathrm{ns}}/\varepsilon$. This is not the case for nonspecific interactions between DNA strands, which depend strongly on the sequence of nucleotides. However, as a coarse representation nonspecific interactions of uniform strength should be sufficient to capture the emergence of competition to the native interactions.

The time to 50% yield for a range of nonspecific interaction strengths is presented in figure 4.16. It is also important to note that although the interactions in this model are no longer entirely specific, the cluster is still treated as fully addressable. This means that

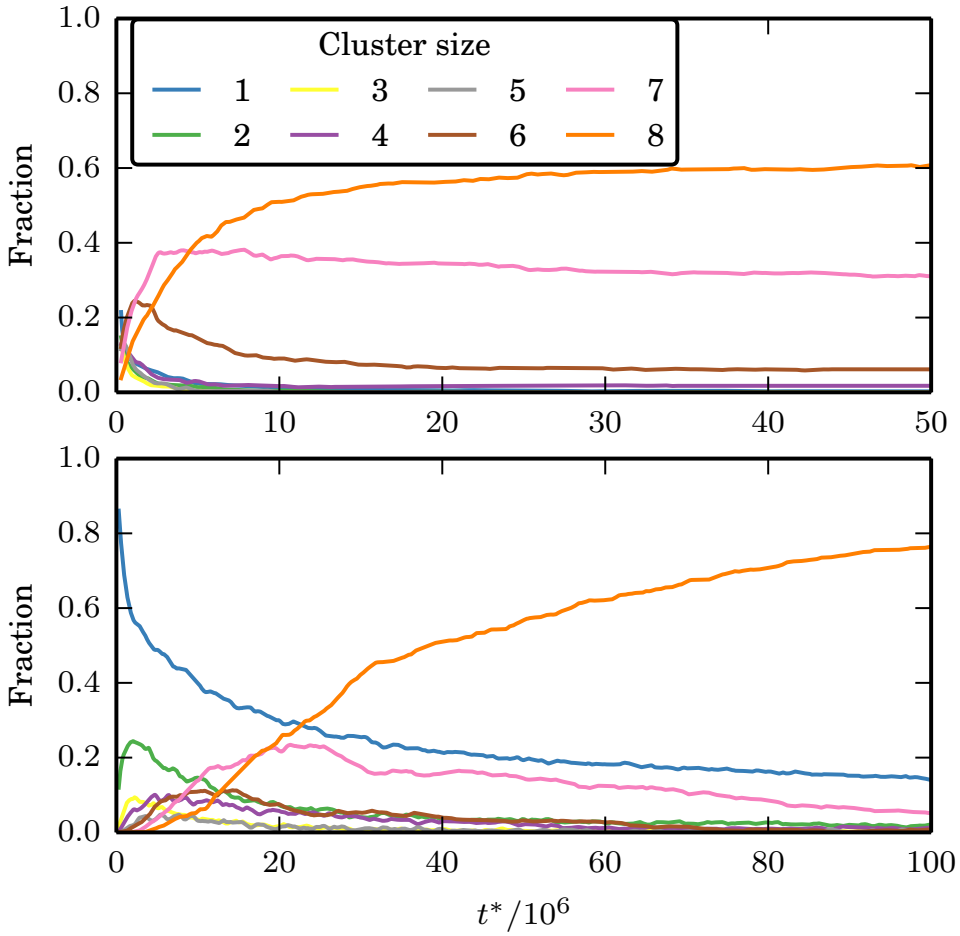

Figure 4.15: Populations of fragments of 1 to 8 building blocks during assembly at reduced temperature $T^* = 0.045$ for the one-component model C (top) and addressable model F (bottom).

the yield refers to clusters of the correct geometry and composition with all particles in the right positions and orientations. If the nonspecific interaction is equal to the specific interaction ($\varepsilon_{\mathrm{ns}}/\varepsilon = 1$) the model should behave like the equivalent sequential model C. In this limiting case there would be no preference for clusters to contain only one of each particle type, or for the particles to bind to their intended neighbours.

Small strengths of nonspecific interaction ($\varepsilon_{\mathrm{ns}}/\varepsilon < 0.3$) have very little impact on the nature of assembly. There is a small improvement in the rate of assembly towards low temperatures. It may be that the introduction of some nonspecific interaction helps condense particles making assembly easier. However, it should be noted that there does not appear to be a significant difference in the assembly pathway, or the kinds of traps encountered with small nonspecific interactions. As the nonspecific interaction strength

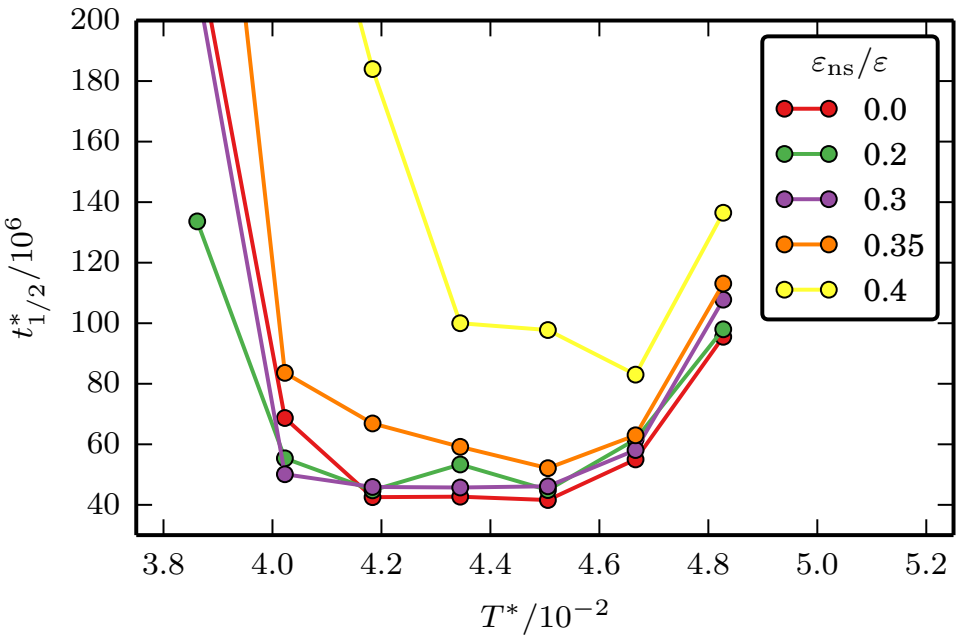

Figure 4.16: The time $t^*_{1/2}$ taken to reach 50% yield of the target with all particles in their correct positions as a function of reduced temperature. The strength of interaction between patches on faces not bound in the target is given as a ratio of the nonspecific interaction energy ε_{ns} to the specific interaction energy ε.

is increased beyond this point assembly becomes worse. At $\varepsilon_{ns}/\varepsilon = 0.4$, $t^*_{1/2}$ is significantly larger for all temperatures simulated. Furthermore, the range of temperature in which assembly takes place has become narrower. Beyond this point, nonspecific interactions are strong enough that incorrect bindings between faces not connected in the target occur frequently. Although the nonspecific interactions are still relatively weak, the large number of possible incorrect bindings reduces the likelihood of assembling a cluster with the correct composition. At $\varepsilon_{ns}/\varepsilon = 0.5$, 50% yield was not achieved at any temperature within the simulation time.

4.4 Discussion

For the small, highly symmetrical octamer target used in this work, the fastest and most robust strategy for self-assembly was the sequential addition of identical building blocks with a single type of interaction site. This result is consistent with the observation of efficient single-component strategies in some, highly symmetric virus capsids. Furthermore, optimal assembly occurs at temperatures where bond formation is easily reversible.

This has previously been identified as an important factor in avoiding kinetic traps and ensuring reliable assembly in an idealised model of capsid assembly.[25,75]

The minimal viable design (model B) has two patches on each of its interacting faces, which create an effective torsional constraint that suppresses amorphous aggregation. Additional patches can be used to improve the efficiency and reliability of self-assembly somewhat further (model C) at the expense of greater complexity of building block. The importance of a torsional potential has also been noted in more coarse-grained models.[71,73] By implementing angular restrictions on binding explicitly through a pattern of countable patches, rather than by an implicit effect built into the potential, the patchy cube model provides a way of quantifying the information required to implement a particular type of interaction.

Compared to the models for sequential addition of monomers, hierarchical approaches require a greater complexity of design, featuring multiple particle types and specifically interacting patches. However, despite the extra specification in design, the hierarchical strategies displayed little advantage over the minimal model. This may be at least in part due to the high symmetry of the target with no intrinsic modular structure, which naturally benefits assembly from identical components. Nevertheless, the benefit of multi-step assembly over sequential assembly has been called into question, even for structures with a hierarchical structure.[99]

The relatively small size of the structure may also help favour a sequential rather than a hierarchical mechanism. Experimental and theoretical evidence supports the idea that some capsids may assemble in a two-stage scheme, with capsomeres existing as the intermediate state between proteins and the capsid.[102,138,139] In this case a hierarchical pathway may beneficial due to the large size of the target, as with many more particles in the target the number of competing states acting as kinetic traps and hindering assembly is greater.

The design with highest complexity studied here was the fully addressable model F. This model bears some resemblance to the clusters formed by DNA bricks.[86,87] The simulations presented here found that configurations featuring incompatible fragments were a major factor in slowing assembly in the addressable case. It was previously believed that this type of kinetic trap would make addressable structures made from DNA impossible to achieve with acceptable yields.[140,141] However, this is not the case for DNA bricks, where a large, late nucleation barrier helps to prevent such states from accumulating.[88] Nevertheless, kinetic traps of this kind may be an important consideration for the assem-

bly of other addressable clusters. Other theoretical investigations of self-assembly often consider only the building blocks required for one copy of the target, and so are not able to capture this important source of frustration.

To ensure reasonable comparison between the models, all simulations were conducted at the same total density and temperature remained constant. This approach neglects the important impact that the physical conditions have on self-assembly. The simulation density was chosen based on preliminary simulations. It is unlikely to be ideal for any model, and indeed the optimal density may vary between models. For example, introducing unique particle types while keeping the density constant in model F has the effect of reducing the concentration (and chemical potential) of the individual building blocks. A more sophisticated temperature protocol could also help optimise self-assembly. A slow quench has been demonstrated to be important for successful assembly of DNA bricks, both experimentally[86] and in theory.[83,89]

4.5 Conclusion

In this chapter the patchy model described in chapter 3 was used to study a range of self-assembly strategies for the same target. All of these are examples of programmable assembly, where a desired outcome and mechanism of assembly are ensured through designing the building blocks. In contrast to more coarse simulations of patchy particles, where an interaction may be represented by a single interaction site, the interfaces in our model feature explicit interactions. It is therefore possible to capture aspects of assembly that may be neglected by other models, such as the susceptibility of certain designs to incorrect binding. The simulations presented here have shown that different assembly pathways may be promoted through the design of a patchy interface. Through the use of a single model this range of strategies may be fairly compared.

The building blocks presented here were designed using intuition, by placing attractive patches on faces so that they can reasonably be expected to promote the target structure, while avoiding the most obvious sources of competition by choosing patch distances carefully. For example the triangular arrangement of patches in model C was intentionally chosen to not be equilateral and this would result in a triply degenerate ground state for dimers. The designs presented here cannot therefore be considered optimal and there is a lot of scope for improvement. Optimising particle design in will be explored in chapter 6. In that chapter an algorithm will be presented which optimises the patterning of faces to

create addressable interactions.

The addressable model F displayed quite distinct assembly characteristics compared to the equivalent sequential model. The formation of fragments incapable of joining to create the target was particularly pronounced in the fully addressable model. To reliably assemble structures with addressable complexity it is necessary to understand the conditions which lead to these frustrated configurations. A method for quantifying the level of frustration in a system of addressable building blocks will be presented in chapter 5. This algorithm will be used to explore aspects of design that can aid in avoiding such configurations. Both chapters 5 and 6 will look at the assembly of more complex targets that may be larger or less symmetrical than the octamer cube.

CHAPTER 5

CONTROLLING FRAGMENT COMPETITION ON PATHWAYS TO ADDRESSABLE SELF-ASSEMBLY

5.1 INTRODUCTION

The successful demonstration of addressable self-assembly in the cases of DNA bricks[86,87] and tiles[85] is perhaps a surprising result. For addressable assembly to work, there must be sufficiently strong energetic interactions between particles to overcome the large entropic cost of forming a cluster with no permutations of building blocks. Self-assembling systems relying on relatively few yet strong interactions are known to be particularly susceptible to kinetic traps.[142]

Furthermore, synthesising building blocks at the precise stoichiometry required is challenging. It was long thought that small discrepancies from the ideal concentrations of building blocks would make assembly very poor. The idea was that particular components with a lower than average concentration would become exhausted and stall assembly.[140] In fact this problem was thought to be so critical that DNA origami,[22] a technique where DNA "staples" fold a long single stranded DNA (ss-DNA) molecule into shape, was developed partially to avoid it.[141] The authors of the DNA tile and brick work suggested that this problem is avoided by a slow nucleation process. This has been supported by theoretical work[88,92] which identified a large nucleation barrier.

Further studies by Jacobs *et al.*[83,89] established that the assembly of DNA brick structures occurs through a non-classical nucleation procedure. Due to the unfavourable entropy of the target, it may not be the most stable structure in all conditions. Indeed, they found that the temperatures at which nucleation is rate-limiting and equilibrium yield of the target is high do not overlap. It was therefore proposed that a time-dependent tempera-

ture protocol is required where first the nucleation and growth of a stable partial structure occurs, before lowering the temperature so that the target is the free energy minimum, allowing it to assemble completely. This proposal is consistent with the use of a slow annealing procedure to assemble the DNA structures, although isothermal assembly has been demonstrated in some cases.[87]

The simulations of the fully addressable octamer in chapter 4 revealed a kinetic trap which significantly hindered assembly at low temperatures, but was not an important source of trapping in the corresponding one-component model. In this trap, all particles in the system became incorporated into aggregates that are not able to combine to form the target without first breaking up. The source of this problem is an overlap in the sets of building blocks in different fragments. As an addressable cluster contains only one of each building block, any overlap between a pair of aggregates renders them incompatible. Throughout this chapter such configurations will be referred to as frustrated. For assembly to proceed from a frustrated configuration, the overlap in sets particles must be removed through the fission of aggregates.

The model and algorithm presented in the preceding chapters provide a good basis for studying the frustration in addressable systems. This is in part because the dynamical algorithm evolves the system in time in a reasonable manner and so captures the formation of incompatible fragments after quenching. Also, simulating multiple replicas of a complete set of building blocks simultaneously is important because with only one replica it is impossible to generate frustrated configurations. Many of the existing simulation studies of addressable self-assembly only include one replica and so neglect this effect entirely.

Furthermore, the versatility of the patchy cube model gives the opportunity to explore a number of factors which may help avoid frustration. For example within the patchy model it is possible to arbitrarily change the strength of binding between any pair of patches. Jacobs *et al.* identified that heterogeneous binding energies were beneficial to ensuring error-free assembly.[83,89] The authors found that if bond strengths were chosen randomly, strong links were unlikely to be arranged to favour small, potentially incompatible fragments, instead promoting large pre-critical clusters.

It is also possible, as seen in chapter 4, to devise different sets of building blocks for the same target cluster. In terms of fully addressable targets, this can take the form of changing the connectivity of the structure, that is the network of bonds which hold the cluster together. The connectivity of a cluster may be used to offset entropy loss. If the network of links in a cluster contains loops, then when a loop is completed, a link

is made between two particles which are already bound to the aggregate. We therefore gain binding enthalpy for a smaller entropy loss, as both bound particles have already lost a significant amount of translational and rotational freedom. Completing loops within a structure therefore gives an extra free energy payback.

Aspects of connectivity, in particular coordination number, were explored by Jacobs *et al.*.[89] They found that the low coordination number of four of DNA bricks is important to their success in assembly. Higher coordination numbers of six and twelve produced structures well described by classical nucleation theory, possessing large free energy barriers to nucleation, and with all intermediate structures unstable with respect to a monomer fluid. Such coordination numbers will therefore require a large supersaturation to encourage assembly, which they propose will lead to significant kinetic traps in a system with addressable complexity.

In order to investigate the effects of the particles' design on frustration it is necessary to quantify it. In section 5.3 a general procedure for quantifying frustration in addressable systems will be presented. This method works by considering the fragments present at a particular time and then combining them into as many complete targets as possible. The maximum number of targets that may be created is termed the completability. If the completability is less than the number of replicas of each building block present then we know the system has entered a frustrated configuration. In this way it is possible to distinguish between a system that has become stuck because of frustration, and one where assembly is simply slow, or not completed due to the thermal instability of the target. The algorithm is not specific to the patchy cube model and so is applicable to any instance of addressable self-assembly.

The completability algorithm will be applied to the final configuration of simulations after a set time. It will prove useful in revealing successful approaches to controlling addressable assembly and avoiding frustrated configurations. The simulations will cover a range of approaches to avoid frustration including varying connectivity, semi-hierarchical strategies and random bond strengths. These approaches will be trialled on a pair of targets with equal number of components but contrasting structure. One target is compact and tightly packed whereas the other has an open, cage-like structure.

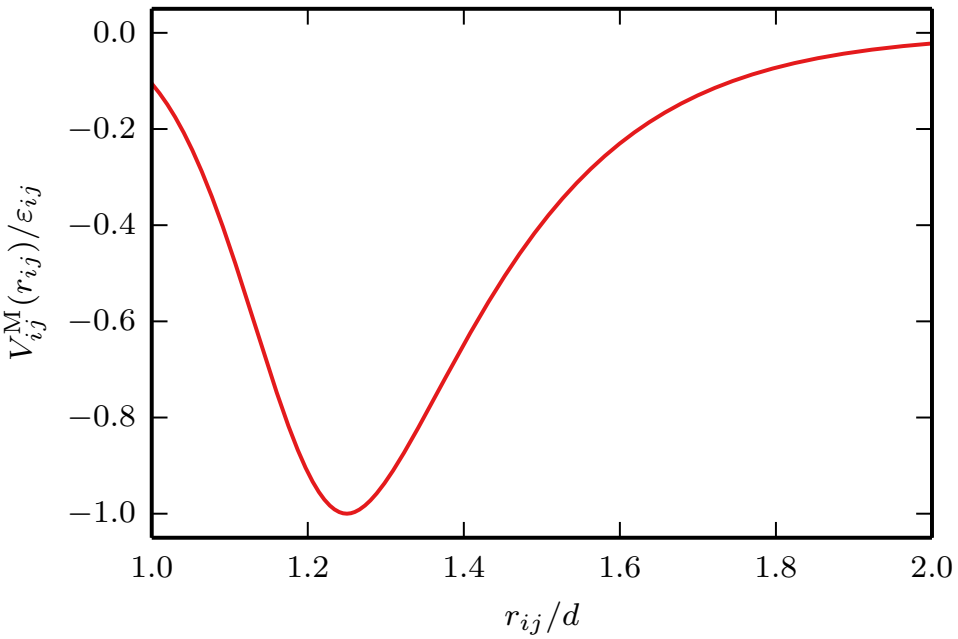

Figure 5.1: The form of the modified potential V_{ij}^{M} as a function of patch–patch distance r_{ij}.

5.2 Modified Patchy Potential

In this chapter a variant of the patchy particle model introduced in section 3.2 is used. These modifications were made in order to eliminate two sources of kinetic traps which are not of interest, in order to focus on the assembly pathways. The original model will be restored in chapter 6 where the fine detail of interactions will be investigated. The traps the modified potential seeks to avoid are the incorrect binding of particles with multiple patches (like model B in chapter 4 (figure 4.3)), and stalled assembly due to building blocks not being able to fit into tight spaces. Both of these sources of kinetic traps are important to the kinetic of self-assembly and may be accounted for by the patchy model. However, they are not the concern of this chapter and so the model was adapted to eliminate them so that they did not affect assembly and disrupt the results. The modified potential features only a single patch per face, with an implicit torsional potential to ensure particles orientate correctly when bound. The minimum of the potential has also been shifted outward so that it no longer occurs when the faces of two particles are touching.

The modified pairwise interaction between patches takes the form of a step potential which switches between a Morse potential and Gaussian repulsion around the minimum of the potential. The value, gradient and curvature of the two functions are identical at the

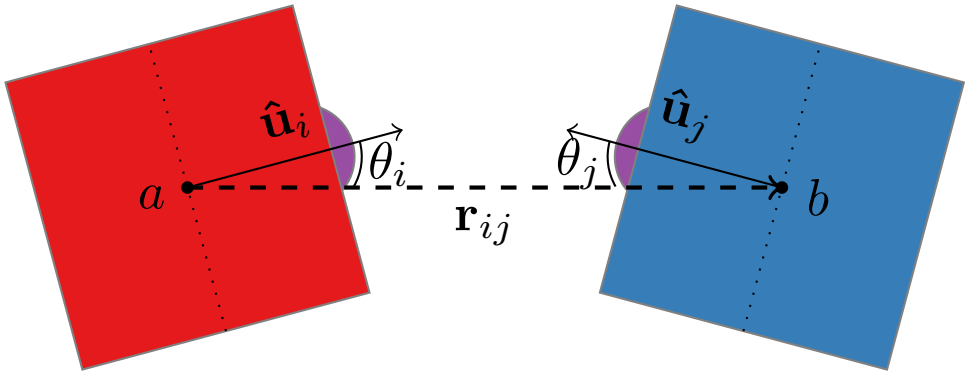

Figure 5.2: Schematic representation of the interaction between patches i and j on two cubes a and b, showing the definition of the angles θ_i and θ_j.

point of handover. The interaction between two patches i and j is given by

$$V_{ij}^{\mathrm{M}}(r_{ij}) = \begin{cases} \varepsilon_{ij}\left[e^{-2\alpha(r_{ij}-1.25d)} - 2e^{-\alpha(r_{ij}-1.25d)}\right] & \text{if } r_{ij} > 1.25d \\ -\varepsilon_{ij}\exp\left(-\alpha^2(r_{ij}-1.25d)^2\right) & \text{otherwise}\,, \end{cases} \tag{5.1}$$

where r_{ij}, α and ε_{ij} have the same definitions as before (section 3.2). Figure 5.1 shows the form of the potential. As before each patch is embedded at a depth of $d/2$ (figure 5.2). The distance dependence of the exponent in equation 5.1 therefore ensures that the minimum of the interaction between two patches occurs when two particles' faces are separated by a distance of $0.25d$. This separation helps ensure that the hard repulsion of particles does not prevent dense clusters from forming because of the need to insert a building block into a tight space. As before the potential is truncated at a distance of $2d$. To avoid a discontinuity at the cut-off, the potential is shifted by $V_{ij}^{\mathrm{M}}(2d)$. The potential is then scaled so that the value of the potential at the minimum is $-\varepsilon_{ij}$.

Each patch has an intrinsic orientation, pointing to the centre of one of the four edges of the face which the patch belongs to. A torsional attenuation of the interaction between two patches gives the effect of an arrangement of several patches at a more coarse-grained level. The attenuation has a Gaussian form, depending on the orientation vectors of a pair of interacting patches,

$$V^{\mathrm{tor}}(\mathbf{r}_{ij}, \hat{\mathbf{v}}_i, \hat{\mathbf{v}}_j) = \exp\left(\frac{-\phi_{ij}^2}{2\sigma_{\mathrm{tor}}^2}\right)\,, \tag{5.2}$$

where ϕ_{ij} is the torsional angle between the two orientation vectors ($\mathbf{v}_i$ and $\mathbf{v}_j$) of patches i and j. The width of the Gaussian, σ_{tor}, controls how quickly the potential is attenuated as the torsional angle deviates from 0. σ_{tor} is set at 0.7 which was found to be suitably

strict to ensure particles orientated correctly.

The overall form of the potential between patches is therefore

$$V_{ij}^{\text{patch}}(\mathbf{r}_{ij}, \hat{\mathbf{u}}_i, \hat{\mathbf{u}}_j) = \left[\frac{V_{ij}^{\text{M}}(r_{ij}) - V_{ij}^{\text{M}}(2d)}{\varepsilon_{ij} - V_{ij}^{\text{M}}(2d)}\right] \times \Theta(2d - r_{ij}) V^{\text{ang}}(\hat{\mathbf{r}}_{ij}, \hat{\mathbf{u}}_i, \hat{\mathbf{u}}_j) V^{\text{tor}}(\mathbf{r}_{ij}, \hat{\mathbf{v}}_i, \hat{\mathbf{v}}_j)\,, \quad (5.3)$$

where $\mathbf{r}_{ij} = r_{ij}\hat{\mathbf{r}}_{ij}$ and Θ is the Heaviside step function, which enforces the cut-off at $r_{ij} = 2d$. The total interaction energy between any two particles is given by

$$V_{ab}^{\text{cube}}(\mathbf{r}_{ab}, \mathbf{\Omega}_a, \mathbf{\Omega}_b) = \sum_{i \in a} \sum_{j \in b} V_{ij}^{\text{patch}}(\mathbf{r}_{ij}, \hat{\mathbf{u}}_i, \hat{\mathbf{u}}_j) \Delta_{ij}(\hat{\mathbf{r}}_{ij}, \mathbf{\Omega}_a, \mathbf{\Omega}_b)\,, \quad (5.4)$$

which has the same form as the original potential (equation 3.4) but with a different definition of V_{ij}^{patch}.

5.3 MEASURING COMPLETABILITY

In order to assess the success of a set of particles in assembling a target, we have devised a completability algorithm that identifies any frustration which may prevent a system from reaching maximum yield. The completability algorithm can be applied to any instance of addressable assembly, where each particle in the target cluster is unique and so appears exactly once in each complete cluster. This technique was developed in collaboration with David Bourne of the Department of Mathematical Sciences at Durham University.

The algorithm identifies all aggregates present in a given configuration, and determines the maximum number of complete target clusters that can be formed by combining them without breaking them up. An example of this concept is shown in figure 5.3. The sorting of aggregates is achieved by expressing the problem in terms of integer linear programming (LP).

LP is a method to maximise or minimise a linear function, subject to a set of constraints, each of which may be either an equality or an inequality. All LP problems may

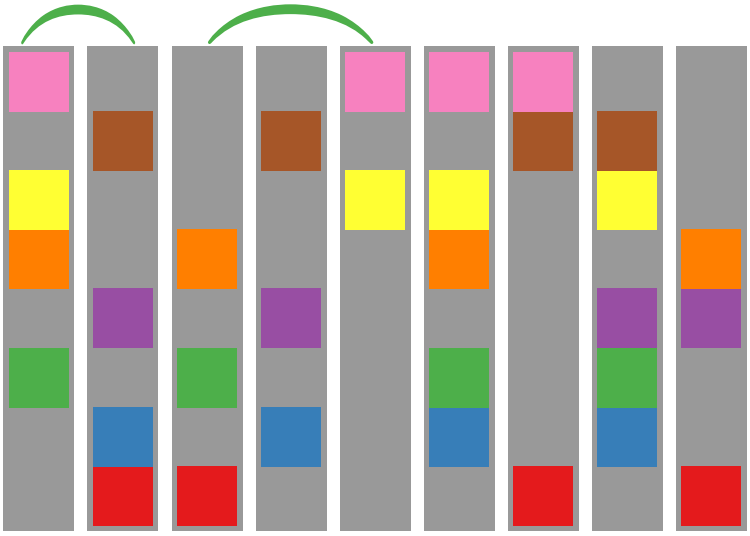

Figure 5.3: A schematic of the completability algorithm at work. Here a cluster is composed of eight components, each with a unique colour. There are four copies of each component, enough to make a maximum of four complete clusters. The fragments in this example are represented by placing the coloured components on a bar together. The algorithm finds that the sets of fragments covered by green ties may be combined to form two complete targets. No further targets may be made, nor may any of the remaining aggregates be combined to form larger aggregates without first breaking up the aggregates into smaller fragments.

be expressed in the most general form,

$$\begin{aligned} \text{maximise} \quad & \mathbf{c}^\top \mathbf{x} \\ \text{subject to} \quad & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ \text{and} \quad & \mathbf{x} \geq 0\,. \end{aligned} \tag{5.5}$$

Here, $\mathbf{x}$ is a variable vector of length m. The values of $\mathbf{x}$ are changed in order to find the optimal solution to $\mathbf{c}^\top\mathbf{x}$. The vectors $\mathbf{c}$ and $\mathbf{b}$ are of length m and n respectively. The values of both $\mathbf{c}$ and $\mathbf{b}$ are known. $\mathbf{A}$ is a known $m \times n$ matrix. The vector $\mathbf{c}$ contains the coefficients of the variables $\mathbf{x}$ in the objective function ($\mathbf{c}^\top\mathbf{x}$), while $\mathbf{A}$ and $\mathbf{b}$ define the constraints.

LP is often used to solve assignment problems, which commonly take the form of assigning N workers to N tasks in the optimal way, given that the performance of individual workers on different tasks may vary. The completability algorithm instead sorts the fragments present at a given time in a simulation into as many complete target clusters as possible. As the composition of fragments and number of targets are integer quantities,

the completability algorithm is an example of an integer LP problem. Integer LP is a subset of LP with the additional constraint $\mathbf{x} \in \mathbb{Z}$.

The first step of sorting fragments into targets is therefore to characterise the fragments present in a configuration. A list of all fragments present and the particles they contain is required, as well as whether these fragments are valid. A valid fragment is a sub-fragment of the target cluster, which implies no more than one of each particle type, no bonds between particles not present in the target and all particles in the correct orientation. Any fragments which are not valid must be discarded and the number of possible targets updated accordingly. For example if the target cluster consists of four particles, ABCD, and there are ten copies of each particle, but the cluster ABA has been identified, then it is now only possible to complete a maximum of eight targets as two A particles have been removed.

With the fragments identified, the following problem is solved. A number of sets equal to the maximum number of possible targets are defined. Each fragment may belong to one set, taking with it all of its particles. The score of a set is simply the number of particles it contains, and the objective function is the sum of all set scores. The solution is given by maximising the objective function subject to two constraints. The first is that each fragment may only appear in at most one set, and the second that each set may only contain at most one of each type of particle. The formulation of this assignment as a LP problem (like that in equation 5.5) is presented in appendix 5.A.

5.4 CONNECTIVITY

In order to investigate the impact of target geometry on addressable self-assembly, two test clusters are used. The ideal (minimum energy) geometry of each of the clusters is shown in figure 5.4. Both clusters are composed of twenty unique particles. Target A (figure 5.4 top) is a compact $5 \times 2 \times 2$ array of particles. Target B (figure 5.4 bottom) in contrast has an open, cage-like structure, and resembles a $3 \times 3 \times 3$ cube with the face-centre and body-centre particles absent. These two clusters can contrast the advantages and shortcomings of compact, densely connected structures and more open targets.

Dynamical MC simulations were conducted using the algorithm described in chapter 3. Each data point is the mean of 25 simulations at a particular set of simulation conditions. Each of the 25 simulations begins from a different starting configuration, each being a disordered monomer fluid, generated from a single high-temperature simulation.

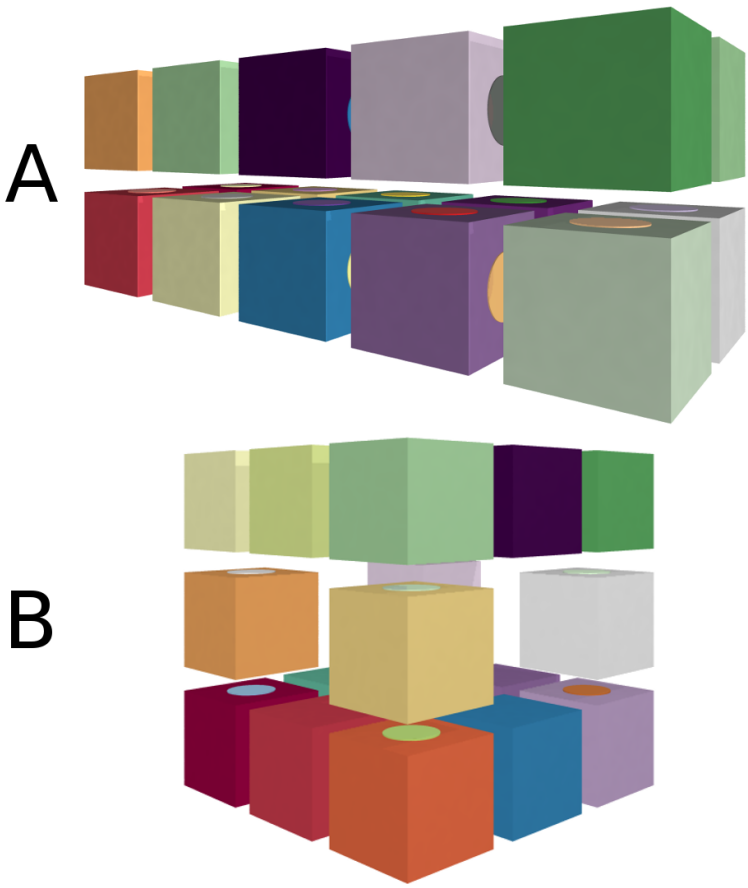

Figure 5.4: The two target clusters used in this work both consisting of twenty particles, the compact structure (target A, top), and an open cage structure (target B, bottom)

The starting configurations are then instantaneously quenched to a chosen assembly temperature at the beginning of the simulation.

In order to ensure the simulations of different structures and connectivities are as comparable as possible, in all simulations each component is present at a number density of $0.002d^{-3}$. This ensures that the chemical potential of each component is identical across all simulations. All simulations contain 80 particles, enough to make a total of 4 targets. Simulating multiple replicas of each component is necessary to observe competition.

For the first time in this thesis, patch strengths will not always be identical. In this chapter the reduced temperature T^* is defined as

$$T^* = k_\mathrm{B}T/\varepsilon\,, \tag{5.6}$$

where ε is the mean interaction strength between interacting patches ($\varepsilon_{ij} \neq 0$). As before,

the reduced temperature defines a time scale according to equation 3.19.

5.4.1 TARGET CLUSTERS

The yield and potential maximum yield of both the target clusters, as a function of temperature after a time of $t^* = 1.33 \times 10^8$ are shown in figure 5.5. In these plots the fraction of particles in complete and correct clusters (with all particles in the correct orientation) is shown by the blue squares. The pink triangles show the completability at that time, determined using the completability algorithm introduced in section 5.3. This value measures the instantaneous frustration in the system. Fragments which may not combine to form a complete target will reduce the completability. The red circles indicate the fraction of particles in valid fragments, *i.e.* fragments with at most only one of each component, in the correct orientation. Fragments of this type are therefore always sub-fragments of the target.

For target A (figure 5.5 top) optimal assembly occurs within a narrow temperature window of $T^* \approx [4.25, 4.5] \times 10^{-2}$. Above this range the cluster is thermally unstable and the system exists mainly as small aggregates and monomers. This is visible from the combination of low actual yield and high completability in figure 5.5. At temperatures below optimal this cluster suffers from severe frustration which greatly limits the extent of assembly possible. The close pairing of actual yield and completability lines shows that assembly has proceeded as far as is possible with the existing fragments, and the system is depleted of the components it needs to complete the fragments present. For the actual yield to increase, the aggregates present must first break up. This frustration is so severe that the completability rapidly approaches zero as we explore temperatures below optimal. Yields at optimal temperatures remain relatively low, approximately 50%, with the completability significantly above actual yield. This implies that assembly is slow for this structure, and that there may be steric effects, which stall assembly as fragments near completion.

The equivalent plot for target B has a markedly different character. The region of optimal yield is considerably broader than for target A, in the approximate range $T^* \approx [2.75, 3.75] \times 10^{-2}$. Yields at optimal temperatures are similar to those for target A. The completability line demonstrates that assembly is again limited by frustrated configurations. As with target A the fraction of correct fragments remains very high across all temperatures.

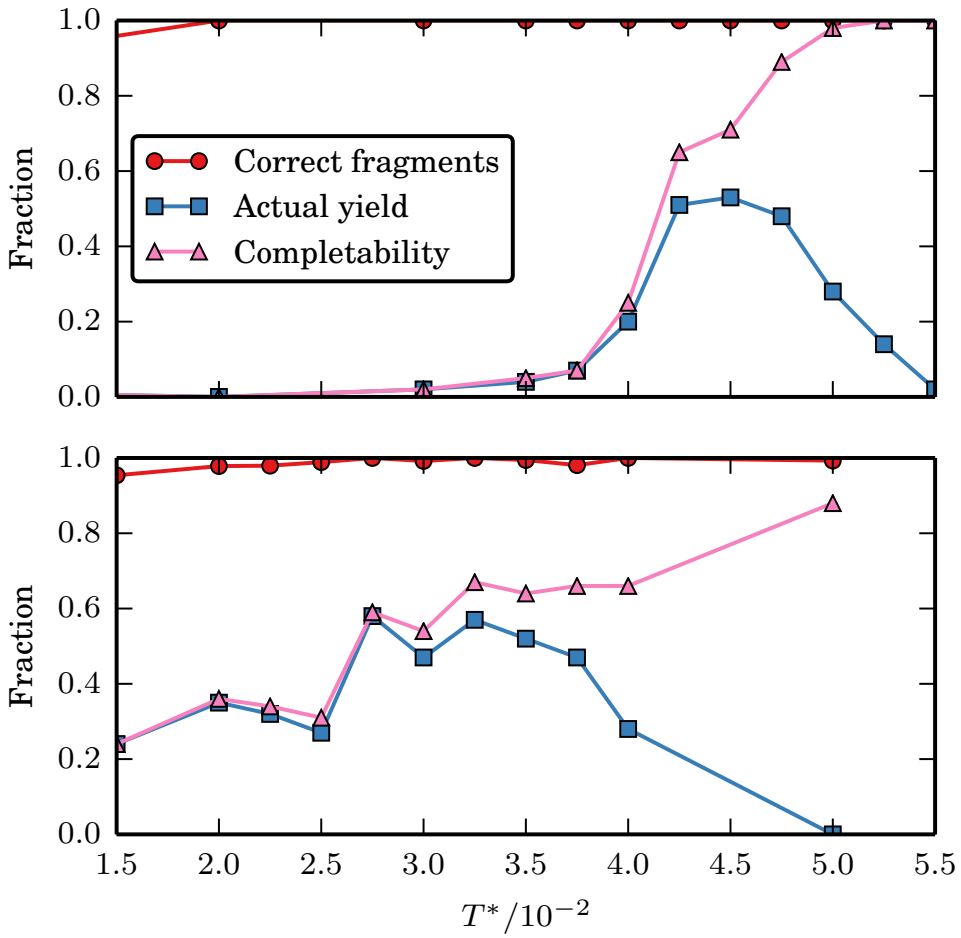

Figure 5.5: Plots of yield and completability for both of the test clusters simulated, target A (top) and target B (bottom).

5.4.2 Linear and Looped Connectivities

In the previous section the frustration caused by incompatible fragments was demonstrated to be a significant factor in limiting the self-assembly of the addressable clusters targeted. This section will explore the origin of that problem, and demonstrate that the severity of fragment competition is strongly affected by the characteristics of the bonding network in the target cluster.

The lowest possible connectivity which still results in successful assembly, is a network of bonds with no loops at all. Here, we term such connectivities as linear. If there were any fewer bonds it would not be possible to connect all particles together in the target geometry. The advantage of such an arrangement of bonds is that it makes it impossible to form fragments that are incompatible with being joined into complete targets,

as long as there are equal numbers of all components. However, with the addition of just one more link, making the network of bonds a single loop, frustrated aggregates may be created.

For example, consider a cluster composed of four components A–D and a system containing eight particles, enough to form two complete targets. The particles may be arranged into aggregates as follows, ABC, AB, CD, D. In the case where the cluster is linearly connected, where A only binds to B, B only binds to C and C to D, we may combine clusters ABC with D and AB with CD to form two complete targets. This is the only way to combine these fragments and any set of legitimate fragments of this system can always be combined. On the other hand if the cluster has a looped connectivity, where A and D also bind, there is also the possibility to combine fragments AB and D, to produce DAB, ABC and CD. This configuration is frustrated as the fragments may not be combined to create any targets without first breaking into smaller clusters.

Two new connectivities of target A were designed, one linear, and one featuring a single loop. These connectivities are shown in figure 5.6, A1 and A2, where blue spheres represent the location of particles and red lines the links between them. The yield plots for these connectivities are presented in figure 5.7.

For the linearly connected structure, the fraction of correct fragments and completability remain at 1 across the entire temperature range simulated. Achieving the maximum value of completability indicates that there is no frustration present in the system. This result therefore confirms that using a linearly connected design does indeed remove the possibility of creating a system of frustrated aggregates. It should also be noted that other kinds of kinetic traps, like misbound particles, are avoided by using the simplified potential, unlike the model with explicit patches.

As with the maximally connected cluster presented in the previous section, at high temperatures the achieved yield drops as the target becomes thermally unstable. Below this point however the yield is relatively insensitive to temperature. This is because the main limiting factor in the former case, configurations where fragments may not be combined, has been eliminated. The flat profile of the actual yield line further indicates that assembly may be slowed by the difficulty in fitting aggregates together in the correct orientation. Although particles move more slowly at low temperatures (represented in our simulations by completing fewer MC cycles), final yields reach similar levels, as at all temperatures the yield has reached a level where diffusion of particles is not the limiting factor in assembly.

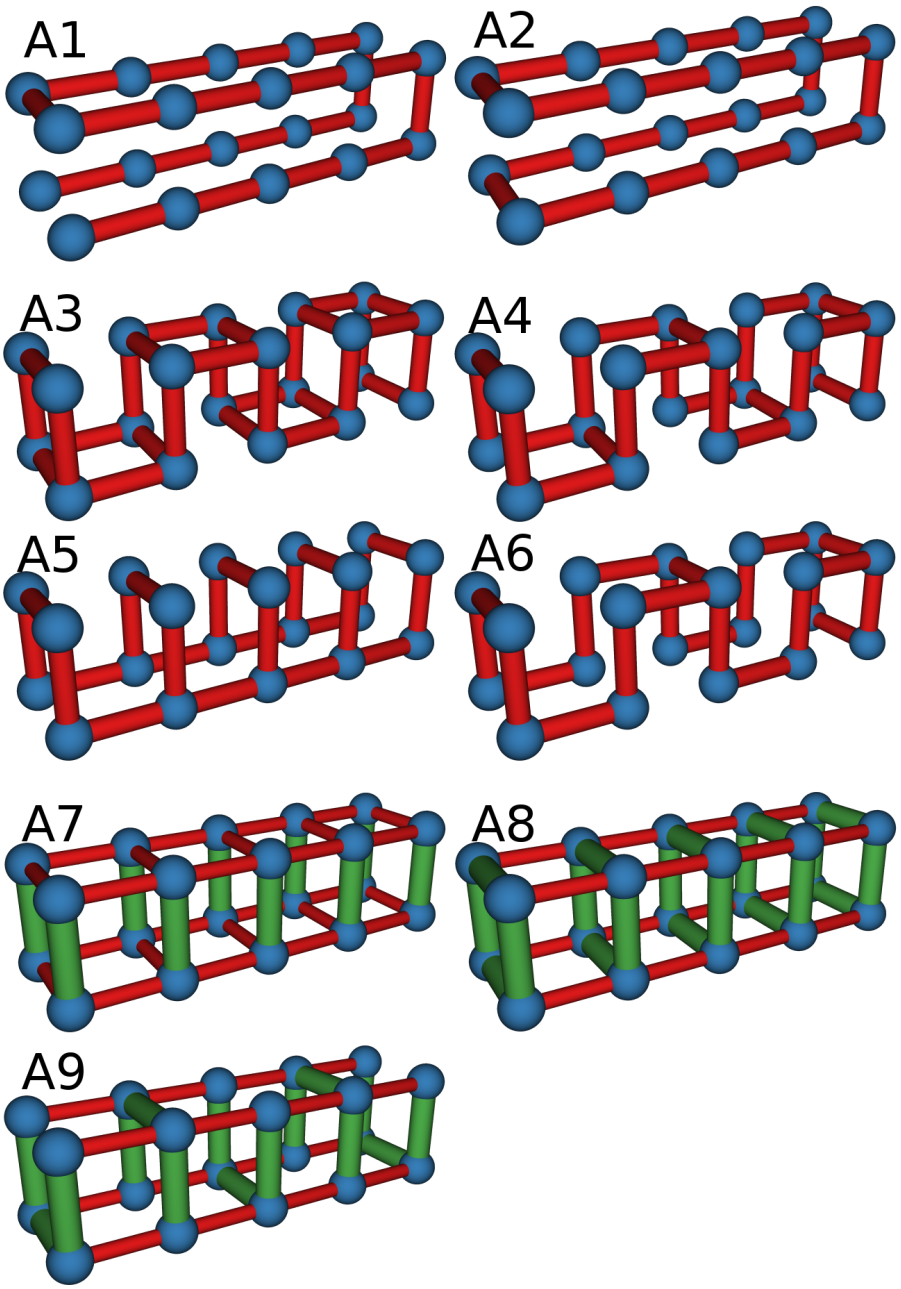

Figure 5.6: Bonding networks for variations on target A simulated in this work. Blue spheres represent the particles, and lines between them bonds. Each building block is unique, even though they are rendered identically in these diagrams. A1 and A2 are the linear and looped connectivities respectively. A3–6 were used to explore the effect of number and size of rings in the connectivity on assembly. A3 contains loops of four particles, A4 has four loops of six and one of four, A5 loops of eight and A6 has two loops of ten and one of four. Connectivities A7 and A8 contain a mixture of strong and weak bonds, strong bonds represented by thick green lines. In A7 the cluster is divided into strongly linked dimers, whereas in A8 the cluster is formed from strongly linked tetramer slices. In A9 a single link has been removed from each tetramer slice, to ensure these fragments are linearly connected.

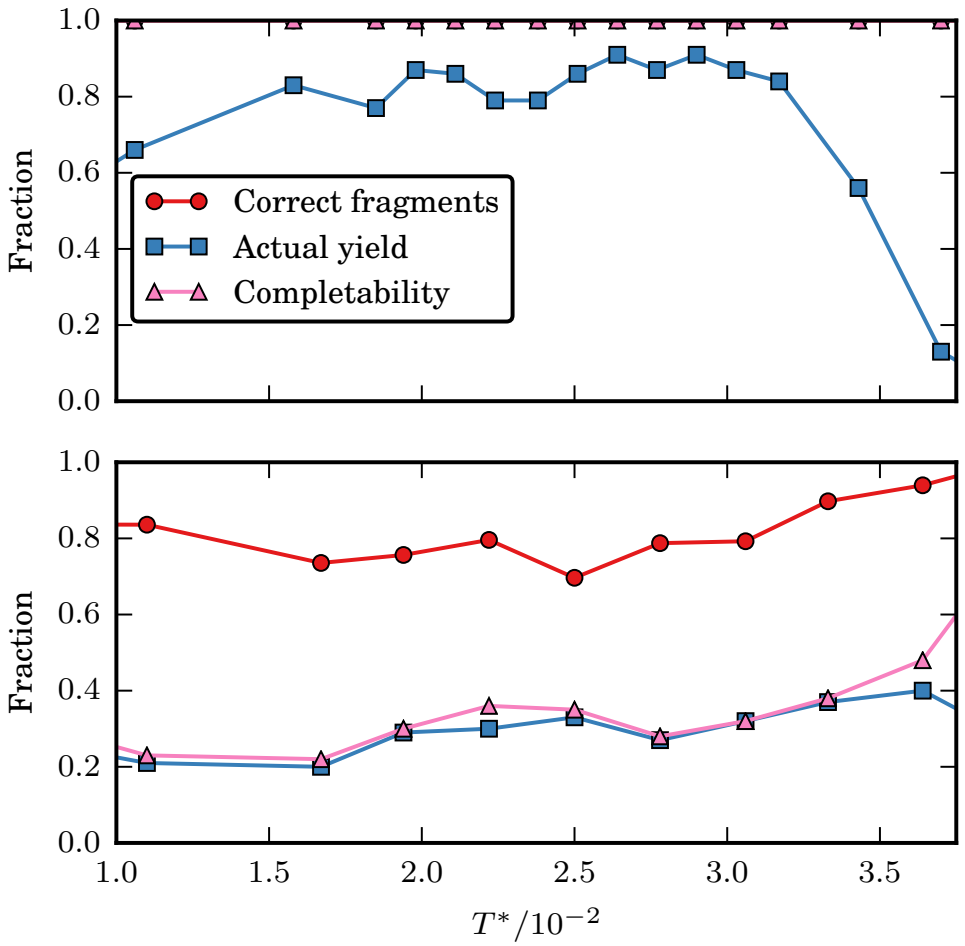

Figure 5.7: Yield and completability for the linearly connected (top) and singly looped (bottom) connectivities of target A.

While the linearly connected structure may seem successful, both in terms of its yield and lack of frustrated configurations, it is unlikely to be a viable strategy for addressable self-assembly. The lack of links in the target makes it fragile; breaking any one bond will cause the target to fall apart. In contrast, a more highly connected structure, like the fully connected compact target, requires multiple bonds to break before a building block is lost. Loops of connectivity in a target provide resilience, and reinforce correct assembly.

By adding a single bond to the linearly connected designs, singly looped structures may be created (figure 5.6 A2). Comparing the linear and looped variants of target A in figure 5.7 it is immediately clear that the addition of one link has had a very damaging effect on assembly. Whereas the linearly connected structure had no possibility of forming frustrated configurations, with a single loop of bonds incompatible fragments form

and severely limit yield. The yield of the looped connectivity never reaches the maximum yields previously seen in the fully connected and linearly connected variations. The temperature at which frustration begins to become less of a limiting factor (seen by an upward tend in the completability with temperature in figure 5.7) is also similar to the temperature at which the target becomes thermally unstable. As the completability rises the actual yield therefore falls.

The addition of a single link also introduces problems with incorrect aggregates (evidenced by the red line in figure 5.7). An image of one such structure is shown in figure 5.8. When there is a loop of connectivity, assembly is terminated when a complete cycle of links is formed. This is different from the linearly connected structures where there are "capping" particles, which only form one link and terminate assembly. It is therefore possible for the structures with looped connectivities not to be terminated, and grow without bounds if there is sufficient flexibility in the structure to avoid closing a loop. This is what has happened to the aggregate in the centre of figure 5.8. The long edges of the cluster are able to splay outwards so that the loop is not completed, but rather extra particles are incorporated. While the presence of incorrect aggregates is not the limiting factor in assembly (figure 5.7), this result does suggest that for larger structures, an over-specification of links may be beneficial as it adds rigidity to the cluster, making such outcomes less likely.

5.4.3 Intermediate Connectivities

As well as the extremes of a minimal number of links (the linear case explored in section 5.4.2) and the fully connected bond networks (section 5.4.1) there also exist connectivities which sit in between. In this section we will explore a series of intermediate connectivities of target A, illustrated in figure 5.6 A3–A6.

Section 5.4.2 demonstrated how a linearly connected structure rules out the possibility of frustrated configurations where the aggregates present may not combine to form the target. It also showed that the addition of one more bond, to make a singly looped connectivity, causes such configurations to severely limit assembly. However, it is known from figure 5.5 that increasing the connectivity further is eventually beneficial, as the fully connected target A performed better than A2 with only one loop of bonds. The question that remains is how the number of links and also the size of loops within the connectivity affect assembly.

The bonding networks of the intermediate connectivities simulated are shown in fig-

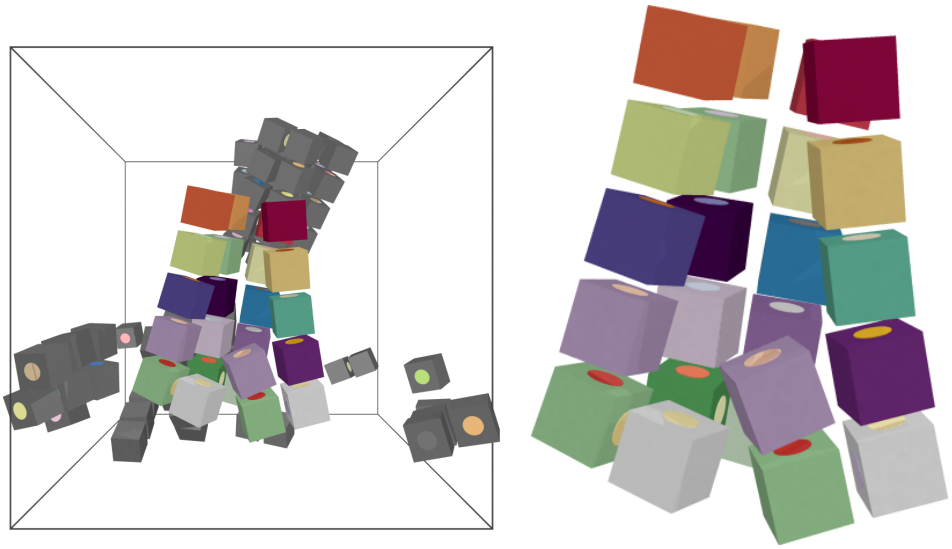

Figure 5.8: Left, A simulation snapshot of target A with a single loop of bonds connecting particles in the target geometry. An incorrect aggregate is highlighted in the centre of the simulation box, where the flexibility of the structure has allowed the aggregate to grow rather than closing the loop an terminating assembly. A larger view of the isolated aggregate is presented on the right.

ure 5.6 A3–A6. The most connected structure is A3, consisting of nine edge-sharing rings of four particles. Connectivity A4 has four edge sharing rings of six particles with one ring of four in order to incorporate all particles. A5 contains three edge-sharing rings of eight particles. Finally the connectivity with least links, A6, has two edge-sharing rings of ten particles. As with the connectivity containing rings of six particles, a ring of four is added to A6 in order to make up to the twenty particles in the structure.

The yield plots for the connectivities introduced above show a number of trends as the size of loops in the structure is varied. The plot for the least connected structure A6 (figure 5.9) resembles that of target A with a single loop (figure 5.7 bottom). In both cases assembly is limited by frustrated configurations. There are however signs of improvement. The completability line does rise more than the same line for the singly looped connectivity, although actual yields lag behind. This is possibly due to the lack of energetic return from binding with so few links. Instances of incorrect structures (caused by the floppiness of the structure with only one loop) are reduced.

When the connectivity is further increased, the situation improves further, which can been seen in the plots connectivities A5 and A4 (figure 5.9). Incorrectly bound structures are almost completely eliminated, and the number of frustrated configurations continues

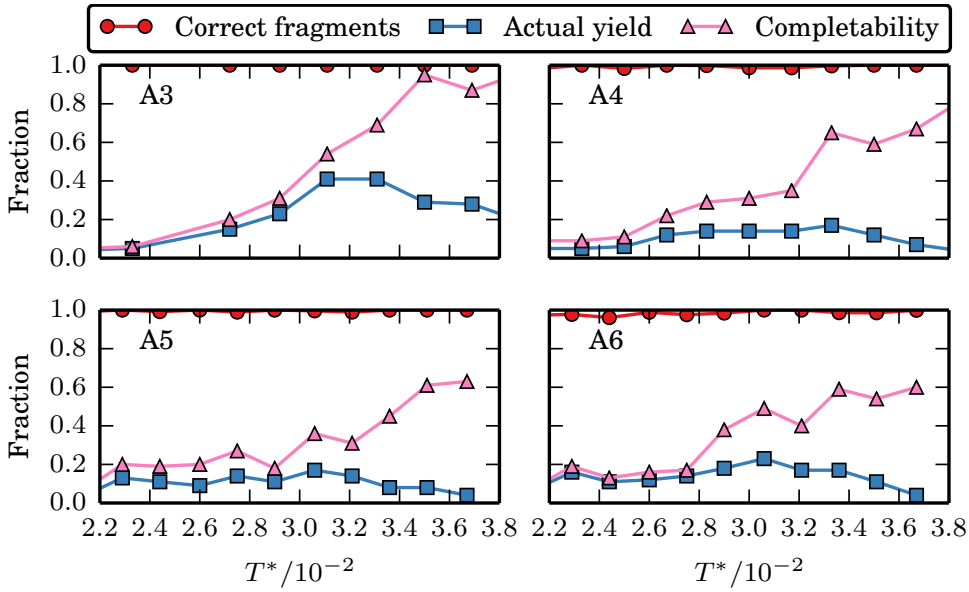

Figure 5.9: Yield curves as a function of temperature for the target A with variety of connectivities. The connectivities shown are A3, A4, A5 and A6 using the convention introduced in figure 5.6.

to fall. However, assembly can still not be considered successful, as actual yields remain low at temperatures where the system is not frustrated. The most connected structure considered here, A3, consisting of rings of four particles begins to approach what may be considered a successful design. The completability line has now risen sufficiently in the temperature range where the target is thermally stable to allow for moderate actual yields of around 40%. This yield is comparable, although still significantly worse, than the fully connected case.

The series of connectivities presented in this section have moved from the minimal case (a linearly connected cluster) back towards the maximally connected target A. While the linearly connected cluster prevented the formation of configurations with overlapping sets of particle in clusters, it is unlikely to be a viable assembly strategy. The maximally connected case, on the other hand, while suffering from frustration, shows good assembly within a narrow temperature window. Intermediate cases with a small number of large rings have the worst features of both limits. The structure is often unstable and floppy, due to a lack of links, but yields are also strongly limited by frustration. It is worth noting that the actual extent of floppiness within a cluster will depend on the nature of the interaction potential and building blocks. For example, in the patchy cube model, if particles are touching in the energetic ground state there would be less flexibility as the

motion of any building block is restricted by its confinement by neighbouring particles.

5.5 Heterogeneous Bond Energies

5.5.1 Intuitive Placement of Strong Bonds

As well as switching on and off interactions between patches, the model also gives control over the relative strength of any interaction between a pair of patches. This is achieved by adjusting values of ε_{ij} in the interaction matrix. This may be seen as a coarse representation of changing the sequence of bases and length of binding sequences in ss-DNA based interactions. Changing the sequence of the binding strands affects the free energy of binding. With careful consideration it is therefore possible in experimental systems to selectively tune the binding energy of particular attractive sites.[89]

With the structures of moderate size considered here, it is possible to intuitively place strong bonds selectively, to encourage smaller fragments of the target cluster early into the assembly. This approach will encourage a (partially) hierarchical assembly mechanism like those discussed in the previous chapter (section 4.3.2). The advantage of a hierarchical approach is that if small fragments of the target are reliably assembled first, the size of the assembly problem has been effectively reduced. The fragments in the hierarchical pathway, if they form a large population rapidly, may be considered the new, larger building blocks of the target. A multi-step approach may not be necessary for small targets, but as the size of clusters grow it could prove advantageous.

Linearly connected structures eliminate the possibility of frustrated configurations; it is the introduction of loops to the bonding network which allows for incompatible fragments to form. A hierarchical approach may be expected to benefit from the sub-fragments of the target being linearly connected. This will ensure that frustration does not occur at this early stage. As these sub-fragments will be linked by stronger than average bonds, they should form rapidly and they will be less likely to break apart. These fragments may therefore be particularly susceptible to frustration if not linearly connected, as they will be less able to mend incompatible configurations.

The first design featuring intuitively placed strong links presented here is a modification of the fully connected target A. The introduction of ten strong bonds on perpendicular to the long edge divides the cluster into ten strongly bound dimers. Weak bonds account for the remaining connections. A schematic of this connectivity is shown in figure 5.6 A7 where thick green lines indicate the strong links and thin red lines weak links. The

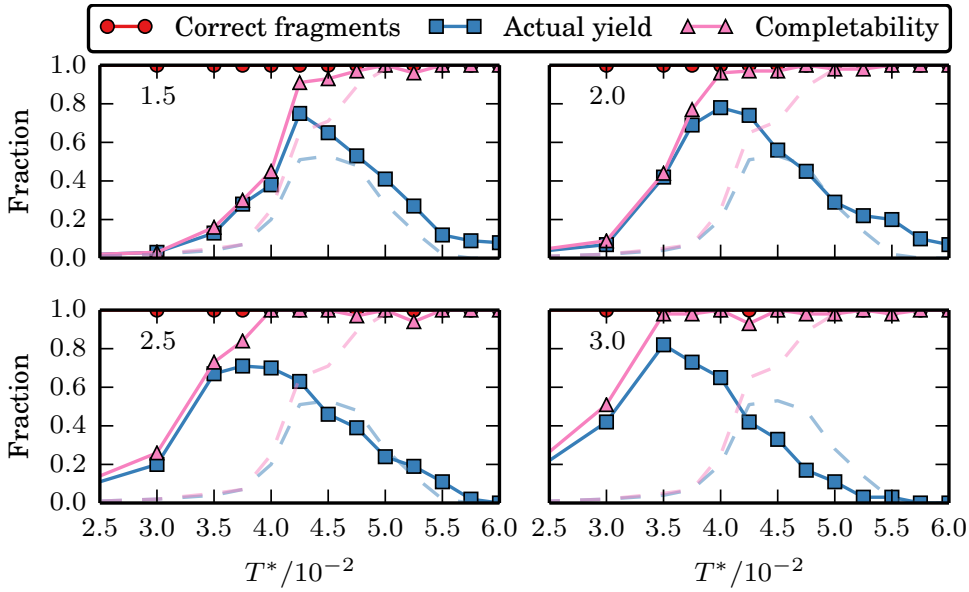

Figure 5.10: Yield curves for target A with strong bonds dividing the cluster into strongly linked dimers. The ratio of strong to weak bond strengths varies across the plots with the values 1.5, 2.0, 2.5, 3.0. The faint dashed lines show corresponding trends for the fully connected target A (figure 5.5).

strengths of the bonds are set so that the total binding energy of the cluster remains the same as in the case with homogeneous bond strengths. If successful, this approach should lead to the rapid and reliable assembly of dimers. This effectively reduces the size of the problem from a twenty-component assembly to one featuring only ten. The yield and completability for this system is presented are figure 5.10 with a range of disparity between the strength of strong and weak bonds. The ratios of strengths between strong and weak bonds shown are 1.5, 2.0, 2.5, 3.0.

The introduction of the strong bonds has a dramatic effect on the yields obtained for target A. When the trends in yield are compared with the same cluster featuring homogeneous bonds (visible in figure 5.10 as faint dashed lines) it is immediately obvious that the change in bond strengths leads to higher yields, and a wider temperature window for assembly. The largest improvement in yield comes at lower temperatures, where previously maximum yield was limited by incompatible fragments. The optimal yield rises somewhat as the ratio of bond strengths increases, reaching a maximum of around 80%. Along with the improvement in yield obtained, there is a large improvement in completability, which continue to rise as the ratio of bond strengths increases. This suggests that the difference in binding energies has indeed promoted the formation of dimers which, being linearly connected, may always combine to form the target. Accordingly, the proportion

of frustrated configurations seen strongly falls.

There is also initially an improvement in yield towards high temperatures which disappears as the disparity in bond strengths is increased. Eventually yields at high temperatures fall below those of the homogeneous system and there is an overall shift towards a lower optimal temperature of assembly. The loss in yield may be attributed to the binding between weak patches not being strong enough to ensure that the transient intermediates are bound strongly enough to stay together until encountering a complementary fragment.

Target A may also be divided into five tetramers connected by strong links. Each tetramer is square in shape, being a slice of the cluster along its long axis (figure 5.6 A8). The yields for this distribution of strong and weak bonds are presented in figure 5.11, with the same range of bond strength ratios as when the cluster was divided into dimers.

Dividing the cluster into strongly bound tetramers does give some improvement in the temperature range of feasible assembly, particularly at a bond strength ratio of 2.0. However, the yield at optimal temperature is at best similar to the uniform system, and often worse. This is in contrast to when the cluster was divided into dimers where there was a large improvement in yield. Most notably, unlike the dimer design, the completability does not rise significantly, indicating that there are still many frustrated configurations forming. This may be due to the tetramer fragments not being linearly connected. As a result it is possible for frustration to arise in the first stage of assembly (building a set of tetramers from monomers) which is not the case for dimers.

The difference between performance of assembly when target A is divided into dimers and tetramers suggests that the success of a hierarchical approach critically depends on the chosen fragments. Further confirmation of this may be obtained by by removing a single link from each tetramer slice in A8 to make each slice a linear fragments. The resulting connectivity A9, displayed in figure 5.6, contains tetramer fragments with the same shape as those in A8. However, they are now connected with three links arranged in a "U" shape, rather than a square of four links. As with the previous two arrangements of strong links, the bond energies are scaled such that the total energy of the cluster remains the same.

The yield curves for connectivity A9 are shown in the right hand column of figure 5.11. For the two smaller bond ratios (1.5 and 2.0) there is little improvement in actual yield, although the completability rises. As the ratio of bond energies increases the situation improves and the amount of frustration at the optimal temperature falls. At the maximum bond ratio simulated of 3.0 the maximum yield reaches 80%, a similar

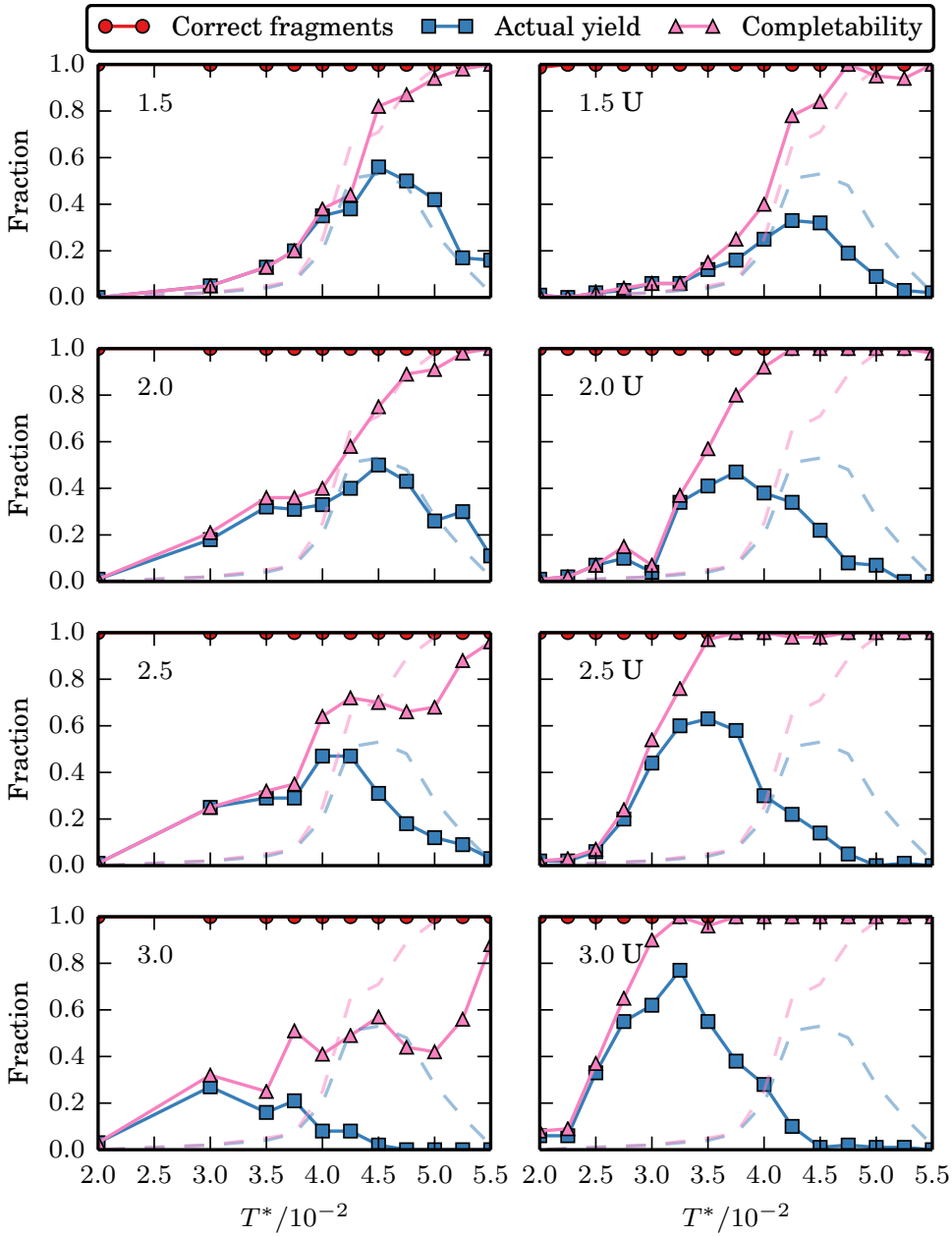

Figure 5.11: Yield as a function of temperature for target A, divided into square tetramer slices by strong links. The ratio of strong to weak links in each plot from top to bottom is 1.5, 2.0, 2.5, 3.0. The plots in the left column show the results when the tetramer slices are connected by a loop of bonds, A8 in figure 5.6. The plots in the right column show the results for model A9 where the tetramer fragments are linearly connected by three bonds arranged in a "U" shape. As before, the faint dashed lines show corresponding trends for the fully connected target A (figure 5.5).

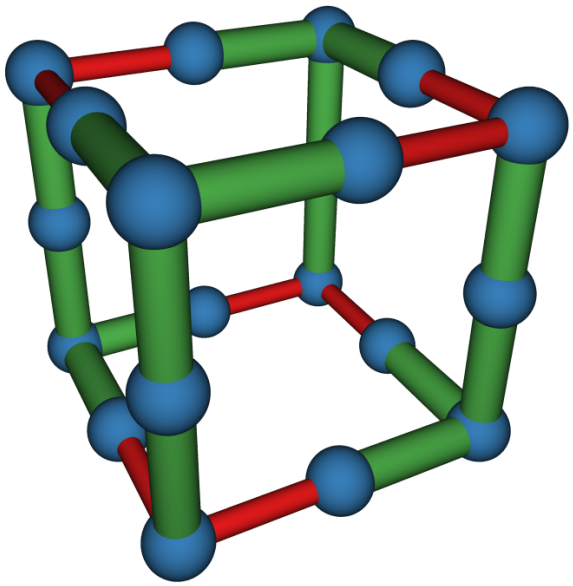

Figure 5.12: The connectivity of target B with intuitively placed strong links. Green lines indicate strong links, and red lines indicate weak links.

performance to when the target is divided into dimers (figure 5.10). The difference in performance after removing a link from the tetramer sub-fragments indicates that it is important for fragments to be linearly connected. This ensures incompatible fragments are less likely to occur during the early stages of assembly. The ability to reliably build the sub-fragments appears to be more important than reducing the size of the final step, as the more successful dimer approach requires assembly from ten components, whereas the tetramer approach uses half that number in the final step.

A heterogeneous connectivity for target B that divides the cluster into four pentamers was also simulated. A schematic of this connectivity is shown in figure 5.12, as before thick green lines indicate the strong links. The pentamer fragments consists of one of the four vertical edges of the cluster, along with the two particles adjacent to either the top or bottom particle of that vertical edge. Each is a linearly connected fragment of the target, despite the branched network of bonds.

Yield curves for target B divided into strongly linked pentamers are presented in figure 5.13. Bond strength ratios of 1.5, 2.0, 2.5 and 3.0 were simulated. As the disparity between bond strengths is increased the temperature of optimal yield is reduced, in agreement with the previously described arrangements of strong links. Also in line with the other successful designs featuring strong links, selectively placed strong links are seen to reduce the occurrence of frustrated configurations. This is visible in figure 5.13 as an

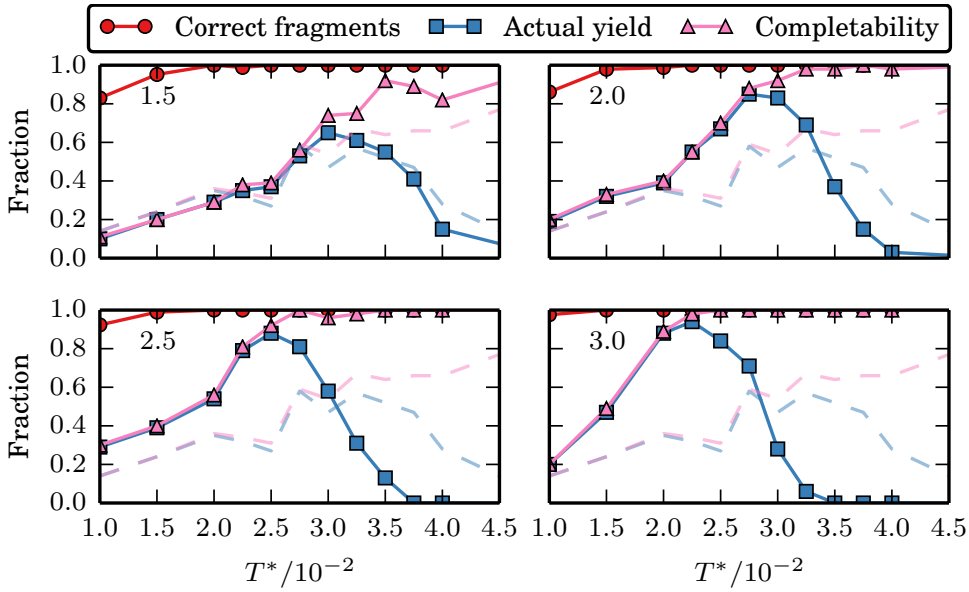

Figure 5.13: Yields as a function of temperature for the open, cage like structure with strong links dividing the cluster into pentamers. The ratio of strong to weak bond strengths shown are 1.5, 2.0, 2.5 and 3.0. The faint dashed lines show corresponding trends for the fully connected target B (figure 5.5).

increase in the completability relative to that of the homogeneous design. Frustrated configurations limited the yields of target B across a very wide range of temperature. As the bond strength disparity is increased there is a large increase in both yield and completability. At optimal temperature the actual yield exceeds 90%, the highest yield observed for any strategy of either target.

5.5.2 Random Placement of Strong Bonds

In order to ensure that the intuitive placement of strong links was indeed beneficial to the assembly, simulations of sets of particles where a given number of strong links had been placed randomly in the target cluster were also conducted. Results were averaged over a set of 25 simulations, each using a unique, random distribution of strong links.

The yield curves for target A with randomly placed strong links are presented in figure 5.14. For each plot in the figure the strong links interact with double the energy of weak links. The number of strong links in the displayed plots are 6, 12, 18, and 24 out of a total of 36. As with all previous examples featuring strong and weak bonds, the total energy of the cluster remains constant and equal to that of the homogeneous case.

Figure 5.14 shows that the introduction of some strong links, even in random loca-

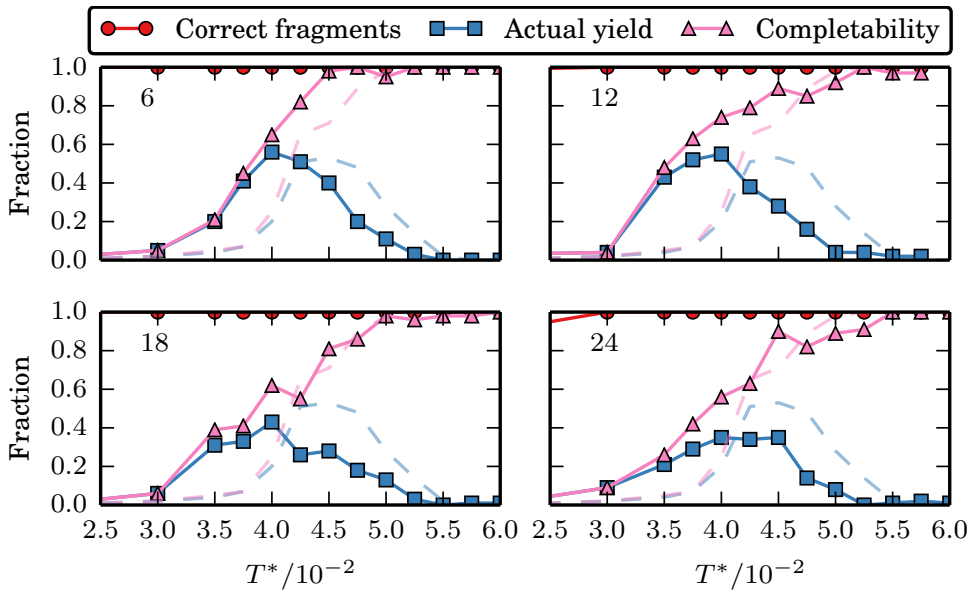

Figure 5.14: Yield plots as a function of temperature for target A with strong links randomly placed within the structure. The number of strong links in the plots shown are 6, 12, 18 and 24 out of a total of 36.

tions, can improve assembly. With 6 and 12 strong links there is a small improvement in the maximum yield, and a slight broadening of the temperature window of assembly. The prevalence of frustrated configurations falls and the optimal temperature for assembly falls, similarly to the designs with intuitively placed strong links. However, there is not a significant improvement in yield, unlike the most successful intuitive designs.

As the number of strong links is increased further, performance is worse relative to homogeneous bond strengths. In particular with 18 and 24 strong links, maximum yields fall below that of the homogeneous case. The number of frustrated configurations at low temperatures also begins to grow. With a large number of strong links, more links are expected to form irreversibly, and there are also likely to be more loops of strongly connected particles within the cluster. The combination of these features is likely to result in more frustrated configurations which hinder assembly. As the number of strong bonds approaches the maximum number of 36, the profile of the yield curve returns to closely resemble that of the homogeneous system because having all links strong is equivalent to the (rescaled) homogeneous case already considered.

Considering both the connectivities with strong bonds placed intuitively and randomly, it is clear that the placement of the links is important to the success or failure of a design. In particular, intuitive designs—intended to promote a semi-hierarchical as-

sembly pathway—perform far better than designs where strong links have been placed randomly. When strong links are placed it is important to not create loops of strong links. The rapid association of strong links will in this case likely end up in a system featuring many incompatible aggregates. If the strongly bound fragments are linearly connected, however, this is not possible.

5.5.3 Random Bond Energies

The previous section showed that strategic placement of strong links can have a positive effect on the assembly of discrete clusters. Providing that the strongly linked fragments are suitable, there is an improvement in both maximum yield at optimal conditions, and the reliability of assembly across a range of temperatures. If strong links are placed randomly there may still be a limited improvement in assembly. An alternative way to introduce heterogeneity into the strength of links is to have a stochastic, continuous distribution of bond energies. In the case of DNA bricks, it was found that using random ss-DNA binding sequences, and hence uncontrolled distribution of bond energies, produced yields no worse than a specially designed set of sequences.[87] The designed sequences aimed to reduce the stability of secondary structures and reduce sequence symmetry. Jacobs *et al.* found, in their model, that a random distribution of bond strengths reduced the nucleation barrier of addressable clusters and helped the system avoid kinetic traps, rationalising the experimental observation.[89] In this section the effect of randomly chosen bond strengths will be tested briefly.

For both of the targets, variants with random bond energies based on the fully connected connectivities were generated. For each link a binding energy was picked from a Gaussian distribution with mean ε and a given standard deviation. The results presented here are the averages of 25 individual simulations, where each simulation uses a distinct, random distribution of bond strengths. Within any given simulation, each of the four copies of the target target has the same pattern of bond strengths.

Yield plots for target A with random bond energies picked with standard deviations of 0.1ε, 0.2ε and 0.3ε, are presented in figure 5.15. Initially the stochastic bond energies seem to make little difference, and are perhaps detrimental to assembly. However, as the size of the standard deviation of bond energies is increased (0.2ε and 0.3ε) there begins to be some improvement of the kind previously observed with strong and weak links. The number of frustrated configurations at low temperatures falls and there is a corresponding increase in yield. This is also met with a decrease in yield at high temperatures. Overall

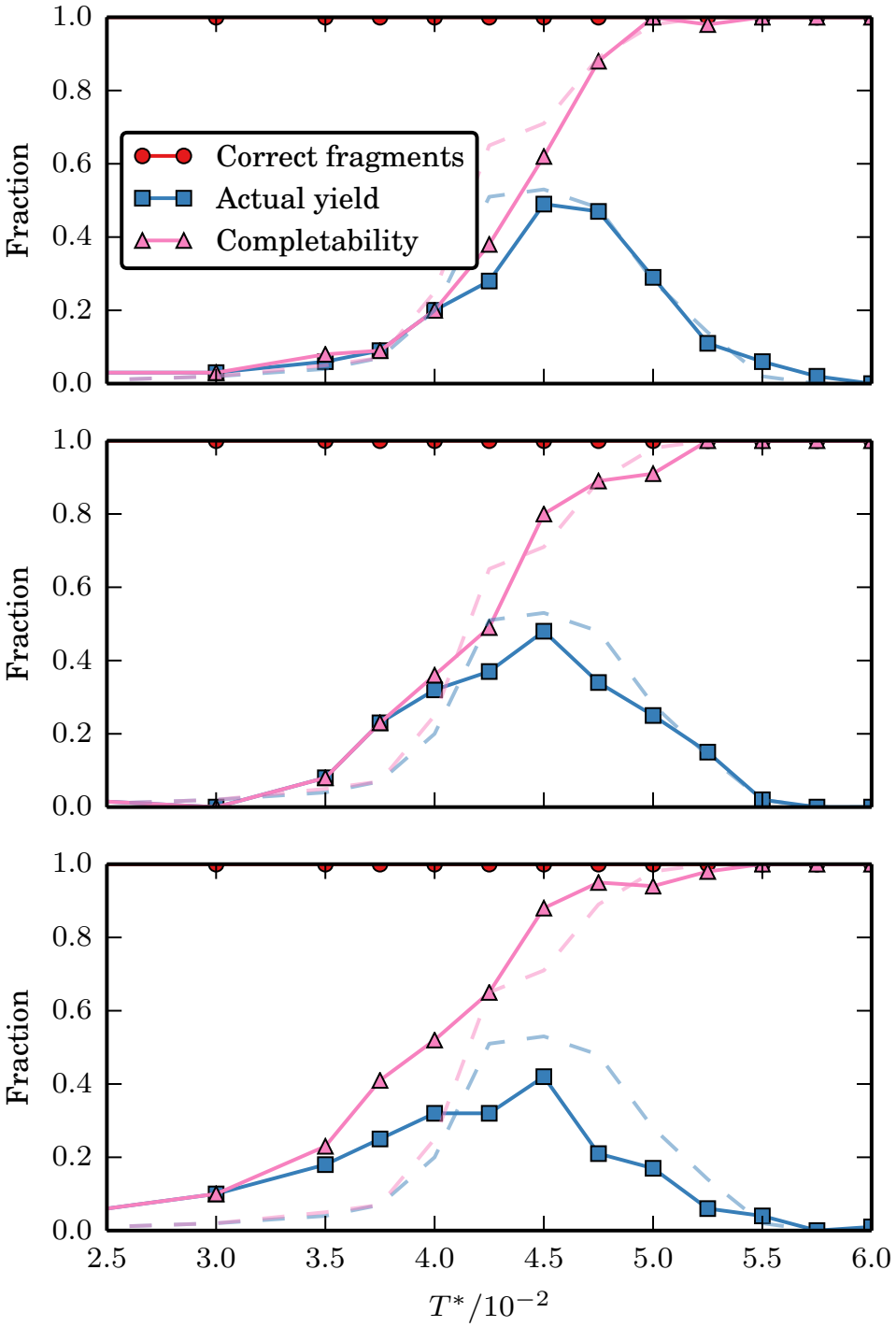

Figure 5.15: Yield plots for target A with all binding energies chosen randomly from a Gaussian distribution about ε. From top to bottom the standard deviation of the Gaussian is 0.1ε, 0.2ε and 0.3ε. The faint dashed lines show the corresponding yield and completability for the homogeneous target.

this approach seems to provide only a small advantage. The temperature window of assembly is a little wider, but the maximum yields fall slightly.

5.6 DISCUSSION

Previous work has explored optimising interactions between building blocks[96–98] in addressable clusters. However, less research has been conducted into the role played by connectivity in determining the success of assembly. Coordination number has been investigated by Jacobs *et al.* in the context of a maximally connected addressable cluster.[89] In this chapter the network of links in a target was not considered immutable, but was modified while maintaining the geometry and overall binding energy of the target. It was found that if a cluster is connected by a linear network of links it is not possible for incompatible fragments to form, which can otherwise hinder assembly. However, this approach is unlikely to be viable as it results in fragile targets which rely on all links being formed simultaneously.

Bonding networks with low connectivity perform poorly as not only can incompatible fragments form, but floppy structures also lead to large incorrect aggregates. The latter of these kinetic traps occurs when the flexibility of a structure allows a loop of connectivity to not be closed, but rather to grow unbounded. This effect is allowed by any addressable bonding network with loops, but its prevalence will depend on the nature of the interactions and building blocks themselves, which determine how flexible a cluster is. This effect may also only be observed when multiple copies of each building block are present, which is often neglected in other simulation studies. The simulations presented here are therefore able to capture this type of error, while many others have not.

Similarly to Jacobs *et al.*[83,89] we find that a heterogeneous bond strength distribution enhances assembly. This is in contrast to some earlier studies which favoured homogeneous binding energies as they enhanced stability of target[96] relative to competing structures and it was proposed they may reduce errors during growth.[98] The success of heterogeneous binding energies in the patchy cube model, however, appears to be for different reasons than in the model of Jacobs *et al.*. In our case semi-hierarchical pathways to assembly are promoted, which help avoid frustrated configurations. Because of this the most effective approach requires an informed design where strongly bound fragments are linearly connected. For the relatively small targets studied here, it was sufficient to design the bond networks by hand. For much larger clusters such as those formed by DNA bricks

this may not be practical. However, given the importance of fragments being linearly connected, it is feasible that an algorithm may be developed to design good distributions of strong links based on the geometry of a cluster and a set of design rules.

In the case of Jacobs *et al.*[83,89] and the assembly of large DNA brick clusters,[86] a random distribution of bond strengths was effective. There, the heterogeneous bond energies enhance the late nucleation barrier. In our simulations, random placement of strong bonds or picking each bond strength from a distribution does not show any great benefit. This is an issue of size. Our targets are much smaller and so do not see the same nucleation kinetics as the DNA brick clusters featuring hundreds of building blocks. This work is mainly concerned with how competition may be limited and how to prevent frustrated configurations, which requires multiple copies of our building blocks. In contrast the work of Jacobs *et al.* aimed to characterise assembly by calculating the free energy profiles of assembly for a single replica. While stochastic bond energies may be beneficial to nucleation of an addressable cluster, this may not have a strong positive impact on the occurrence of frustrated configurations.

The completability metric has been useful in interpreting the state of a system at a given simulation time. It helps inform why assembly may have slowed or stalled, as it can distinguish configurations with incompatible fragments from assembly that is slow because of diffusion or thermal instability of the target. In the former case, for assembly to proceed further the fragments present first need to break up to produce compatible fragments. Alternatively, the building blocks may be modified to avoid such configurations on the pathway to assembly. The completability approach to assessing self-assembly may be helpful in other theoretical models to identify this type of kinetic trap and find how to deal with it.

5.7 CONCLUSION

This chapter has explored in depth the frustration caused by incompatible fragments of addressable targets discovered in chapter 4. The simulations have revealed the origin of the frustration problem in the connectivity of addressable clusters and suggested approaches to control the competition. The variant of the patchy model used here employs a more coarse representation of interactions than the one introduced in chapter 3. The modification was made to avoid any specific multi-patch arrangement leading to kinetic traps arising from the incorrect pairwise binding of building blocks. This allows the sim-

ulations presented here to better probe the effect of connectivity on assembly. In a real system these effects can not be ignored and all sources of error are combined.

It has been assumed here that interactions may be made perfectly addressable with no cross interactions, which have previously been considered in chapter 4. However, in practice this is never the case and care must be take to ensure that interactions are specific enough. For example, when using ss-DNA to create addressable interactions it must be ensured that the choice of sequences omits any significant incidental interaction. In chapter 6 the nature of the addressable interactions themselves will be explored. By returning to the model with multiple patches per face, it will be possible to optimise the patch patterns to create the best addressable interfaces subject to a limited number of specifically interacting patches.

APPENDIX 5.A FRAGMENT COMPLETION AS AN ASSIGNMENT PROBLEM

For an addressable cluster of any size, the fragments may be represented as a binary string. For instance, in an eight-component cluster with particles A to F, the fragment ABDF would be $(1, 1, 0, 1, 0, 1, 0, 0)$. The sum of the elements of these strings give an indication of how complete a fragment is, with only a correct target achieving the maximum value.

Here we consider a system of N_r copies of each N_t types of building blocks, divided into N_f independent fragments. Note that N_r is also the maximum number of targets that can be created simultaneously. Similarly to the fragment strings, the assignment of fragments to sets (combinations of fragments in our proposed solution) may be expressed with a binary vector of length N_f . For example in a system with four fragments the set vector $(0, 1, 0, 1)$ would imply that fragments 2 and 4 belong to the set but that 1 and 3 do not.

The assignment of the N_f fragments into sets may therefore be represented by a concatenation of all the N_r individual set vectors. There are N_r set vectors, as this is the maximum number of targets possible. The assignment vector $\mathbf{x}$ therefore has the follow-

ing form,

$$\mathbf{x} = \begin{pmatrix} \left.\begin{pmatrix}\vdots\end{pmatrix}\right\} & \mathbf{s}_1 \\ \left.\begin{pmatrix}\vdots\end{pmatrix}\right\} & \mathbf{s}_2 \\ \vdots & \\ \left.\begin{pmatrix}\vdots\end{pmatrix}\right\} & \mathbf{s}_{N_\mathrm{r}} \end{pmatrix}, \tag{5.7}$$

where $\mathbf{s}_1$ is the first set vector. $\mathbf{x}$ has length $N_\mathrm{f}N_\mathrm{r}$.

We can collect the scores (sum of the binary strings) of the N_f fragments in a vector,

$$\mathbf{n} = \begin{pmatrix} n_1 \\ n_2 \\ \vdots \\ n_{N_\mathrm{f}} \end{pmatrix}, \tag{5.8}$$

where n_1 is the score of the first fragment. Let us also define a vector $\mathbf{c}$ as N_r repeats of $\mathbf{n}$,

$$\mathbf{c} = \begin{pmatrix} \mathbf{n} \\ \mathbf{n} \\ \vdots \\ \mathbf{n} \end{pmatrix}, \tag{5.9}$$

which has length $N_\mathrm{f}N_\mathrm{r}$, the same as $\mathbf{x}$.

The vectors $\mathbf{c}$ and $\mathbf{x}$ form our objective function $\mathbf{c}^\top\mathbf{x}$, which is equal to the total number of particles placed into sets. The maximum of this function corresponds to all particles being placed into sets. To find the optimal assignment of fragments to sets, which contains the most targets, $\mathbf{c}^\top\mathbf{x}$ should be maximised subject to two constraints:

1. No fragment appears more than once in $\mathbf{x}$.

2. No type of building block appears more than once in any set.

In order to construct the constraint matrix $\mathbf{A}$, we first introduce the matrix $\tilde{\mathbf{A}}$ with dimensions $N_\mathrm{t} \times N_\mathrm{f}$. The elements $\tilde{A}_{ij}$ of $\tilde{\mathbf{A}}$ are 1 if fragment j contains building block

type i, and 0 otherwise. The constraint matrix is then given by

$$\mathbf{A} = \begin{pmatrix} \tilde{\mathbf{A}}_{1,1} & & & \\ & \tilde{\mathbf{A}}_{2,2} & & \\ & & \ddots & \\ & & & \tilde{\mathbf{A}}_{N_\mathrm{r},N_\mathrm{r}} \\ \mathbf{I} & \mathbf{I} & \cdots & \mathbf{I} \end{pmatrix}, \tag{5.10}$$

where $\mathbf{I}$ is the $N_\mathrm{f} \times N_\mathrm{f}$ identity matrix.

The constraint matrix $\mathbf{A}$ has dimensions $(N_\mathrm{r}N_\mathrm{t} + N_\mathrm{f}) \times N_\mathrm{f}N_\mathrm{r}$. The product $\mathbf{Ax}$ therefore has $N_\mathrm{r}N_\mathrm{t} + N_\mathrm{f}$ elements. The first $N_\mathrm{r}N_\mathrm{t}$ elements give the number of building blocks of each type, in each set, starting with the number of A blocks in set 1, the number of A blocks in set 2, *etc.*, then the number of B blocks in set 1, and so on. The remaining N_f elements give the number of times each fragment has been assigned to any set in the order fragment 1, fragment 2, *etc.*.

The two constraints are fulfilled so long as every element of $\mathbf{Ax}$ is less than or equal to 1. Considering the first inequality in equation 5.5 it follows that the constraint vector $\mathbf{b}$ is simply a vector of length $(N_\mathrm{r}N_\mathrm{t}+N_\mathrm{f})$ where every element is equal to 1. We now have the definitions of the vectors and matrices required by equation 5.5 ($\mathbf{c}$, $\mathbf{x}$, $\mathbf{b}$ and $\mathbf{A}$) in order to optimise the objective function. Maximisation of this function will sort fragments into the sets which complete as many of the target structure as possible. Standard libraries exist to define and solve LP problems. In this work the GNU Linear Programming Kit[143] was used with the PuLP frontend[144]

As mentioned in section 5.3 if an incompatible fragment arises, using more than one of a particular type of building block, then the total number of targets than can be formed is now less than N_r . In this case, the value of N_r must be effectively reduced to the new maximum number of possible targets. This in turn reduces the number of sets to be created in the vector $\mathbf{x}$.

When it is not possible to sort the system into a full contingent of completed targets the objective function may not have a single maximum. A set of degenerate maxima may exist where one contains the largest number of completed targets possible, and the others have many large yet incomplete sets. To ensure that the maximum number of targets has been found we use the following procedure. If a full contingent of targets is not found an attempt is made to build one more than was found repeatedly. When it is not possible to build one more than previously then the previous number of targets found is the largest

possible. This may be expressed algorithmically as,

N_{r} = maximum
maximise objective
if $\mathbf{c}^{\top}\mathbf{x} = N_{\mathrm{r}}$ **then**
  finish
else
  $N_{\mathrm{r}} = \mathbf{c}^{\top}\mathbf{x}$
  while $\mathbf{c}^{\top}\mathbf{x} = N_{\mathrm{r}}$ **do**
    $N_{\mathrm{r}} = \mathbf{c}^{\top}\mathbf{x} + 1$
    maximise objective
  end while
  finish
end if

CHAPTER 6

DESIGNING OPTIMAL INTERFACES FOR ADDRESSABLE SELF-ASSEMBLY

6.1 INTRODUCTION

To build an addressable structure it is necessary to have control over the interactions between building blocks. Specifically, building blocks must interact strongly with their intended neighbours relative to other components. This is a necessary requirement for building blocks to adopt their intended position within an addressable structure. In the preceding chapters, perfectly exclusive interactions between building blocks have been achieved through the patch interaction matrix, using as many distinct patch types as were required. However, this is a simplification, as some amount of unintended interaction is inevitable in practice. In this chapter, a design procedure will be used to generate optimally distinct interfaces for addressable assembly.

Ss-DNA is widely used to mediate self-assembly, whether it is grafted to the surface of colloidal particles[20,64,145] or the building blocks themselves are folded strands (as in the case of DNA tiles[85] and bricks[86,87]). In the latter case, clusters with hundreds of distinct building blocks have been successfully created. Addressable interactions are generated by synthesising strands of ss-DNA with unique binding regions called sticky ends. Each sticky end consists of a string with a fixed number of nucleotides from the four-letter alphabet C, G, T and A. The bases form two complementary binding pairs, C with G and T with A. The correct binding partner to any ss-DNA string therefore amounts to a reflection of the original sequence with each base in the original swapped with its complementary partner. The successful assembly of these structures relies on the possibility of generating as many distinct nucleotide sequences (and their complementary sequences) as is

necessary.

As the intended interactions will feature the exact pairing of multiple nucleotides (eight in the case of DNA bricks[86]), the chance of an incidental interaction being competitively strong is low. Effective addressable strands may therefore be built by generating random sequences of nucleotides, as long as it is ensured the sequences are not too homogeneous. Homogeneous sequences make strong incidental interactions more common, a consideration that is also relevant in protein design.[146,147] Ss-DNA sticky ends also have an inherent directionality; the $5'$ end of each strand in a complementary pair must bind with the $3'$ end of the other. For an unintended interaction to be strong, it must therefore not only feature a substantial segment of complementary base pairs in the correct order, but this sequence must also be aligned in the correct direction. The rigidity of ss-DNA in the bound state is also exploited in DNA bricks to ensure that the building blocks assemble with a strict torsional conformity, which produces the alternating brick-like structure.[86]

DNA is currently unparalleled in its use to create large numbers of specific interactions. This raises the question of what other schemes could be used to design interactions for addressable self-assembly. For example, we may want to assemble addressable clusters in cases where the range of specific interactions available is less than is granted by sequences of nucleotides. The viability of such a scheme will depend on the most complex target that can be assembled for a given limitation on the complexity of building blocks.

The complexity of a colloidal particle may be qualitatively assessed by considering how much information is required to define its interactions. For instance, a Janus particle with a single metallic patch, or a colloid uniformly grafted with a single type of polymer are examples of colloids with low complexity. On the other hand, the particles of Liu *et al.*[64] (see figure 1.3), which used a rigid DNA cage to support multiple ss-DNA sequences with a high degree of directionality, are of greater complexity.

In an idealised model of self-assembling tiles on a square lattice, somewhat similar to DNA tiles, Ahnert *et al.* developed a scheme to quantify the complexity of their designs.[133] In their model, each edge of a tile may be given a label, and a set of rules determine which labels interact with each other. "Assembly kits", consisting of a set of particle designs were created to assemble into particular targets. The complexity of any assembly kit was determined by calculating the number of bits required to describe it. By minimising the complexity of the kit for a given target, it was found that structures with high symmetry or modularity required less complex kits for reliable assembly. In

this simple model it was therefore possible to optimise the components for the minimal requirements of self-assembly. However, by minimising the size of the kit, and hence number of unique particles, the strategies developed were not addressable.

The design of interactions in more general cases may also be optimised. A successful design should have a number of desirable features such as high yield and efficiency (speed of assembly). With an appropriate fitness function, which scores more successful assembly higher, a genetic algorithm may be applied to propagate positive design features through successive generations of competing designs.[100,148,149] The drawback of this approach is that evaluation of a design in a population requires a detailed simulation, and many generations (and hence simulations) may be required to find a good solution.

The weakness of an optimisation based on a fitness function is that any design is rated by a single metric that aims to quantify the quality of assembly. In the context of self-assembly, the metric may be yield, rate of assembly or a suitable combination of such measurable quantities. An alternative approach, where all the configurations visited during a simulation are used in the optimisation, was recently proposed.[150] Here, design parameters are changed iteratively over a series of simulations so that the probability of finding the system in a configuration resembling the desired outcome is better than the ensemble average. The parameters are therefore guided to their optimal values in a rational way. In principle, this approach can be applied to any system with a statistical model (such as any system that may be simulated by MD or MC). As with any iterative approach, this method requires multiple simulations, which can be costly. It is sometimes possible to avoid this by adjusting parameters on the fly, as demonstrated in optimisation of spherically symmetric pair potentials for lattice structures during single MD simulations.[151]

In some cases it is possible to design interactions without the need for full simulation. Energetic arguments may be used to propose designs to produce a target that is stable with respect to competing structures. This approach has proved successful in protein folding[152,153] and assembly.[154] One noteworthy example of using a static design procedure to optimise building blocks is the protein design method of King *et al.*[155,156] Here, the authors searched a database of proteins for suitable candidates to self-assemble into desired target clusters. The amino acid sequences of selected proteins were subsequently modified to optimise the binding interfaces. A sample of the optimised proteins were then synthesised and characterised, demonstrating that they did indeed assemble as intended. The theoretical design procedure was therefore successful in predicting physical outcomes.

This process for optimising protein interactions is an example of "positive design", where the successful assembly of a target is ensured by stabilising the target itself. The advantage of positive design is that only the (free) energy of the intended structure needs to be considered. This is suitable in instances when a strong incidental interaction is unlikely to arise by chance. Alternatively, a design may be optimised by destabilising undesired states that compete with the target. This approach is called negative design.[135,157] The evolution of sequential strategies for self-assembly in chapter 4 could be considered an example of negative design. The most simple design (model A in figure 4.3) produced a stable target, but competing structures of similar energy prevented it from being successful. The addition of a torsional constraint in models B and C improved assembly dramatically by destabilising the competing structures relative to the target, while not changing the reduced energy of the target structure itself.

The interactions between proteins bear some relation to those between patchy colloids. Strong and selective binding between proteins is ensured through a combination of shape complementarity and interactions between attractive regions of the proteins surface. Both the shape and surface interactions of a protein are encoded into its sequence of amino acids. Patchy interactions may be optimised for self-assembly in a similar manner to protein interfaces where, instead of modifying the proteins sequence, the position of the patches are changed.

In this chapter, a global optimisation procedure to generate patchy interfaces for self-assembly will be presented. This method will then be applied to generate a set of particles using the patchy cube model to test the viability of these designs. In sections 6.3.2 and 6.3.3 the limits of interaction exclusivity for a given complexity will be explored. Subsequently, in section 6.3.6 the optimal designs will be applied to two model clusters with contrasting connectivity, to assess how the interfaces perform in more realistic addressable assembly scenarios.

6.2 Designing Patch Patterns

6.2.1 Face Design Procedure

Although the goal of optimising assembly is to maximise yield and the rate of assembly, it is rarely possible to predict these quantities *a priori*, so the design procedure would require assembly simulations to measure them. To avoid this computational expense, the optimal interfaces presented here are designed statically using energetic arguments,

similarly to the protein optimisation of King *et al.*. The spatial distribution of patches on a face is an explicit implementation of addressability, in contrast to some other theoretical models (and the addressable building blocks in chapters 4 and 5) which implicitly impose addressability with a set of interaction rules.[88,90,97,98]

For any placement of patches on the face of a cube, the ideal complementary binding partner is obtained by constructing the mirror image of that face. It is therefore trivial to ensure that native interactions are strong, and so the task of optimisation is to maximise the difference between undesired interactions and the native binding. A strong undesired interaction occurs when there is a configuration in which multiple patches overlap when the faces of non-complementary particles are in contact. We therefore propose to optimise interfaces using geometric arguments, where interactions are made as distinct as possible by maximising the collective differences between the inter-patch distances.

A more comprehensive optimisation procedure is possible. For example, beyond ensuring that all individual distances are as different as possible, a penalty could be added for any two faces having multiple similar distances. However, this is a higher-order consideration that may not drastically improve the specificity of binding beyond only ensuring inter-patch distances are distinct, which should account for the largest contribution to incorrect binding. It is geometrically unlikely for similar distances to occur on two faces in an arrangement where multiple comparable distances may overlap at once.

Let $\left(x_\alpha^{(i)}, y_\alpha^{(i)}\right)$ be the coordinates of patch α on face i, $1 \leq \alpha \leq n_\mathrm{p}$ where n_p is the number of patches per face. In the following, n_p has the same value for all faces. $1 \leq i \leq n_\mathrm{f}$ where n_f is the number of faces being designed. Each $x_\alpha^{(i)}$ or $y_\alpha^{(i)}$ may take values in the range $[-d/2, d/2]$ (where d is the edge length of the cubic building blocks). This corresponds to a coordinate system for the position of patches on a face where $(0, 0)$ is the centre of the face. Each face i also has a complementary face to which it is intended to bind. This complementary face is simply the mirror image of i, as this is the design for which all patches will perfectly align. The mirror faces are not counted in n_f.

The separation between a pair of patches α and β on face i is given by

$$r_{\alpha\beta}^{(i)} = \left[\left(x_\alpha^{(i)} - x_\beta^{(i)}\right)^2 + \left(y_\alpha^{(i)} - y_\beta^{(i)}\right)^2\right]^{\frac{1}{2}} . \tag{6.1}$$

Additionally we define the difference between two patch separations

$$\Delta_{\alpha\beta\gamma\delta}^{(ij)} = r_{\alpha\beta}^{(i)} - r_{\gamma\delta}^{(j)} . \tag{6.2}$$

There are $\frac{1}{2}n_\mathrm{p}\left(n_\mathrm{p}-1\right)$ pairs of patches per face. Hence, the total number of separations n_s is

$$n_\mathrm{s} = \frac{n_\mathrm{f} n_\mathrm{p}\left(n_\mathrm{p}-1\right)}{2}\,, \tag{6.3}$$

and the number of unique differences between separations n_Δ is

$$n_\Delta = \frac{n_\mathrm{s}\left(n_\mathrm{s}-1\right)}{2}\,. \tag{6.4}$$

To make the separations as distinct as possible, the following objective function is minimised

$$\chi = \sum_{j>i}^{n_\mathrm{f}}\sum_{\beta>\alpha}^{n_\mathrm{p}}\sum_{\delta>\gamma}^{n_\mathrm{p}}\left(\Delta^{(ij)}_{\alpha\beta\gamma\delta}/d\right)^{-2} + \sum_{i=1}^{n_\mathrm{f}}\sum_{\alpha=1}^{n_\mathrm{p}}\sum_{\beta>\alpha}^{n_\mathrm{p}}\sum_{\gamma\geq\alpha}^{n_\mathrm{p}}\sum_{\delta>\beta}^{n_\mathrm{p}}\left(\Delta^{(ii)}_{\alpha\beta\gamma\delta}/d\right)^{-2} + \sum_{i=1}^{n_\mathrm{f}}\sum_{\beta>\alpha}^{n_\mathrm{p}}\left(\frac{2r_0}{r^{(i)}_{\alpha\beta}}\right)^{N} + \sum_{i=1}^{n_\mathrm{f}}\sum_{\alpha=1}^{n_\mathrm{p}}\left[\left(\frac{x^{(i)}_\alpha}{d/2-r_0}\right)^{M} + \left(\frac{y^{(i)}_\alpha}{d/2-r_0}\right)^{M}\right]\,, \tag{6.5}$$

where r_0 is the nominal radius of patches, and M and N are large, positive and even integers. In the following work $r_0 = 0.1d$ and both M and N are set to 100. The four terms in equation 6.5 can be understood as follows.

1. All patch separation terms where the patches belong to different faces ($i \neq j$).

2. All patch separation terms where the patches belong to the same face ($i = j$). The conditions that $\gamma \geq \alpha$ and $\delta > \beta$ are required to prevent double counting and terms like $\Delta^{(ii)}_{\alpha\beta\alpha\beta}$ which equal zero.

3. A repulsion term for patches on the same face, to ensure they do not overlap. If $r^{(i)}_{\alpha\beta} < 2r_0$ then this term is very large, if $r^{(i)}_{\alpha\beta} > 2r_0$ this term is approximately zero.

4. A repulsion between patches and the edge of the face to confine patches to within the face. If a patch is closer than r_0 to any edge then this term is large, otherwise it is approximately zero.

Hence, by minimising equation 6.5 for given values of n_f and n_p we can obtain optimally distinct interfaces that are physically reasonable.

The global minima of the objective function were found using a basin-hopping procedure.[95] Basin-hopping was introduced in section 2.3 as a method for exploring the minima

of energy landscapes. Here the same technique is applied to equation 6.5. An optimisation begins from a set of n_f faces, each with n_p patches randomly placed within the bounds of the face. Each MC step involves randomly selecting a patch and displacing it in both the x and y dimensions. The displacement in each direction is picked uniformly in the range $[-d/4, d/4]$. χ for this perturbed configuration is then optimised to a tolerance of 10^{-6}, using the Fletcher–Reeves formulation of the conjugate gradient method.[158] The local minimisation uses the derivatives of the objective function, which are presented in appendix 6.A.

The probability of a move being accepted is given by a Metropolis-like criterion

$$W_\mathrm{acc} = \min\left\{1, \exp\left(-\beta\Delta\chi_\mathrm{min}\right)\right\} , \tag{6.6}$$

where $\Delta\chi_\mathrm{min}$ is the difference between the minimised scores (χ) of the new configuration and the current configuration. The fictitious reciprocal temperature β is set at 10, to allow for efficient exploration of the χ landscape. Basin-hopping optimisations of 5×10^4 steps were run for various values of n_f and n_p. For each simulation, the configuration with the lowest value of χ was taken to be the global minimum and recorded. The running lowest value of χ identified rapidly decreased at the start of a simulation, generally converging after around 10^3 steps, after which any decreases in score were generally small.

The preceding equations assume that all pairs of patch distances must be made distinct. However, if not all patches interact with each other this is no longer necessary. For instance, consider the case of two distinct patch types A and B, and their binding partners A′ and B′. If there is no interaction between A and B patches, then there is no need to make A–A distances distinct from A–B or B–B distances because the overlap of different patch types does not lower the energy of the system. In this case only contributions from pairs of matching patch type are included in the first two terms of equation 6.5. As the third term relates to the physical overlap of patches on a face, all pairs must be included there.

6.2.2 Optimal Designs

The optimisation algorithm was first applied to faces where all inter-patch distances must be made distinct. In other words, all patches on any given face are of the same type. When these designs are applied to the patchy cube particles in section 6.3.2, a patch alphabet of size two is used. All patches on designed faces are of one type, and all

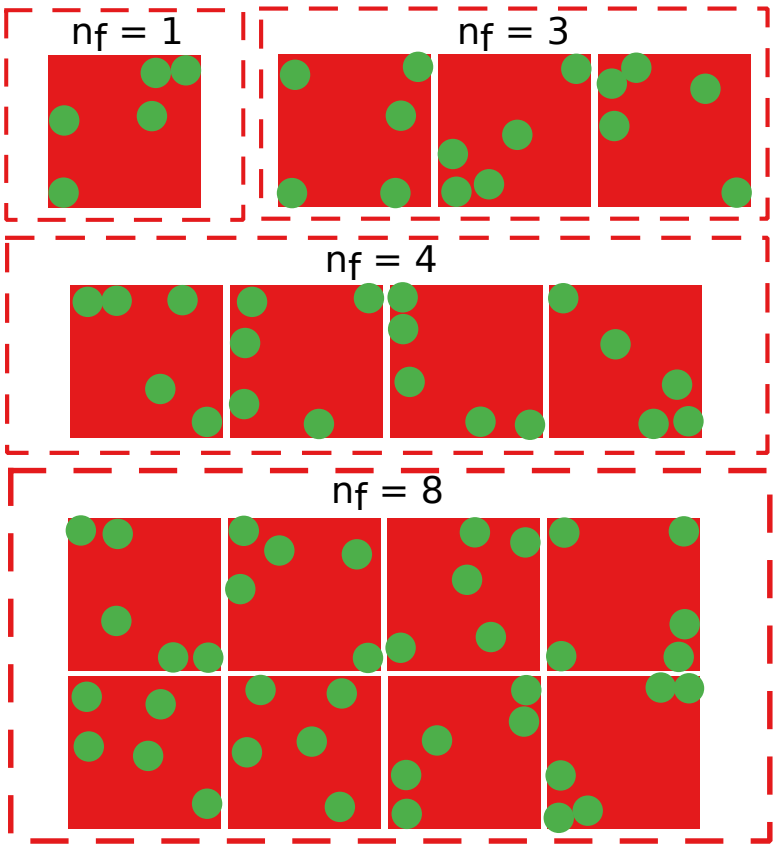

Figure 6.1: Optimised patch patterns with $n_\mathrm{f} = 1$, 3, 4 and 8 and $n_\mathrm{p} = 5$.

patches on complementary faces are of the other type. Patches on designed faces may only interact with those on mirrored faces, and do so with equal strength ε. The two-letter patch alphabet is similar to a (non-palindromic) pair of complementary ss-DNA strands.

Designs for a range of values of n_f and n_p were produced, a selection of which are presented in figure 6.1. It may be confirmed that the differences between pairwise patch distances are as distinct as possible by considering the distribution of $r_{\alpha\beta}^{(i)}$ values, which is shown in figure 6.2. For each value of n_f the values of $r_{\alpha\beta}^{(i)}$ are close to equally distributed between the extrema, indicating that the patch separations are as distinct as possible.

The minimum and maximum patch separation in each design is achieved by having two patches adjacent to each other on the same face, and two patches in opposite corners on the same faces respectively. Examples of these adjacent and opposite patches are visible in figure 6.1. As n_f increases, patches begin to "spill" slightly over the edges of the faces and overlap with each other. This is a trade off between the extra configurational space available to patches by exceeding the boundaries, and the cost of the repulsion from

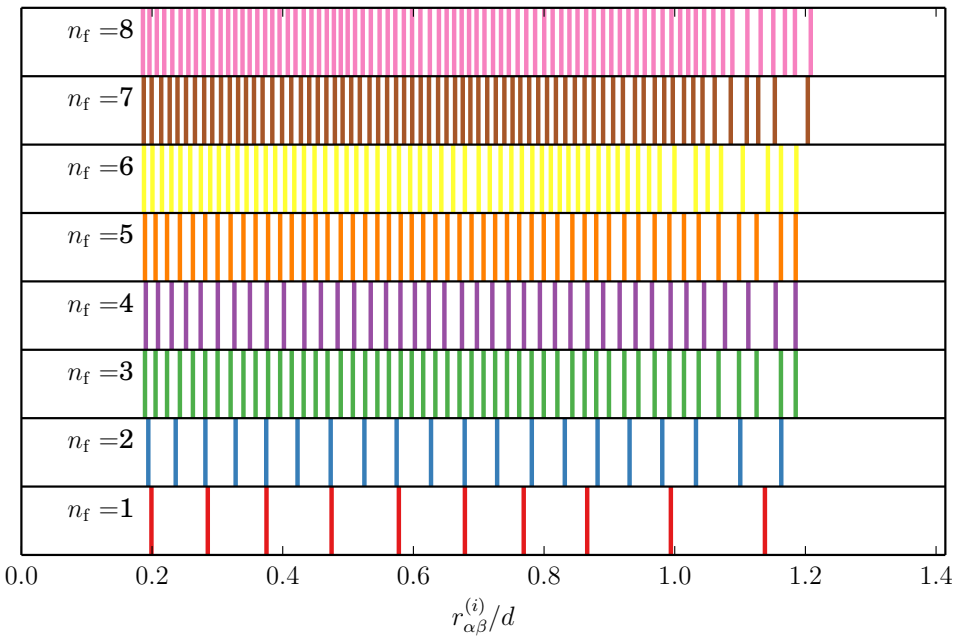

Figure 6.2: Differences in pairwise distances for the optimal solutions with n_f in $[1, 8]$ and $n_p = 5$.

the edge, controlled by M and r_0. Similarly, the slight overlap of pairs of patches is also seen. Both of these effects are a result of using soft repulsions to define the radius of patches. Overlaps could be made less prominent by increasing the M and N values. However, making the repulsion too steep can cause instability in minimising the objective function.

It is important to note that, when initiated from different initial configurations, the basin-hopping procedure reaches final optimal configurations with similar values of χ, but that correspond to significantly different patterns of patches on the designed interfaces. In other words, there is no single decisive global minimum of χ, but rather there are many low-lying minima with similar scores. Therefore, the designs presented in figure 6.1 may not be the true global minima, but the performance in assembly should be similar between distinct solutions.

Designs where every patch on a given face is of a unique type were also generated. In these designs there are n_p patch types on a given face. Later, in the assembly simulations of particles using these designs, an alphabet of $2n_p$ is used, as each patch type on a given face also has a complementary type. Interactions between complementary types have uniform strength ε and all other combinations do not interact. These designs therefore represent a larger alphabet of specific interactions and complementary partners. As

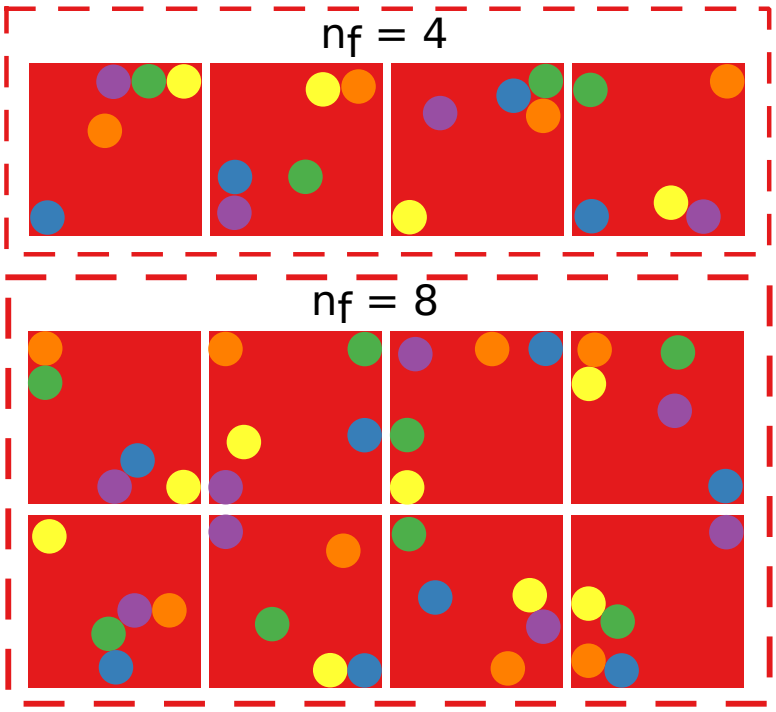

Figure 6.3: Optimised interfaces for $n_\mathrm{f} = 4$ and 8 with $n_\mathrm{p} = 5$ and five patch types. Each patch type is shown as a different colour.

discussed in section 6.2.1, the introduction of different patch types means that the same pairwise distances may be reused so long as they occur between different combinations of patch types. Two examples of designs with $n_\mathrm{p} = 5$ are shown figure 6.3, where different patch types represented by different colours.

The plot of pairwise distances in figure 6.4 demonstrates that the relevant distances are indeed well separated. Here the separations for faces with $n_\mathrm{p} = 3$ are shown. Each of the three patches is a different type, and so there are three pairwise combinations of types. Each of the combinations is represented by a different colour in the plot. Comparing pairs of patch types within a given design (number of faces), duplicate distances are often seen, demonstrating that inter-patch distances are reused where possible in order to spread matching pairs as widely as possible.

The individual histograms for each pair of patch types in figure 6.4 are most often not perfectly identical. This is not, however, an indication of the failure of the optimisation procedure, but rather demonstrates how the constraints on the design (terms to prevent patches from overlapping or moving outside the bounds of a face) cause frustration and may mean a perfectly distributed set of distances is not geometrically possible.

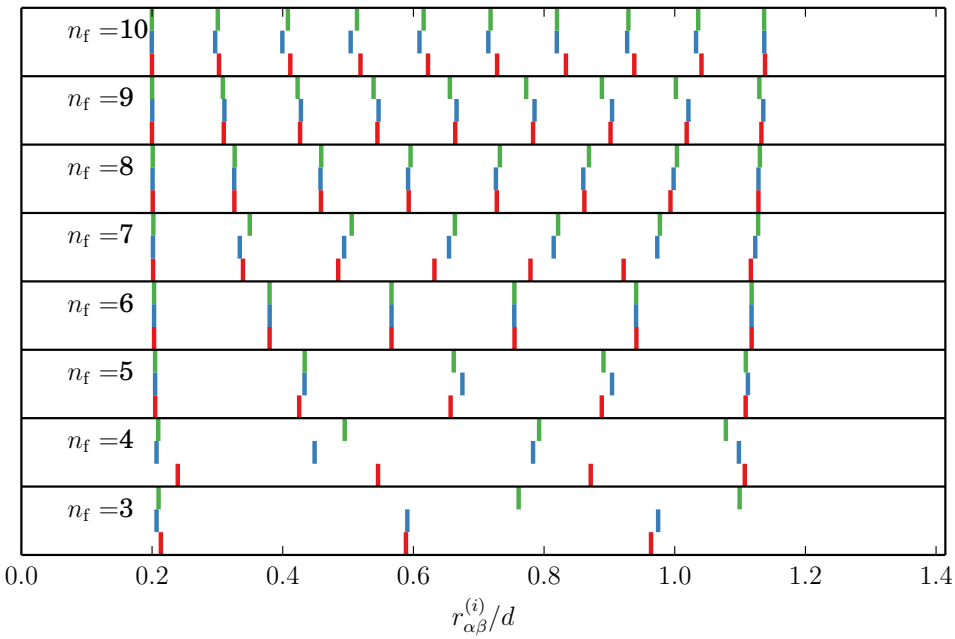

Figure 6.4: Inter-patch distances for a set of optimal solutions with n_f faces in the range $[3, 10]$ and $n_p = 3$ with 3 patch types per face. For each values of n_f there are three distributions, one for each of the three pairs of patch types. Each pair of types is represented here by a different colour.

6.3 Assembly Simulations

Assembly simulations, using the optimal designs obtained in section 6.2.2 are split into two branches. First, the effectiveness of the design procedure in producing faces that interact exclusively is tested. This is done by simulating a system where each particle has only one patterned face, either an optimal design or its complementary mirror image. The largest cluster that will form is therefore a dimer. Dimers were chosen so that binding could be tested independently of structure. This choice isolates the test of exclusivity from other factors that influence self-assembly, such as the role of connectivity seen in chapter 5. If the designs are successful then each dimer should only bind to its complementary partner.

Second, the designs are tested in a more realistic assembly scenario by applying them to the assembly of two example addressable clusters. Our choice of clusters are the octameric cube, introduced in chapter 4, and a cluster resembling the Vitruvian Man, a partially addressable version of which was first assembled experimentally by Liu *et al.*.[64] Ideal structures of both clusters are shown in figure 6.5. These simulations better test the application of the optimal designs to a self-assembly problem, where all interfaces are re-

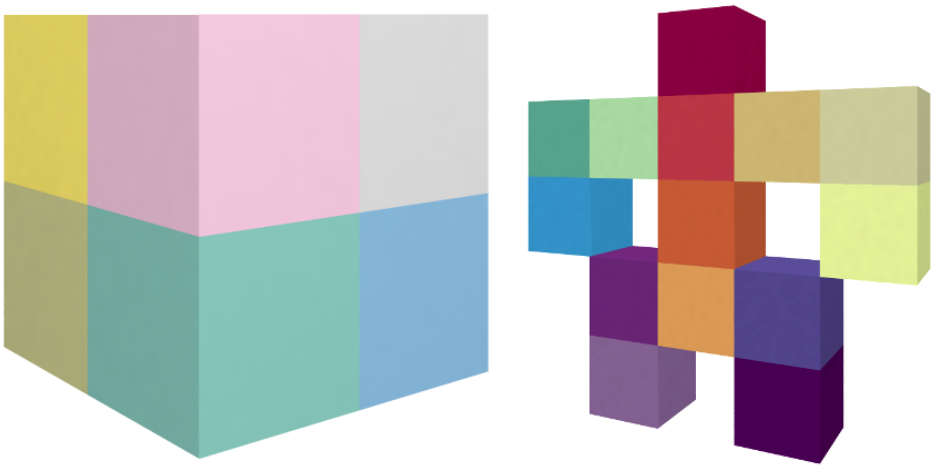

Figure 6.5: The two discrete clusters used to test the assembly of our addressable faces. The octamer cube (left) contains eight particles and requires designs with $n_\mathrm{f} = 12$, while the Vitruvian Man (right) contains fourteen particles and requires designs with $n_\mathrm{f} = 13$.

quired to bind exclusively and simultaneously in order to self-assemble our desired target.

6.3.1 Reduced Units

Throughout this chapter the reduced temperature T^* is defined as

$$T^* = k_\mathrm{B}T / \left|E_\mathrm{face}\right| \,, \tag{6.7}$$

where E_face is the native (minimum) interaction energy between a face and its mirrored partner. This definition is different from those used in previous chapters. It is a convenient definition because, for faces with unique patches, the native interaction energy is simply equal to $-\varepsilon$ (the patch–patch well depth) multiplied by the number of patches per face. Furthermore, all faces have the same number of patches in a given simulation so E_face is the binding energy for all interfaces in a simulation. In the minority of our simulations where not all patches on a face are unique it is assumed that $E_\mathrm{face} = -n_\mathrm{p}\varepsilon$. The exact interface binding energy in this case has small contributions (typically $< 10^{-2}\varepsilon$) from all pairs of patches across the bound interface, these vary slightly between interfaces. The reduced time t^* is defined as before, in equation 3.19.

6.3.2 TWO LETTER ALPHABET

As discussed in section 6.2.2, when running the optimisation algorithm from different starting points, distinct final designs with similar scores will be obtained. This means that under different conditions the algorithm generates equally competitive yet distinct designs. Nevertheless, the different designs with similar scores show variations in the success of their assembly in the simulations because the static design procedure cannot foresee all dynamic aspects of assembly. In order to ensure that the particular behaviour of any one design does not lead to misleading results, all results for each value of n_f and n_p here are averaged over five designs. A further ensemble average of five trajectories per design was taken, making a total of 25 simulations for each combination of n_f and n_p

Each simulation begins from a different state, always a disordered monomer fluid and consists of an equilibration period of 10^7 cycles, after which the simulation continues for a further 10^7 cycles where yield of correctly assembled dimers is averaged every 10^5 cycles. Both parts of the simulation are conducted at $T^* = 0.0267$. This temperature was chosen based on preliminary simulations, and later we will see that this is a near optimal temperature to balance the thermal stability of the target and kinetic accessibility. In all simulations each species of building block is present at a number density of $0.0005d^{-3}$.

As previously mentioned, for the designs where all inter-patch distances were made distinct, a patch alphabet of size two is used in the patchy cube simulations. One patch type was used for all patches on designed faces, and the other for all patches on the complementary faces. The interaction alphabet ensures that patches only interact with those of the other type. Yields for the dimer systems using designs with n_f in the range $[2, 7]$ and $n_\mathrm{p} = 3$, 4 and 5 are shown in figure 6.6. This plot shows that a good yield is only achieved with the smallest number of interfaces and patches. With larger numbers of interfaces yields are poor and continue to fall as n_f is increased. Adding more patches, and so increasing the complexity of the design, does not help and almost always makes the yields observed worse.

The limitations of the designs using this size of alphabet can be understood from a simple consideration of the number of competing, incorrect interactions and how the number of these grows with n_p. Consider the largest contributions to the energy, which occur when patches overlap perfectly. In the native configuration, all patches overlap and so the interaction energy E is $-n_\mathrm{p}\varepsilon$. Each of the n_s patch separations, equation 6.3, present on the designed face are mirrored on the complementary face. Hence, after rotating one face 180° from the correct orientation about the inter-particle axis there are n_s configurations

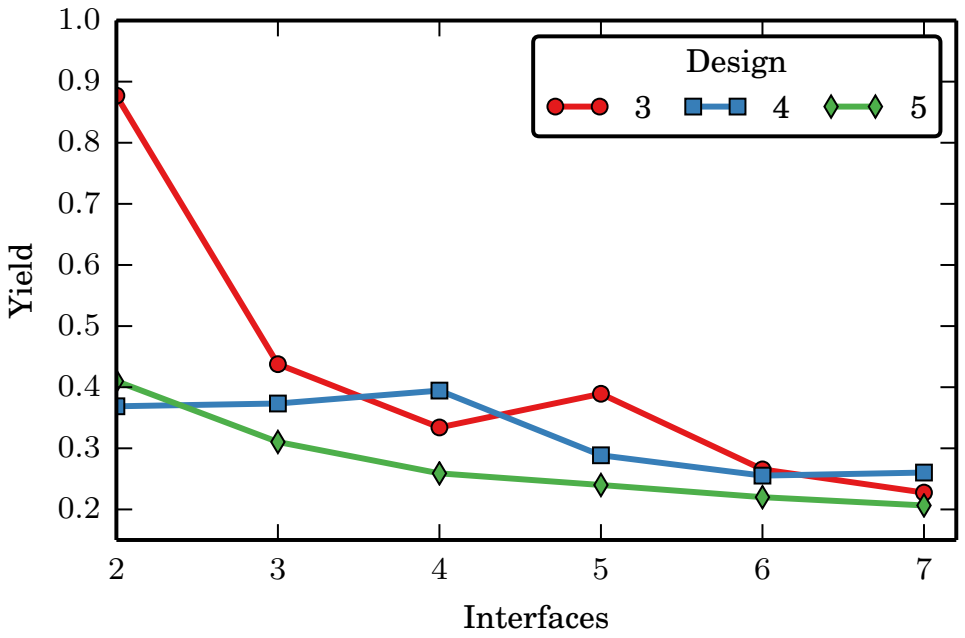

Figure 6.6: Equilibrium yields of dimer systems against the number of interfaces (n_f) for $n_\mathrm{p} = 3$, 4 and 5.

where, through the translation of faces in contact, two patches perfectly overlap. Each of these configurations has an interaction energy of $E = -2\varepsilon$. n_s scales roughly as the square of n_p (equation 6.3) whereas the strength of the native interaction energy scales only linearly with n_p. Furthermore, there are n_p^2 interactions of energy $E = -\varepsilon$ from the binding of a single patch on each face. Larger values of n_p will therefore increase the total statistical weight of of incorrect interactions faster than that of the native state.

6.3.3 Larger Alphabets

We will now investigate how the use of a larger, but limited, alphabet of different patch types can be used to improve over the limitations of a single pair of complementary patches. Unless stated otherwise, the size of the interaction alphabet for all simulations presented from now on is $2n_\mathrm{p}$, meaning every patch on a given face is a unique type and only interacts with the complementary type. Later in section 6.3.4, designs with a smaller number of exclusive interactions will be considered, where the size of the alphabet is $< 2n_\mathrm{p}$.

Yields for $n_\mathrm{p} = 3, 4$ and 5 and n_f in the range $[3, 20]$ are plotted in figure 6.7. The simplest designs, featuring three patches per face, display high yields for small numbers

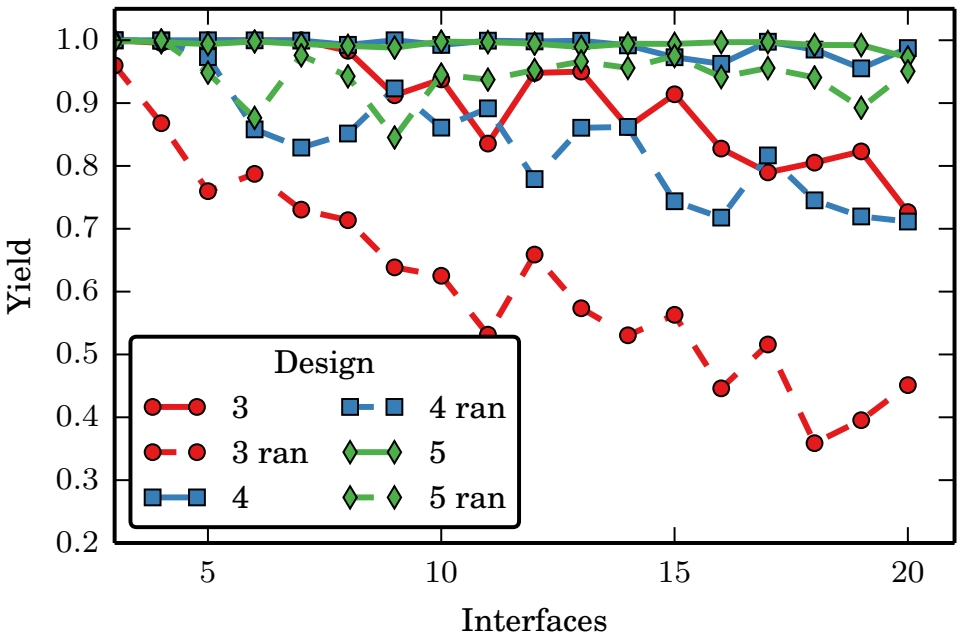

Figure 6.7: Yields for dimer systems as a function of the number of interfaces designed. Yields for designed interfaces with $n_\mathrm{p} = 3, 4$ and 5 are shown alongside randomly generated faces.

of interfaces (≤ 8) but beyond this point the reliability of the designs suffers. For these larger values of n_f, faces are no longer distinct enough, and kinetic traps due to unintended interactions reduce the yield.

It should be noted that although the yields for $n_\mathrm{p} = 3$ may appear to remain high, a yield of approximately 0.8 means that any interface has a one-in-five chance of binding incorrectly. For a target structure with N independent links there would only be a 0.8^N chance of all links forming simultaneously. Hence, designs with yields of this magnitude may not be of practical use for some targets.

Figure 6.7 demonstrates how additional information (in this case, larger n_p) aids in creating more distinct interfaces. Yields for both $n_\mathrm{p} = 4$ and $n_\mathrm{p} = 5$ remain very high for the entire range of n_f, with those for $n_\mathrm{p} = 4$ showing only a slight tendency to fall towards the upper range of n_f studied here. The additional patches grant the design procedure more degrees of freedom with which to explore the scoring function. Additionally, with more patches per face, the difference in energy between the native state and unwanted configurations grows because accidental interactions between one or two pairs of patches represent a smaller fraction of the total native energy.

Also plotted in figure 6.7 are yields for designs which were generated by placing

patches randomly (taking coordinates from a uniform distribution) on the faces, subject to the constraints that patches are not within r_0 of each other or the edge of the face. Similarly to the designed yields, each point was obtained from 25 simulations using 5 random designs.

Yields of dimers using randomly patterned interfaces for both $n_\mathrm{p} = 3$ and 4 perform worse than with the corresponding designed interactions, and fall to impractically low levels for use in assembly of clusters more rapidly with respect to the number of interfaces. This confirms both that design is necessary even when multiple specifically interacting patches are used, and that our proposition of maximising the differences between pairwise distances between patches is an effective method for generating addressable interfaces. The yield for random designs with $n_\mathrm{p} = 5$ remain competitive across the range of n_f simulated. This may be a sign that enough information has been added to the faces to make even random faces distinct enough to assemble correctly, at least up to the largest value of n_f simulated here.

There are two sources of variation between the individual trajectories used to calculate the yields in figure 6.7. First, there is a small variation in performance for the same design in each of its five separate simulations, which start from distinct configurations and use different random seeds. Second, there are intrinsic differences in the performance of different designs. When the mean yield is high (> 0.9) the standard deviation about that mean for any individual design is small, around 0.01. However, the standard deviation about the mean yield for all five designs at a given value of n_f and n_p is larger, typically in the range $[0.02, 0.05]$. These results demonstrate that while there is a high certainty in the performance of any particular design, different designs at the same level of complexity achieve somewhat different levels of success.

The standard deviation about the mean yield for random designs, and where yield is poor, is much larger, typically in the range $[0.1, 0.2]$. In these cases fluctuations in the yield for independent simulations are large because of the increased likelihood of kinetic traps arising from incorrectly bound particles. Furthermore, there is a large uncertainty in the performance of random designs; in some cases a random design may be good (with distinct interfaces) by chance. However, very poor designs are also possible and more common.

The effectiveness of the design procedure may be further demonstrated by comparing the results for designed interfaces with interfaces created by perturbing the optimal designs. In order to do this the five optimal designs for $n_\mathrm{p} = 3$ at $n_\mathrm{f} = 5, 13$ and 20 were

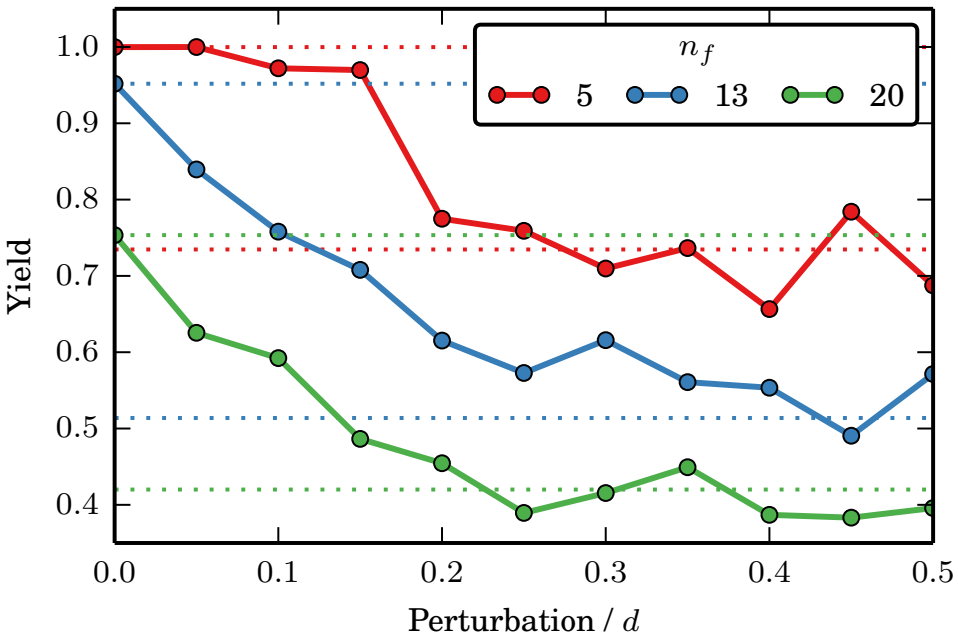

Figure 6.8: Yield lines for a series of designs created by randomly perturbing patch positions in the optimal designs for $n_\mathrm{p} = 3$ at $n_\mathrm{f} = 5, 13$ and 20. The upper and lower dotted lines for each value of n_f represent the yield for the unperturbed designs, and random designs respectively.

taken as a starting point. For each design twenty perturbed versions were created, by moving all patches from their original positions by displacements taken from a Gaussian distribution with standard deviations in the range $[0.05, 0.5]d$. The perturbation is mirrored on the complementary face, so that all patches still overlap perfectly in the native configuration. This is therefore a test of how sensitive assembly performance is to the design, rather than the precision of synthesis (which will be considered in section 6.3.5). Each perturbed design was then simulated in the manner described above and the mean yield of dimers determined.

The yields for the perturbed designs are shown in figure 6.8. This plot shows how the perturbed designs compare to the optimal interface, which is indicated by the data point at a perturbation of 0 and the higher of the two dotted lines in each case. Perturbations of the designed structure rapidly decrease the yield as the size of perturbation is increased, and hence make worse interfaces for addressable assembly. Once the size of perturbation reaches $0.3d$ the performance has generally fallen to the level of a randomly designed face, indicated by the lower of the dotted lines.

Up to this point all simulations have been conducted at the same optimal reduced

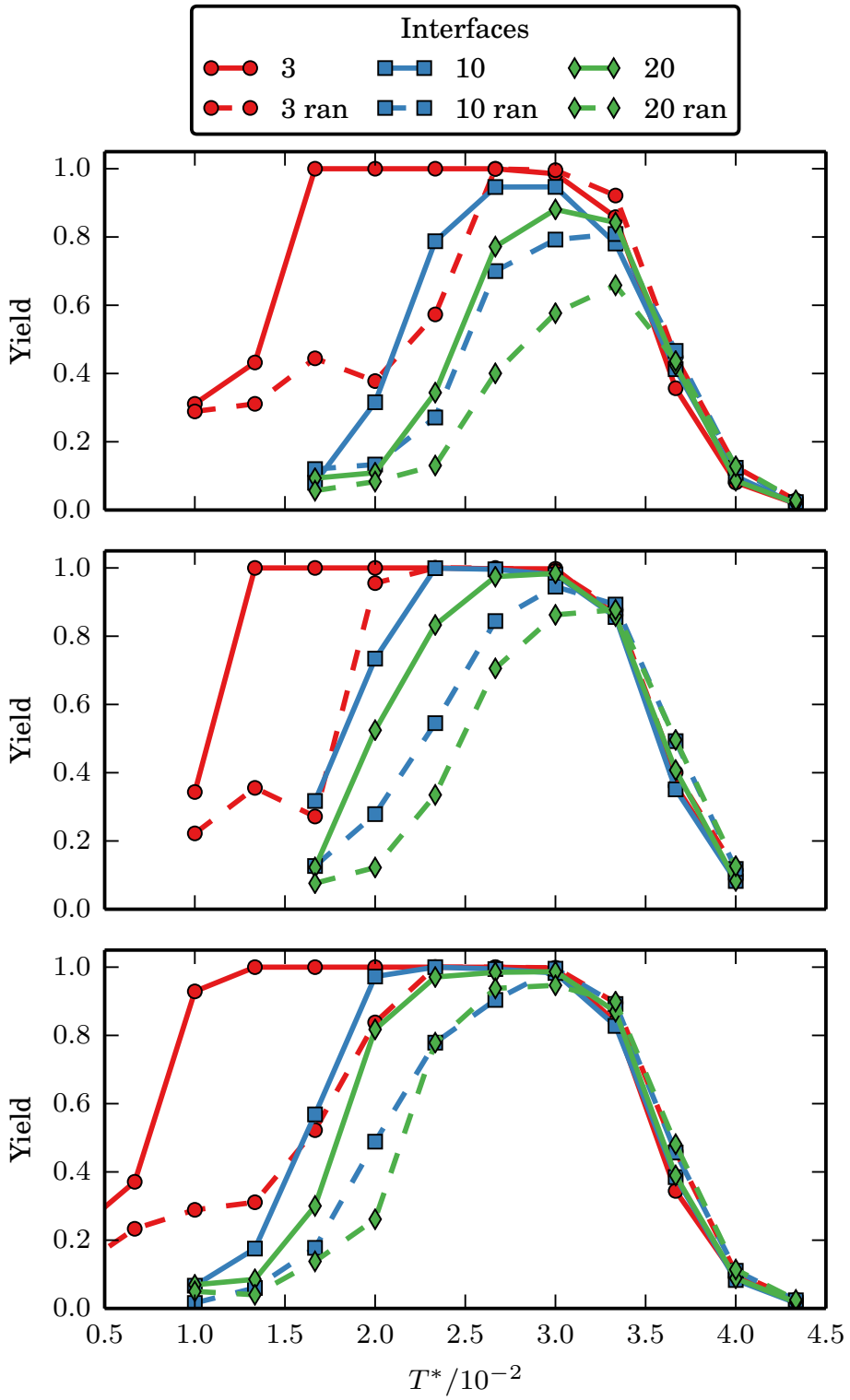

Figure 6.9: Yields of dimer systems as a function of temperature for $n_\mathrm{p} = 3$ (top), 4 (middle) and 5 (bottom). For each value of n_p three values of n_f are plotted ($3, 10$ and 20) for both optimally designed and randomly generated interfaces.

temperature. However, the performance of designs across a range of temperatures is also an important consideration for the reliability of assembly. Simulations at a variety of temperatures were conducted for each value of n_p and $n_\mathrm{f} = 3, 10$ and 20, with both the optimal and random interfaces. The yield curves as a function of temperature are shown in figure 6.9. A number of characteristics of the assembly of the designs can be extracted from these plots. Firstly, for all values of n_p and n_f, randomly designed interfaces have a lower peak yield and a narrower temperature range of near-peak performance. These designs are therefore less robust with respect to deviation from the ideal temperature. Random designs perform worst relative to optimal designs on the low-temperature side, where the incorrect binding of particles is common and can permanently stall assembly. On the high-temperature side of the yield curves, the random designs give similar yields to designed interactions. However, at these temperatures the target is not thermodynamically stable, and so this cannot be taken to mean that the random designs are competitive.

Secondly, for a given number of patches, designs with an increasing number of faces become less resilient to changes in temperature. After reaching a peak yield (lower than the corresponding designed interface) the random design's yield falls quickly as temperature is lowered. This reflects the fact that random designs are more seriously affected by kinetic traps in the form of unwanted interactions, which significantly slow the progress of assembly. In other words, for a given limit on the complexity of a design (n_p) designing a greater number of specifically interacting faces becomes more challenging. The low-temperature edge of the yield curves falls sooner for greater numbers of faces as more numerous kinetic traps hinder assembly.

Lastly, figure 6.9 demonstrates that increasing the information content of the designs can improve the reliability of assembly across a range of temperatures. Comparing yield curves for the same n_f across different plots, the regions of high yield broaden as n_p changes from 3 to 4 to 5. This is a composite effect of the increase in the ratio of native binding energy to unwanted interactions, and also the extra complexity in design allowing for a greater number of specifically interacting interfaces to be designed.

6.3.4 An Intermediate Alphabet

As well as lowering the number of patches per face, the complexity of designs may also be reduced by using a smaller alphabet of patches. In the previous section all patches on a given face were of a different type, and an alphabet of $2n_\mathrm{p}$ was used for the assembly simulations. Here, three patch types for a given face are used and, as each patch has

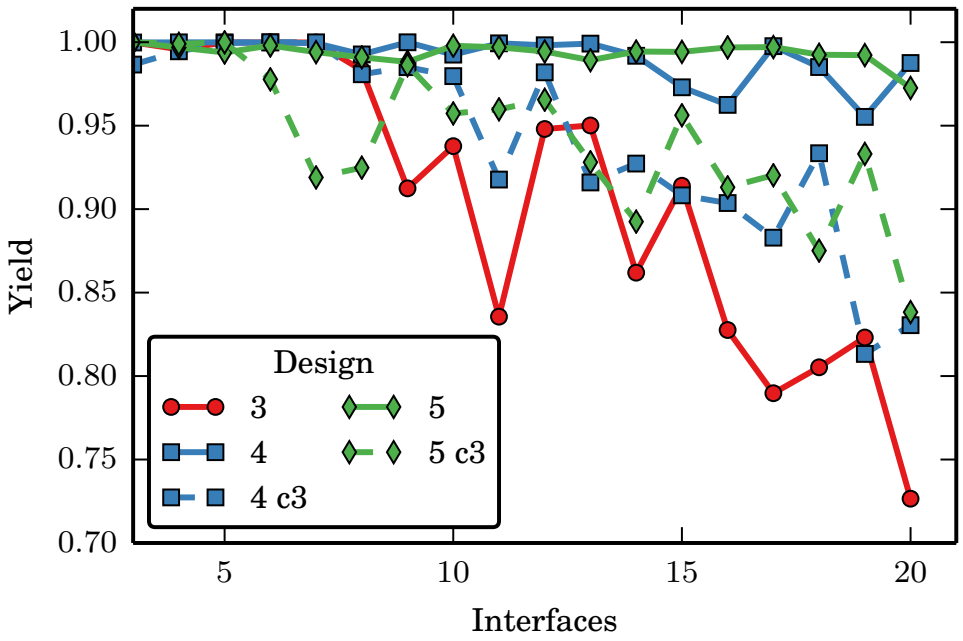

Figure 6.10: Yields as a function of n_f for designed interfaces with $n_\mathrm{p} = 4, 5$ yet with only three patch types. Also plotted are the yields of designed Interfaces for $n_\mathrm{p} = 3, 4$ and 5 as in figure 6.7.

a complementary partner, an alphabet of size six is used in the assembly simulations. Hence, when $n_\mathrm{p} = 4$ one patch type appears twice per face, and for $n_\mathrm{p} = 5$ two patch types appear twice per face. Designs were generated for n_f in the range $[3, 20]$ and $n_\mathrm{p} = 4$ and 5.

Yields for the designs with the more restricted set of patch types are plotted in figure 6.10. They are plotted alongside the yields for the interfaces with an alphabet of size $2n_\mathrm{p}$ (already presented in figure 6.7). The yields obtained with the intermediate patch alphabet are worse than those for designs using the same number of patches, and a larger alphabet. However, they are generally better than the yields for designs with $n_\mathrm{p} = 3$ and three types on a given face. These designs are of intermediate complexity, with more patches than the previous presented designs with $n_\mathrm{p} = 3$ yet fewer patch types than the related $n_\mathrm{p} = 4$ and 5 designs. This, therefore, reinforces the link between complexity and addressability seen so far, as these designs fall between those with higher and lower complexity.

6.3.5 IMPRECISION IN PARTICLE SYNTHESIS

For all designs presented above, it has been assumed that particles may be patterned with total accuracy. The complementary binding partner is always a perfect mirror image, even for random and perturbed designs. However, such a high level of control over design is not possible in practice. Any method for synthesising colloids would have a finite precision, and so it is important to consider how robust the optimal designs are to random errors in the implementation of a design. Error in patch placement has previously been investigated, in the context of single-component assembly, in a system of patchy spheres designed to assemble into icosahedral clusters.[70] Here it was found that the introduction of error into the placement of patches left yields almost unaffected up to a threshold value, beyond which the yield rapidly fell with increasing inaccuracy.

A series of imperfect building blocks were created by perturbing the optimal configurations for $n_{\mathrm{p}} = 4$ and $n_{\mathrm{f}} = 10$. Unlike the perturbed designs presented previously, where perturbations were applied identically to both partners, here each face in a binding pair is perturbed independently. A perturbation is created, as before, by moving each patch with a displacement in each direction picked from a Gaussian distribution. These designs therefore represent a statistical error in the placement of patches, similar to the limited precision of producing physical patchy particles.

The yields of designs with independently perturbed faces as a function of temperature are shown in figure 6.11. The red line is the result for the optimal design (already presented as the solid blue line in the centre plot of figure 6.9). As the size of perturbation is increased, the maximum yield falls and the conditions for optimal assembly move towards lower temperatures. The difference in potential between the energetic ground state and competing configurations is reduced as the size of perturbation is increased. Therefore, lower temperatures are required to achieve a high population of the ground state, and the optimal temperature for assembly falls correspondingly. Yields are lower at optimal temperature because kinetic traps caused by incorrectly bound particles are more common.

Assembly conducted at the optimal temperature of the ideal (unperturbed) design, will produce suboptimal yield for perturbed designs. In any real case, where imprecision in patch patterning is inevitable, it is therefore important to optimise conditions to take into account statistical error in the implementation of a design. After perturbation, the perfectly aligned interfaces are no longer the minimum configuration as the patches no longer overlap completely. Instead, the most stable configuration is where the faces in

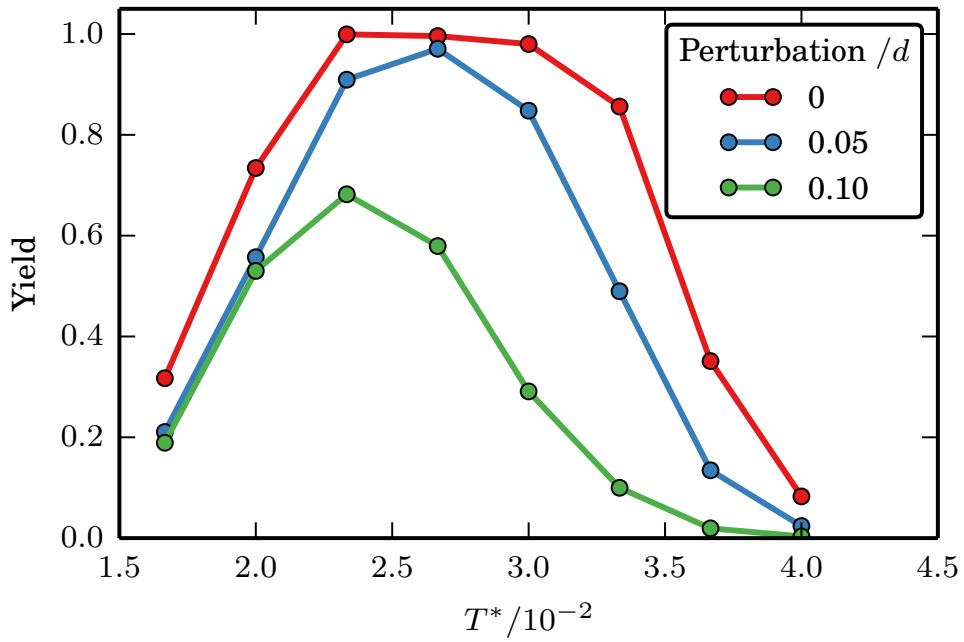

Figure 6.11: Yield as a function of temperature for optimal designs, and designs with binding pairs independently perturbed for $n_\mathrm{f} = 10$ and $n_\mathrm{p} = 4$. The width of the Gaussian distribution in the perturbed designs are 0.05 and $0.10/d$.

contact have translated and rotated to achieve the optimal overlap of patches. Therefore, clusters assembled from imprecise interfaces may deviate from the intended geometry. For small perturbations this may amount to a slightly distorted structure. However, there is a risk that for large levels of imprecision, or in large clusters with many links, the energetic ground state may no longer resemble the target.

6.3.6 CLUSTER ASSEMBLY

Having tested the designed interfaces independently with simulations of dimers, attention is now turned to using the designs to assemble larger addressable clusters. The two test clusters (figure 6.5) were chosen for their contrasting connectivity. The Vitruvian Man cluster is linearly connected, so that a single broken link will break the entire structure. On the other hand, the octamer cube is highly connected with each particle being bound to its three neighbours. There are therefore many small loops of bonds within this cluster which reinforce correct assembly.

The octamer target uses twelve interfaces and their complementary partners, while the Vitruvian man requires thirteen. For both targets simulations consisted of enough particles to form one complete target. This choice avoids the formation of configurations with

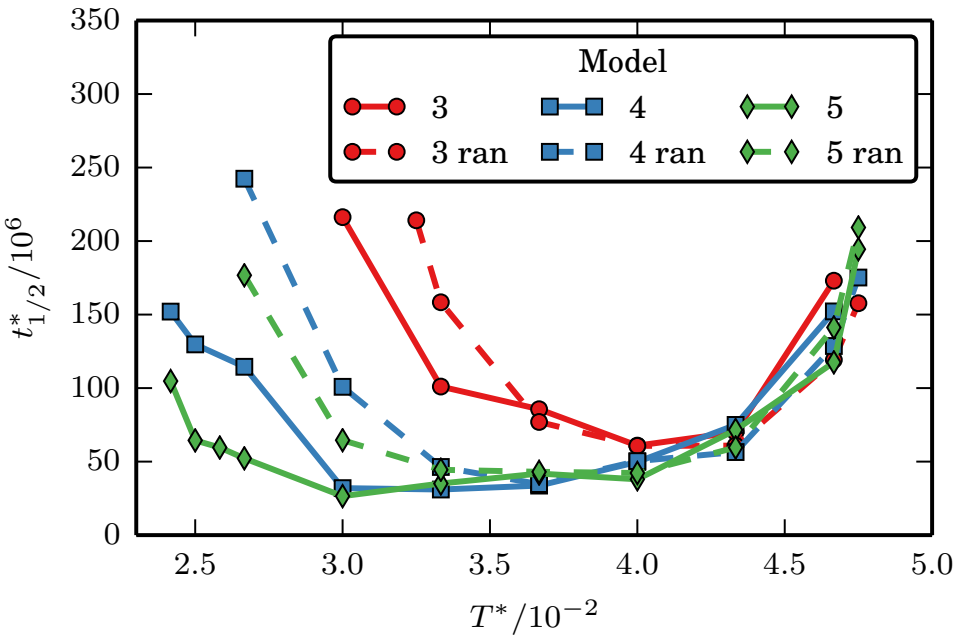

Figure 6.12: Time to 50% yield for the octamer target as a function of temperature for $n_\mathrm{p} = 3, 4$ and 5 with both optimal and random designs.

incompatible fragments, like those studied in chapter 5, and instead focuses on the ability of the designs to avoid incorrect bindings and successfully assemble. As before, all results are averages taken over 25 simulations: five simulations for each of five designs. For both targets the overall number density of building blocks in the simulations is $0.05d^{-3}$.

Figure 6.12 shows the time taken to reach 50% yield of the octamer target as a function of temperature, for all values of n_p simulated. Results for optimised and randomly designed patterns are plotted alongside each other. The plot clearly demonstrates that the $n_\mathrm{p} = 3$ designs are slower to assemble and more sensitive to temperature than those with $n_\mathrm{p} = 4$ and 5. The characteristics of assembly improve as n_p is increased, as can be seen by the wider region of rapid assembly. The broadening of the temperature window occurs mainly through accessing assembly at lower temperatures, which is due to designs with greater n_p being less prone to the misbinding of particles.

Randomly designed interfaces show worse performance with respect to the optimal designs at low temperatures. As with the yields of dimers, this is because random designs are more susceptible to incorrectly bound particles. However, at optimal temperatures for each value of n_p, the random designs show similar performance. Later, we will see contrasting behaviour for the Vitruvian man cluster.

One source of kinetic traps on the pathway to the octamer target involves the larger

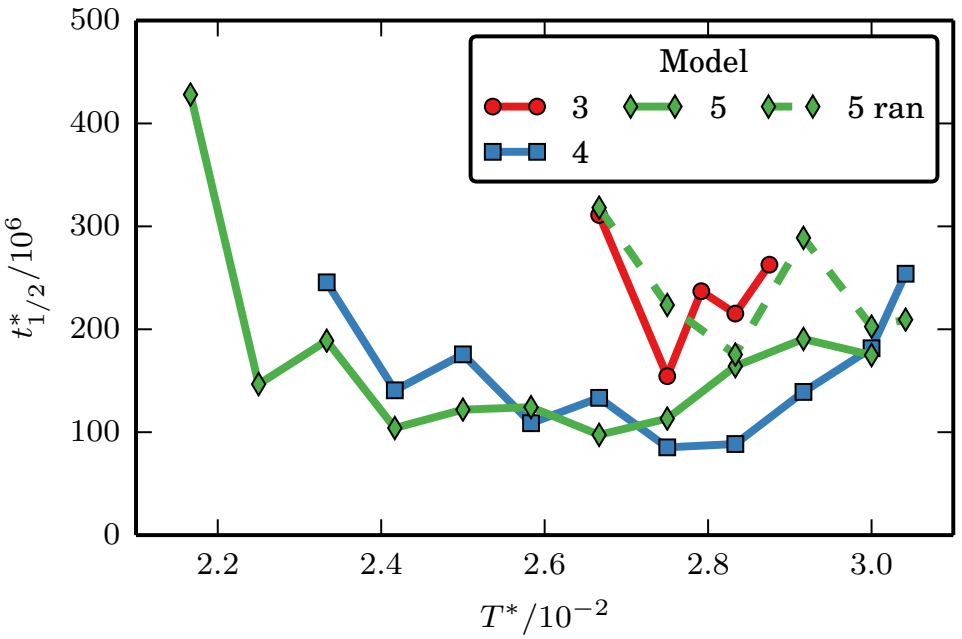

Figure 6.13: Time to reach 50% yield for the Vitruvian man cluster as a function of temperature. Displayed are the results for optimised designs with $n_\mathrm{p} = 3, 4$ and 5 and also $n_\mathrm{p} = 5$ with randomly placed patches. The randomly designed faces for $n_\mathrm{p} = 3$ and 4 did not achieve 50% yield within the simulation time at any temperature.

interfaces created when two or more particles bind correctly. For example, after two particles have bound correctly they create two new interfaces with dimension $2d \times d$. A third particle may now straddle the pair, binding to the higher-order interface. The pattern of patches on the $2d \times d$ interface is not considered by the optimisation algorithm, and so no effort has been made to ensure this new pattern of patches is incompatible with any of the building blocks. The three-patch system suffers the most from this trap because any unintended interaction of two or more patches represents a large fraction of the native binding energy.

The corresponding plot of time to reach 50% yield for the Vitruvian man target is shown in figure 6.13. In this case, randomly generated patterns perform very poorly, with designs for both $n_\mathrm{p} = 3$ and 4 failing to achieve 50% yield at any temperature simulated. Both the optimal designs with $n_\mathrm{p} = 3$ and the random designs with $n_\mathrm{p} = 5$ display only narrow regions of efficient assembly. In contrast, the lines for $n_\mathrm{p} = 4$ and 5 both display a broad region of temperature where assembly is efficient. Assembly extends to low temperatures not possible for the octamer target because the Vitruvian man structure never produces aggregates with patchy interfaces larger than a single face.

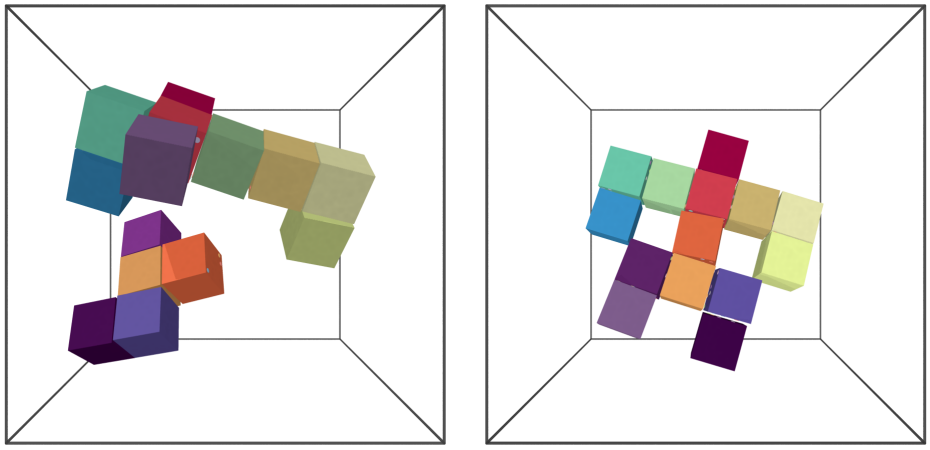

Figure 6.14: Simulations snapshots showing an incorrectly assembled (left) and correctly assembled (right) Vitruvian man. The failed example is taken from a simulation of an optimised patch design with $n_\mathrm{f} = 13$ and $n_\mathrm{p} = 3$, while the successful cluster uses a design with $n_\mathrm{p} = 5$. Both simulations were conducted at $T^* = 0.0267$. Each particle type has a unique colour, so by comparison of the two images it is possible to see where particles in the left image have bound to an incorrect partner.

The Vitruvian man needs building blocks of higher complexity for good assembly than the octamer cube. This is due to the lower connectivity of particles in this structure creating more strict requirements on the building blocks to bind in selectively. In contrast, each building block in the octamer target is supported by three neighbours in the target. Therefore, inserting a particle in the wrong place in the octamer will generally result in the loss of further correct bonds and is likely to be more energetically unfavourable than an incorrect link in the Vitruvian man cluster.

The simulation snapshots in figure 6.14 demonstrate how the limited information in designs with $n_\mathrm{p} = 3$ struggle to create many distinct interfaces. In the figure, a correctly assembled cluster, using an optimal design with $n_\mathrm{p} = 5$, and a system with incorrectly assembled clusters obtained from a design with $n_\mathrm{p} = 3$ are shown. Each particle type is given a unique colour, which is shared between the snapshots. The colour coding allows us to see, by comparison with the correct structure, where particles in the incorrect structure have bound to the wrong partners.

6.4 Discussion

Similarly to chapter 5, this chapter has focused on optimising self-assembly. However, while the previous chapter investigated ways to exploit the network of links and binding

energies between building blocks, here the aim has been to optimise the addressable interactions themselves. Chapter 5 neglected the fine detail of interactions between the patchy cubes in order to study only kinetic traps caused by incompatible fragments. By reintroducing the more detailed patchy model, the performance of interactions can be assessed, as incorrectly bound particles may occur.

Chapter 5 demonstrated how the bonding network of a cluster, in particular the presence of loops, plays an important role in the occurrence of configurations with incompatible fragments. The simulations of cluster assembly in section 6.3.6 featured enough building blocks to assemble exactly one target. This was done in order to eliminate the effect of frustrated configurations, like those seen in chapter 5, so that the specificity of interfaces could be tested in isolation. However, the use of targets with contrasting connectivity in this chapter has further demonstrated how addressable self-assembly can depend on the structure of the target. The reinforcement of correct assembly provided by the loops in the octamer target make this structure more forgiving to less optimised interactions. In contrast, the linearly connected Vitruvian man target had more strict requirements on the design of building blocks.

Simulating only one (or a small number) of targets introduces deviations from the bulk behaviour of the system. Ouldridge *et al.*[159] demonstrated how self-assembly simulations of finite size in the canonical ensemble neglect important concentration fluctuations. When it is not possible to simulate a suitably large system, or use the grand canonical ensemble, the authors suggest applying the corrections they derived to rare-event simulation of the assembly of a single target. Using this method, yields for the bulk system may be obtained. This approach assumes that particles are ideal (*i.e.* they do not interact except as strictly needed on the pathway to self-assembly, and transition smoothly from the unbound to the bound state), which is not the case for the patchy cubes, where unintended interactions play a significant role. The properties of systems presented in this chapter will therefore deviate from bulk behaviour. However, in simulations that are compared directly, for example the assembly of the same target with a different value of n_{p}, the concentration of each component is constant. As a result these sets of simulations are conducted in a way that makes it reasonable to compare them.

Existing investigations of optimising addressable interactions have tended to use more idealised models, where the nature of interactions between building blocks has been overlooked in favour of optimising an interaction matrix.[96,97,160] Successfully designed binding interfaces for self-assembly have been demonstrated for proteins by King *et al.*.[155,156]

However, the approach to design presented in this chapter is quite distinct from theirs. In the case of proteins the task was to enhance the binding energy to ensure the desired target was the energetic ground state. Using the patchy cube model, a perfectly complementary binding partner can be achieved by patterning it with the mirror image of a patch design. Because the native interaction is guaranteed to be the ground state, optimisation consists of destabilising competing configurations, where particles are bound incorrectly, an example of negative design.[135,157]

The reduced performance of patch patterns that have been randomly generated, or perturbed from the optimal design, demonstrates that the design process is effective in avoiding kinetic traps due to incorrect binding. However, the approach adopted here only optimises the most general contributions to selective interactions. There is scope for further development by considering important secondary effects. For example, the algorithm only considers the distances between patches on the same face. Yet an important source of error in assembly was incorrect binding to the composite interfaces formed when building blocks join correctly. It is feasible that the optimisation algorithm could be refined to take into account inter-patch distances on these higher-order interfaces. Such a procedure would produce building blocks that are tailored to the particular target structure in question. Additionally, while the occurrence of similar inter-patch distances is suppressed, there is no consideration of which face the next-nearest distances appear on. So, while it is forbidden for the same distance to appear twice, two faces may share multiple similar distances without penalty.

The number of patches per face available to create a design can be seen as a measure of complexity of the building blocks. A simple model where complexity could be quantified was discussed in section 6.1.[100,133] In that work, building blocks with greater complexity were required to assemble less symmetric or modular targets. In the patchy cube model there is a similar relation where more complex designs are required to ensure the exclusivity of larger numbers of interfaces. Generally, increasing the number of patches and the size of the patch alphabet results in

- a greater number of exclusively interacting binding pairs,
- interfaces that are more robust to error in patch placement,
- self-assembly that is less sensitive to temperature,
- faster assembly of a target.

However, these benefits would come at the cost of a more complicated synthesis in an experimental system. An approach like the one presented in this chapter could therefore be helpful not only in establishing the limits of addressable assembly with designed interfaces, but also in determining the minimum complexity required for success. For example, $n_{\mathrm{p}} = 4$ is the best compromise between complexity and performance for assembling the Vitruvian man cluster. The additional complexity of an extra patch in the $n_{\mathrm{p}} = 5$ designs provides little improvement, while assembly is not very robust with three patches per face.

The optimisation procedure presented here aims to ensure a large difference between the energetic interactions in the native state and all competing states. Additional control over self-assembly may be gained by also considering the entropic contribution to the free energy of assembly.[3,5] For example, the internal flexibility of proteins is exploited in biological self-assembly.[161] The shape of building blocks also plays an important role in determining the outcome of self-assembly, as it does with other self-organising systems such as liquid crystals. Indeed, shape alone is in theory enough to drive addressable assembly[5] through lock-and-key type interactions.[38,50,51] However, relying on this type of interaction alone to create addressable interactions would require the production of many distinctly shaped particles. Shape complementarity could be used in conjunction with ss-DNA (or another set of specific energetic interactions) to enhance specificity of binding, or extend the number of addressable interactions.

6.5 Conclusion

Through simulating a variety of optimised designs within given bounds of complexity, this chapter has revealed a relationship between the addressability of patchy building blocks and the amount of information encoded in them. Designs of higher complexity (featuring more patches, a larger interaction alphabet, or both) are required to reliably create a large number of addressable interactions. The optimisation procedure, based on geometric arguments rather than simulation results, has been shown to be effective in producing unique interfaces. While the stability of complementary interactions are ensured by construction, a negative design approach is required to reduce the stability of the many possible competing states. Furthermore, the decrease in performance when designs are perturbed, demonstrates the sensitivity of assembly to both deviation from optimal design and imprecision in the synthesis of building blocks. This is an important consideration

for the potential application of the designs presented here to real systems, where errors in patch placement are inevitable.

Chapter 5 and this chapter tackled two distinct aspects of optimising addressable self-assembly. The former has focused on optimising the network and strength of bonds connecting an addressable cluster, while this chapter has investigated optimisation of the addressable interactions themselves. Both of these considerations are crucial in the design of self-assembling systems.

APPENDIX 6.A OBJECTIVE FUNCTION DERIVATIVES

To minimise equation 6.5 we need the derivatives of χ with respect to the x and y coordinates of each patch on each face. Application of the chain rule leads to the following terms for the derivative with respect to the x coordinate of patch ε on face k:

$$\begin{aligned}\frac{\partial\chi}{\partial x_\varepsilon^{(k)}} = -2d^2 \sum_{\substack{j=1\\j\neq k}}^{n_\mathrm{f}} \sum_{\substack{\beta=1\\\beta\neq\varepsilon}}^{n_\mathrm{p}} \sum_{\gamma=1}^{n_\mathrm{p}} \sum_{\delta<\gamma}^{n_\mathrm{p}} \left(\Delta_{\varepsilon\beta\gamma\delta}^{(kj)}\right)^{-3} \frac{\partial\Delta_{\varepsilon\beta\gamma\delta}^{(kj)}}{\partial x_\varepsilon^{(k)}} \\ - 2d^2 \sum_{\substack{\beta=1\\\beta\neq\varepsilon}}^{n_\mathrm{p}} \sum_{\substack{\gamma=1\\\gamma\neq\varepsilon}}^{n_\mathrm{p}} \sum_{\substack{\delta>\gamma\\\delta\neq\varepsilon}}^{n_\mathrm{p}} \left(\Delta_{\varepsilon\beta\gamma\delta}^{(kk)}\right)^{-3} \frac{\partial\Delta_{\varepsilon\beta\gamma\delta}^{(kk)}}{\partial x_\varepsilon^{(k)}} - 2d^2 \sum_{\substack{\beta=1\\\beta\neq\varepsilon}}^{n_\mathrm{p}} \sum_{\substack{\delta>\beta\\\delta\neq\varepsilon}}^{n_\mathrm{p}} \left(\Delta_{\varepsilon\beta\varepsilon\delta}^{(kk)}\right)^{-3} \frac{\partial\Delta_{\varepsilon\beta\varepsilon\delta}^{(kk)}}{\partial x_\varepsilon^{(k)}} \\ - \frac{N}{2r_0} \sum_{\substack{\beta=1\\\beta\neq\varepsilon}}^{n_\mathrm{p}} \left(\frac{2r_0}{r_{\varepsilon\beta}^{(k)}}\right)^{N+1} \frac{\partial r_{\varepsilon\beta}^{(k)}}{\partial x_\varepsilon^{(k)}} + M\left(x_\varepsilon^{(k)}\right)^{M-1} \left(\frac{1}{d/2-r_0}\right)^M . \quad (6.8)\end{aligned}$$

The derivatives with respect to a y coordinate have the same form, replacing each occurrence of $x_\varepsilon^{(k)}$ with $y_\varepsilon^{(k)}$.

Three of the derivatives in equation 6.8 simplify to the same result. In both $\Delta_{\varepsilon\beta\gamma\delta}^{(kj)}$ and $\Delta_{\varepsilon\beta\gamma\delta}^{(kk)}$ (which appear in the first and seconds terms of equation 6.8) only one difference of distances depends on $x_\varepsilon^{(k)}$, and so we may simplify these expressions:

$$\frac{\partial\Delta_{\varepsilon\beta\gamma\delta}^{(kj)}}{\partial x_\varepsilon^{(k)}} = \frac{\partial\Delta_{\varepsilon\beta\gamma\delta}^{(kk)}}{\partial x_\varepsilon^{(k)}} = \frac{\partial}{\partial x_\varepsilon^{(k)}} r_{\varepsilon\beta}^{(k)} . \quad (6.9)$$

This derivative also appears in the fourth term of equation 6.8 and evaluates to

$$\frac{\partial r_{\varepsilon\beta}^{(k)}}{\partial x_{\varepsilon}^{(k)}} = \frac{1}{r_{\varepsilon\beta}^{(k)}} \left(x_{\varepsilon}^{(k)} - x_{\beta}^{(k)} \right) . \tag{6.10}$$

The remaining derivative (third term in equation 6.8) features $\Delta_{\varepsilon\beta\varepsilon\delta}^{(kk)}$ in which both distances depend on $x_{\varepsilon}^{(k)}$ and so has two contributions:

$$\frac{\partial \Delta_{\varepsilon\beta\varepsilon\delta}^{(kk)}}{\partial x_{\varepsilon}^{(k)}} = \frac{1}{r_{\varepsilon\beta}^{(k)}} \left(x_{\varepsilon}^{(k)} - x_{\beta}^{(k)} \right) - \frac{1}{r_{\varepsilon\delta}^{(k)}} \left(x_{\varepsilon}^{(k)} - x_{\delta}^{(k)} \right) . \tag{6.11}$$

CHAPTER 7

CONCLUSIONS

7.1 GENERAL CONCLUSIONS AND OUTCOME OF AIMS

The aims of this thesis were set out in section 1.4. Here, we will consider the outcomes of these aims and how the preceding chapters have addressed them.

The patchy cube model has been used to study general aspects of self-assembly rather than the particular features of any one set of self-assembling particles. The ability to place patches arbitrarily on the faces of the particles and control the strength of interactions between them makes the model versatile. It may be used to create particles of varying complexity which represent different approaches to self-assembly, from simple components with high symmetry and non-specific interactions through to fully addressable sets like DNA bricks. The explicit representation of interactions therefore allows the patchy cube model to probe important aspects of self-assembly that some other, more idealised, models neglect. The patchy cube model therefore addresses aim 1.

The survey of the state of the art in simulation of self-assembly in chapter 2 began to address aim 2. Existing MC algorithms featuring cluster moves aim to produce the efficient equilibration of strongly associated systems. The VMMC, and later the SVMMC, algorithm went further, by trying to move aggregates in a dynamically meaningful manner. However, it was found that SVMMC did not move patchy cube aggregates of various sizes in a predictable way. In order to completely satisfy aim 2 it was therefore necessary to develop a new method to simulate our model. The hybrid algorithm introduced in chapter 3 takes an alternative approach to existing methods by separating the diffusion and relaxation of aggregates, and handling each with a different type of move. This allowed us to impose realistic, Stokes–Einstein-like dynamics to the bulk movement of aggregates while keeping the important collective relaxations of the SVMMC algorithm.

As noted in chapter 3 a compromise was made between the accuracy of the diffusion moves and the complexity of the hybrid MC algorithm. In the context of nanoscale particles, more accurate dynamics could be achieved by considering hydrodynamics factors in more detail. An example of this approach is given in the original article defining the VMMC algorithm. Here, the authors took a cross section of the aggregate in the plane parallel to the direction of motion,[114,115] which was then used to approximate the drag that would be experienced. In the VMMC algorithm unrealistically large moves would then be rejected. However, in the hybrid algorithm the move could be rescaled to give aggregates appropriate displacements in all directions.

Chapter 4 addressed aim 3 directly. Building blocks were designed by intuition to represent different self-assembly strategies. Single-component strategies, hierarchical assembly and a fully addressable approach were compared. As the geometry of the target, the number of particles it contained, and the physical conditions (reduced temperature and density) were the same for all strategies, the performance of each strategy could be contrasted. The single-component approach was best able to exploit the high symmetry of the target structure and produced assembly that was fast and robust with respect to changes in temperature compared to the other strategies. However, the fully addressable case could, under certain conditions, achieve higher yields, as the introduction of specific bonds destabilises many of the competing states that exist in the single-component case.

The placement of patches on an interface (and some control over their interactions) can be used to influence assembly pathways, which is important to ensuring the success of self-assembly. Currently, interactions between patchy colloids generally take place through single patches[58–60,63] rather than a larger designed interface, which is more commonly a feature of protein self-assembly. The fabrication of colloidal particles with interfaces like those of the patchy cubes could be a promising route to programmable colloidal assembly. However, it is not currently possible to decorate particles with this level of detail. It remains to be seen what level of complexity is achievable with advances in nanoparticle fabrication.

The prevalence of incompatible fragments in hindering the assembly of the addressable target in chapter 4 is a significant result. This kinetic trap was previously considered such a serious problem that it was assumed to be impossible to achieve acceptable yields using addressable DNA building blocks. The theoretical work of Jacobs *et al.* revealed that a late nucleation barrier prevented many small incompatible clusters of DNA bricks from forming.[83,89] The occurrence of this non-classical nucleation behaviour was found

to be dependent on the particular coordination number of DNA brick clusters. The observation in this work of incompatible fragments indicates that smaller addressable clusters, containing tens rather than hundreds of distinct components, with arbitrary coordination may not be able to avoid this problem. Chapter 5 investigated the relation between connectivity and the occurrence of these frustrated configurations. It was found that incompatible fragments could not form if the target cluster is connected by a linear network of bonds. However, the increased floppiness associated with a low number of links introduced new errors in assembly, and the reinforcement of correct assembly in highly connected structures was lost. Therefore, highly connected structures, with a greater number of loops in the network of bonds than strictly required in an idealised configuration, perform better over all. Chapter 5 therefore addressed aim 4. Subsequently, attention was turned to how bond strength may be used to control assembly pathways.

Heterogeneous bond energies were found to be beneficial to addressable assembly. In the context of aim 5, the results presented in chapter 5 are in agreement with the finding of Jacobs *et al.*,[83,89] and contradict earlier theoretical investigations that suggested homogeneous bond strengths were optimal.[96,98] In both this thesis and the work of Jacobs *et al.*, heterogeneous bond strengths guided assembly through pathways that avoided incompatible fragments. However, the nature of the pathways differed. In the case of Jacobs *et al.*, which focused on DNA brick assembly, the variation in binding energies benefited the formation of large pre-critical aggregates. These aggregates are not seen for the smaller clusters targeted in this thesis. Instead, the placement of strong bonds can be used to guide assembly through semi-hierarchical pathways. An important difference is that, for DNA brick assembly, a stochastic distribution of bond strengths is beneficial, whereas bonds need to be placed carefully in the smaller patchy cube clusters. Chapter 5 gave some understanding of how strong bonds should be placed to achieve reliable assembly, for example, strongly bound fragments should be linearly connected.

While ss-DNA sequences are currently ubiquitous in addressable assembly, chapter 6 showed that it is possible to design specifically interacting building blocks using a restricted set of addressable interactions, as set out by aim 6. The geometric distribution of patches on the faces of the patchy cube particles was used to create distinct interfaces, which should only bind strongly with their complementary mirror image. A negative design procedure is necessary in order to ensure that different patterns are dissimilar, and avoid unintended binding. Global optimisation of the interfaces was conducted using a variant of basin-hopping. Designs of greater complexity, featuring more patches or a

larger interaction alphabet, are able to access greater numbers of exclusive interfaces, and generally result in more robust and reliable assembly. However, in some cases greater complexity may result in diminishing returns in terms of the efficiency and reliability of assembly. Simulations of complementary dimers tested the limit of number of addressable interfaces that may be created, and example clusters provided a more realistic test on the limits of designs of a given complexity (aim 7). The simulations of addressable targets also revealed the best compromise between complexity of design and assembly performance. These interfaces can in principle be applied to physical particles, providing they may be patterned to a suitable accuracy, as deviation from the optimal designs result in degraded assembly quality.

7.2 OUTLOOK

Throughout this work particle design has only considered the energetic interactions between building blocks. Entropy has not been exploited directly to guide self-assembly. Nonetheless, distinct patchy interfaces with identical binding enthalpies can have significantly different associated entropies. For instance, if the patches on a face are placed closer together then the bound particles have greater flexibility to rotate about the bond axis because of a smaller torsion constant. Correspondingly, there is a smaller loss of entropy upon binding and so a more favourable binding free energy. This could be used to control the free energy landscape, making the free energy of binding greater for particular bonds. It could also exploited in more imaginative ways, for example, the conformational changes during a torsional fluctuation could be used to enhance or restrict the steric fit of a building block.

The internal flexibility of building blocks also plays a role in the entropic contribution to the free energy of assembly.[161] This is important in the assembly of protein clusters, where the large number of internal degrees of freedom in the proteins allows them to deform from their native structure in order to favour the binding interactions.[162] In the bound state a loss of entropy when proteins adopt a particular conformation to bind is offset by an enhanced binding energy. This leads to a modest free energy change, so that binding is not irreversible. A representation of internal flexibility could be implemented in the patchy cube model by making the patches mobile within some bounds. For instance, patch MC moves could be added, where an attempt is made to move a patch in a Gaussian well centred on the patches equilibrium position. If the ideal configuration

of patches in the bound state differs from the nominal positions in isolated particles it is possible to create a barrier to assembly, or an intermediate which relaxes to the bound state. Changes to the kinetics and pathway of assembly could further improve assembly of addressable clusters, where the rapid consumption of building blocks into small, incompatible aggregates needs to be avoided.

Beyond simply being flexible, some proteins also show a specific, functional response to binding in a phenomenon called allostery. After the substrate binds to an allosteric site a conformational change is induced in the protein. This change may, for example, enhance or subdue the biological action of the protein. A change in shape upon binding could be exploited in self-assembly to create strictly hierarchical assembly pathways. Consider, for instance, a building block where particular interactive sites are only accessible after a preliminary binding step. Control over multi-step assembly pathways would be more precise than with a static range of binding strengths as demonstrated in chapters 4 and 5. In the patchy cube model an allosteric-like response could be produced by introducing patches that are inactive until, or that move after, a certain binding condition has been met. A highly idealised model of this type has already been studied, where a hierarchical tile structure is assembled by turning on interactions for the next stage of assembly only when the previous step is completed.[163] There, it was found that this scheme was particularly effective in avoiding kinetic traps due to incompatible fragments due to the order of bond formation being strictly controlled. However, the trigger for initiating the next stage of assembly was rather artificial, and it is unclear how this could be implemented in practice as there is no known mechanism to switch interactions on and off in such an arbitrary manner.

The approach to designing specifically interacting interfaces in chapter 6 can be applied to cases where the number of unique interactions available is limited. This could be implemented through the use of a small number of unique ss-DNA strands or may not require DNA at all, but instead another relatively small set of unique interactions. Distinct patchy interfaces could therefore become a new way to create addressable interactions. Although particles of this type may currently not be created, progress in the fabrication of patchy particles has been rapid and new methods for bespoke synthesis are emerging.[164,165]

One source of error in assembly using the optimally designed interfaces was the incorrect binding of particles to the larger interfaces formed when particles bind correctly. This kinetic trap slowed assembly of the octamer cube at low temperatures, as this struc-

ture features many such higher-order interfaces. Designs could be further improved if the optimisation procedure considered the inter-patch distances on the larger interfaces within the target. This would require supplying the design algorithm with information about the geometry of the target, and which design is to applied to each face (and in what orientation) so that the relevant extra distances can be accounted for. The benefit of this approach could easily be tested by comparing the interfaces produced by the updated design procedure against those used in chapter 6. Hence, it might be possible to maximise the performance of building blocks of limited complexity by tailoring them to the intended target.

The most successful distributions of bond strengths in chapter 5 were designed by intuition. However, some general features of successful connectivities were revealed, such as the linear connectivity of hierarchical fragments. Therefore, it is possible, in principle, to develop an algorithm to design suitable distributions of links and bond strengths for target clusters of a given geometry. The bonding networks generated could then by used as an input to an interface design procedure similar to the one introduced in chapter 6. A small extension to that method could allow for different numbers of patches per face, and hence generate a set of optimally distinct, heterogeneous binding interfaces. The combination of these two design steps should reduce the occurrence of kinetic traps from both incompatible fragments, and the incorrect binding of particles.

The algorithm used to identify overlaps between particles, described in appendix 3.A, has been specialised for application to cubes. However, the more general SATOBB theory can be applied to boxes with orthogonal edges of arbitrary length. With a small modification the restriction of all particles being cubes could be removed. This would provide another form of assembly information and addressability where only certain particles may fit in their intended locations. The importance of building block geometry could also be addressed. In particular, building blocks with edge lengths of relative dimensions $2 \times 1 \times 1$ would resemble the shape of DNA bricks within clusters.[86] Such a model could address the role of binding geometry better than existing models that treat DNA bricks as spherical particles with a tetrahedral arrangement of patches.[88,90,91]

This thesis has established some general principles for the reliable and efficient self-assembly of discrete targets. The pathway to assembly features many kinetic traps, in particular, misbound building blocks and the formation of incompatible fragments in addressable systems. These traps have been addressed and techniques presented to minimise their hindrance of self-assembly. The success of assembly is a complex interplay of ef-

fects including connectivity, patch patterns and binding strength. Care has been taken to isolate these aspects so that each could be investigated, and optimised independently. Combination of the principles established here could improve the assembly of a wide range systems, and help reach increasingly complex targets. Furthermore, the computational tools developed in this work should continue to be useful as the field advances, particularly through designing and testing optimal interactions.

BIBLIOGRAPHY

[1] G. M. Whitesides and B. Grzybowski, *Science*, 2002, **295**, 2418–2421.

[2] K. J. M. Bishop, C. E. Wilmer, S. Soh and B. A. Grzybowski, *Small*, 2009, **5**, 1600–1630.

[3] F. A. Escobedo, *Soft Matter*, 2014, **10**, 8388–8400.

[4] G. van Anders, D. Klotsa, N. K. Ahmed, M. Engel and S. C. Glotzer, *Proc. Natl. Acad. Sci. U.S.A.*, 2014, **111**, E4812–E4821.

[5] D. Frenkel, *Nat. Mater.*, 2015, **14**, 9–12.

[6] M. Grzelczak, J. Vermant, E. M. Furst and L. M. Liz-Marzán, *ACS Nano*, 2010, **4**, 3591–3605.

[7] S. Kralj and D. Makovec, *ACS Nano*, 2015, **9**, 9700–9707.

[8] J. M. McBride and C. Avendano, *Soft Matter*, 2017, **13**, 2085–2098.

[9] J. Vermant and M. J. Solomon, *J. Phys.: Condens. Matter*, 2005, **17**, R187.

[10] A. Böker, J. He, T. Emrick and T. P. Russell, *Soft Matter*, 2007, **3**, 1231–1248.

[11] B. J. Park and D. Lee, *Soft Matter*, 2012, **8**, 7690–7698.

[12] W. Yang, Y. Yu, L. Wang, C. Yang and H. Li, *Nanoscale*, 2015, **7**, 2877–2882.

[13] M. Fialkowski, K. J. M. Bishop, R. Klajn, S. K. Smoukov, C. J. Campbell and B. A. Grzybowski, *J. Phys. Chem. B*, 2006, **110**, 2482–2496.

[14] H. Hess, *Soft Matter*, 2006, **2**, 669–677.

[15] J. Boekhoven, W. E. Hendriksen, G. J. M. Koper, R. Eelkema and J. H. van Esch, *Science*, 2015, **349**, 1075–1079.

[16] H. R. Vutukuri, Z. Preisler, T. H. Besseling, A. van Blaaderen, M. Dijkstra and W. T. S. Huck, *Soft Matter*, 2016, **12**, 9657–9665.

[17] B. A. Grzybowski and G. M. Whitesides, *J. Phys. Chem. B*, 2002, **106**, 1188–1194.

[18] J. Schwarz-Linek, C. Valeriani, A. Cacciuto, M. E. Cates, D. Marenduzzo, A. N. Morozov and W. C. K. Poon, *Proc. Natl. Acad. Sci. U.S.A.*, 2012, **109**, 4052–4057.

[19] K. Grünewald, P. Desai, D. C. Winkler, J. B. Heymann, D. M. Belnap, W. Baumeister and A. C. Steven, *Science*, 2003, **302**, 1396–1398.

[20] F. Lu, K. G. Yager, Y. Zhang, H. Xin and O. Gang, *Nat. Commun.*, 2015, **6**, 6912.

[21] S. Sacanna, L. Rossi and D. J. Pine, *J. Am. Chem. Soc.*, 2012, **134**, 6112–6115.

[22] P. W. K. Rothemund, *Nature*, 2006, **440**, 297–302.

[23] M. Rubenstein, A. Cornejo and R. Nagpal, *Science*, 2014, **345**, 795–799.

[24] M. G. Mateu, *Arch. Biochem. Biophys.*, 2013, **531**, 65–79.

[25] D. C. Rapaport, *Phys. Biol.*, 2010, **7**, 045001.

[26] M. R. Perkett and M. F. Hagan, *J. Chem. Phys.*, 2014, **140**, 214101.

[27] J. M. G. Llorente, J. Hernández-Rojas and J. Bretón, *Soft Matter*, 2014, **10**, 3560–3569.

[28] D. L. D. Caspar and A. Klug, *Cold Spring Harbor Symposium Quant. Biol.*, 1962, **27**, 1–24.

[29] D. L. Caspar, *Biophys. J.*, 1980, **32**, 103–138.

[30] J. E. Johnson and J. A. Speir, *J. Mol. Biol.*, 1997, **269**, 665–675.

[31] B. J. Alder and T. E. Wainwright, *J. Chem. Phys.*, 1957, **27**, 1208–1209.

[32] W. G. Hoover and F. H. Ree, *J. Chem. Phys.*, 1968, **49**, 3609–3617.

[33] M. Robles, M. López de Haro and A. Santos, *J. Chem. Phys.*, 2014, **140**, 136101.

[34] P. N. Pusey and W. van Megen, *Nature*, 1986, **320**, 340–342.

[35] S. C. Glotzer and M. J. Solomon, *Nat. Mater.*, 2007, **6**, 557–562.

[36] A. T. Cadotte, J. Dshemuchadse, P. F. Damasceno, R. S. Newman and S. C. Glotzer, *Soft Matter*, 2016, **12**, 7073–7078.

[37] P. F. Damasceno, M. Engel and S. C. Glotzer, *Science*, 2012, **337**, 453–457.

[38] S. Sacanna, D. J. Pine and G.-R. Yi, *Soft Matter*, 2013, **9**, 8096–8106.

[39] S.-Y. Zhang, M. D. Regulacio and M.-Y. Han, *Chem. Soc. Rev.*, 2014, **43**, 2301–2323.

[40] E. B. Mock, H. De Bruyn, B. S. Hawkett, R. G. Gilbert and C. F. Zukoski, *Langmuir*, 2006, **22**, 4037–4043.

[41] D. Nagao, K. Goto, H. Ishii and M. Konno, *Langmuir*, 2011, **27**, 13302–13307.

[42] J. R. Wolters, G. Avvisati, F. Hagemans, T. Vissers, D. J. Kraft, M. Dijkstra and W. K. Kegel, *Soft Matter*, 2015, **11**, 1067–1077.

[43] G. Avvisati and M. Dijkstra, *Soft Matter*, 2015, **11**, 8432–8440.

[44] Y. Sun and Y. Xia, *Science*, 2002, **298**, 2176–2179.

[45] L. Carbone, C. Nobile, M. De Giorgi, F. D. Sala, G. Morello, P. Pompa, M. Hytch, E. Snoeck, A. Fiore, I. R. Franchini, M. Nadasan, A. F. Silvestre, L. Chiodo, S. Kudera, R. Cingolani, R. Krahne and L. Manna, *Nano Lett.*, 2007, **7**, 2942–2950.

[46] S. Asakura and F. Oosawa, *J. Chem. Phys.*, 1954, **22**, 1255–1256.

[47] Y. Mao, M. E. Cates and H. N. W. Lekkerkerker, *Physica A*, 1995, **222**, 10–24.

[48] A. S. Karas, J. Glaser and S. C. Glotzer, *Soft Matter*, 2016, **12**, 5199–5204.

[49] D. J. Kraft, R. Ni, F. Smallenburg, M. Hermes, K. Yoon, D. A. Weitz, A. van Blaaderen, J. Groenewold, M. Dijkstra and W. K. Kegel, *Proc. Natl. Acad. Sci. U.S.A.*, 2012, **109**, 10787–10792.

[50] S. Sacanna, W. T. M. Irvine, P. M. Chaikin and D. J. Pine, *Nature*, 2010, **464**, 575–578.

[51] S. Sacanna, M. Korpics, K. Rodriguez, L. Colón-Meléndez, S.-H. Kim, D. J. Pine and G.-R. Yi, *Nat. Commun.*, 2013, **4**, 1688.

[52] H.-G. Liao, D. Zherebetskyy, H. Xin, C. Czarnik, P. Ercius, H. Elmlund, M. Pan, L.-W. Wang and H. Zheng, *Science*, 2014, **345**, 916–919.

[53] H.-W. Song, N.-Y. Kim, J.-e. Park, J.-H. Ko, R. J. Hickey, Y.-H. Kim and S.-J. Park, *Nanoscale*, 2017, **9**, 2732–2738.

[54] L. Rossi, S. Sacanna, W. T. M. Irvine, P. M. Chaikin, D. J. Pine and A. P. Philipse, *Soft Matter*, 2011, **7**, 4139–4142.

[55] H. R. Vutukuri, F. Smallenburg, S. Badaire, A. Imhof, M. Dijkstra and A. van Blaaderen, *Soft Matter*, 2014, **10**, 9110–9119.

[56] F. Smallenburg, L. Filion, M. Marechal and M. Dijkstra, *Proc. Natl. Acad. Sci. U.S.A.*, 2012, **109**, 17886–17890.

[57] D. Frenkel, *Proc. Natl. Acad. Sci. U.S.A.*, 2012, **109**, 17728–17729.

[58] A. B. Pawar and I. Kretzschmar, *Macromol. Rapid Commun.*, 2010, **31**, 150–168.

[59] E. Bianchi, R. Blaak and C. N. Likos, *Phys. Chem. Chem. Phys.*, 2011, **13**, 6397–6410.

[60] G.-R. Yi, D. J. Pine and S. Sacanna, *J. Phys.: Condens. Matter*, 2013, **25**, 193101.

[61] Z. Zhang, A. S. Keys, T. Chen and S. C. Glotzer, *Langmuir*, 2005, **21**, 11547–11551.

[62] Q. Chen, S. C. Bae and S. Granick, *Nature*, 2011, **469**, 381–384.

[63] Y. Wang, Y. Wang, D. R. Breed, V. N. Manoharan, L. Feng, A. D. Hollingsworth, M. Weck and D. J. Pine, *Nature*, 2012, **491**, 51–55.

[64] W. Liu, J. Halverson, Y. Tian, A. V. Tkachenko and O. Gang, *Nat. Chem.*, 2016, **8**, 867–873.

[65] N. Kern and D. Frenkel, *J. Chem. Phys.*, 2003, **118**, 9882–9889.

[66] E. Bianchi, G. Doppelbauer, L. Filion, M. Dijkstra and G. Kahl, *J. Chem. Phys.*, 2012, **136**, 214102.

[67] T. Vissers, Z. Preisler, F. Smallenburg, M. Dijkstra and F. Sciortino, *J. Chem. Phys.*, 2013, **138**, 164505.

[68] T. Vissers, F. Smallenburg, G. Munaò, Z. Preisler and F. Sciortino, *J. Chem. Phys.*, 2014, **140**, 144902.

[69] Z. Preisler, T. Vissers, F. Smallenburg and F. Sciortino, *J. Chem. Phys.*, 2016, **145**, 064513.

[70] A. W. Wilber, J. P. K. Doye, A. A. Louis, E. G. Noya, M. A. Miller and P. Wong, *J. Chem. Phys.*, 2007, **127**, 085106.

[71] A. W. Wilber, J. P. K. Doye, A. A. Louis and A. C. F. Lewis, *J. Chem. Phys.*, 2009, **131**, 175102.

[72] A. W. Wilber, J. P. K. Doye and A. A. Louis, *J. Chem. Phys.*, 2009, **131**, 175101.

[73] G. Villar, A. W. Wilber, A. J. Williamson, P. Thiara, J. P. K. Doye, A. A. Louis, M. N. Jochum, A. C. F. Lewis and E. D. Levy, *Phys. Rev. Lett.*, 2009, **102**, 118106.

[74] Z. Zhang and S. C. Glotzer, *Nano Lett.*, 2004, **4**, 1407–1413.

[75] D. C. Rapaport, *Phys. Rev. Lett.*, 2008, **101**, 186101.

[76] M. R. Jones, N. C. Seeman and C. A. Mirkin, *Science*, 2015, **347**, 1260901.

[77] C. A. Mirkin, R. L. Letsinger, R. C. Mucic and J. J. Storhoff, *Nature*, 1996, **382**, 607–609.

[78] P. L. Biancaniello, A. J. Kim and J. C. Crocker, *Phys. Rev. Lett.*, 2005, **94**, 058302.

[79] A. J. Kim, P. L. Biancaniello and J. C. Crocker, *Langmuir*, 2006, **22**, 1991–2001.

[80] B. Nickel and T. Liedl, *Nat. Mater.*, 2015, **14**, 746–749.

[81] M. N. O'Brien, M. R. Jones, B. Lee and C. A. Mirkin, *Nat. Mater.*, 2015, **14**, 833–839.

[82] S. C. Glotzer, *Science*, 2004, **306**, 419–420.

[83] W. M. Jacobs and D. Frenkel, *J. Am. Chem. Soc.*, 2016, **138**, 2457–2467.

[84] L. Cademartiri and K. J. M. Bishop, *Nat. Mater.*, 2015, **14**, 2–9.

[85] B. Wei, M. Dai and P. Yin, *Nature*, 2012, **485**, 623–626.

[86] Y. Ke, L. L. Ong, W. M. Shih and P. Yin, *Science*, 2012, **338**, 1177–1183.

[87] Y. Ke, L. L. Ong, W. Sun, J. Song, M. Dong, W. M. Shih and P. Yin, *Nat. Chem.*, 2014, **6**, 994–1002.

[88] A. Reinhardt and D. Frenkel, *Phys. Rev. Lett.*, 2014, **112**, 238103.

[89] W. M. Jacobs, A. Reinhardt and D. Frenkel, *Proc. Natl. Acad. Sci. U.S.A.*, 2015, **112**, 6313–6318.

[90] A. Reinhardt and D. Frenkel, *Soft Matter*, 2016, **12**, 6253–6260.

[91] H. K. Wayment-Steele, D. Frenkel and A. Reinhardt, *Soft Matter*, 2017, **13**, 1670–1680.

[92] W. M. Jacobs, A. Reinhardt and D. Frenkel, *J. Chem. Phys.*, 2015, **142**, 021101.

[93] N. Arkus, V. N. Manoharan and M. P. Brenner, *Phys. Rev. Lett.*, 2009, **103**, 118303.

[94] G. Meng, N. Arkus, M. P. Brenner and V. N. Manoharan, *Science*, 2010, **327**, 560–563.

[95] D. J. Wales and J. P. K. Doye, *J. Phys. Chem. A*, 1997, **101**, 5111–5116.

[96] S. Hormoz and M. P. Brenner, *Proc. Natl. Acad. Sci. U.S.A.*, 2011, **108**, 5193–5198.

[97] Z. Zeravcic, V. N. Manoharan and M. P. Brenner, *Proc. Natl. Acad. Sci. U.S.A.*, 2014, **111**, 15918–15923.

[98] L. O. Hedges, R. V. Mannige and S. Whitelam, *Soft Matter*, 2014, **10**, 6404–6416.

[99] T. K. Haxton and S. Whitelam, *Soft Matter*, 2013, **9**, 6851–6861.

[100] I. G. Johnston, S. E. Ahnert, J. P. K. Doye and A. A. Louis, *Phys. Rev. E*, 2011, **83**, 066105.

[101] D. J. Wales, *Energy Landscapes*, Cambridge University Press, 2003.

[102] B. S. John, A. Stroock and F. A. Escobedo, *J. Chem. Phys.*, 2004, **120**, 9383–9389.

[103] B. S. John and F. A. Escobedo, *J. Phys. Chem. B*, 2005, **109**, 23008–23015.

[104] E. Sanz and D. Marenduzzo, *J. Chem. Phys.*, 2010, **132**, 194102.

[105] F. Romano, C. D. Michele, D. Marenduzzo and E. Sanz, *J. Chem. Phys.*, 2011, **135**, 124106.

[106] N. Metropolis and S. Ulam, *J. Am. Stat. Assoc.*, 1949, **44**, 335–341.

[107] M. E. J. Newman and G. T. Barkema, *Monte Carlo Methods in Statistical Physics*, Oxford University Press, 1999.

[108] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller and E. Teller, *J. Chem. Phys.*, 1953, **21**, 1087–1092.

[109] M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids*, Oxford Science Publications, 1989.

[110] D. Frenkel and B. Smit, *Understanding Molecular Simulation From Algorithms to Applications*, Academic Press, 2nd edn., 2002.

[111] R. H. Swendsen and J.-S. Wang, *Phys. Rev. Lett.*, 1987, **58**, 86–88.

[112] U. Wolff, *Phys. Rev. Lett.*, 1989, **62**, 361–364.

[113] A. Troisi, V. Wong and M. A. Ratner, *J. Chem. Phys.*, 2004, **122**, 024102.

[114] S. Whitelam and P. L. Geissler, *J. Chem. Phys.*, 2007, **127**, 154101.

[115] S. Whitelam and P. L. Geissler, *J. Chem. Phys.*, 2008, **128**, 219901.

[116] S. Whitelam, E. H. Feng, M. F. Hagan and P. L. Geissler, *Soft Matter*, 2009, **5**, 1251–1262.

[117] C. Matek, T. E. Ouldridge, A. Levy, J. P. K. Doye and A. A. Louis, *J. Phys. Chem. B*, 2012, **116**, 11616–11625.

[118] J. P. K. Doye, D. J. Wales and M. A. Miller, *J. Chem. Phys.*, 1998, **109**, 8143–8153.

[119] J. Janin, R. P. Bahadur and P. Chakrabarti, *Q. Rev. Biophys.*, 2008, **41**, 133–180.

[120] J. Zhang, A. Kumbhar, J. He, N. C. Das, K. Yang, J.-Q. Wang, H. Wang, K. L. Stokes and J. Fang, *J. Am. Chem. Soc.*, 2008, **130**, 15203–15209.

[121] S. Gottschalk, M. C. Lin and D. Manocha, in *Proceedings of ACM Siggraph*, 1996, pp. 171–180.

[122] B. S. John, A. Stroock and F. A. Escobedo, *J. Chem. Phys.*, 2004, **120**, 9383–9389.

[123] M. F. Hagan, O. M. Elrad and R. L. Jack, *J. Chem. Phys.*, 2011, **135**, 104115.

[124] J. Grant, R. L. Jack and S. Whitelam, *J. Chem. Phys.*, 2011, **135**, 214505.

[125] D. Klosta and R. L. Jack, *Soft Matter*, 2011, **7**, 6294–6303.

[126] D. Morphew and D. Chakrabarti, *Soft Matter*, 2016, **12**, 9633–9640.

[127] Y. He, T. Ye, M. Su, C. Zhang, A. E. Ribbe, W. Jiang and C. Mao, *Nature*, 2008, **452**, 198–201.

[128] J. A. Fallas, L. E. R. O'Leary and J. D. Hartgerink, *Chem. Soc. Rev.*, 2010, **39**, 3510–3527.

[129] A. H. Gröschel, A. Walther, T. I. Löbling, F. H. Schacher, H. Schmalz and A. H. E. Müller, *Nature*, 2013, **503**, 247–251.

[130] B. Chen and R. Tycko, *Biophys. J.*, 2011, **100**, 3035–3044.

[131] M. Tsiang, A. Niedziela-Majka, M. Hung, D. Jin, E. Hu, S. Yant, D. Samuel, X. Liu and R. Sakowicz, *Biochemistry*, 2012, **51**, 4416–4428.

[132] W. M. Jacobs, D. W. Oxtoby and D. Frenkel, *J. Chem. Phys.*, 2014, **140**, 204109.

[133] S. E. Ahnert, I. G. Johnston, T. M. A. Fink, J. P. K. Doye and A. A. Louis, *Phys. Rev. E*, 2010, **82**, 026117.

[134] A. Sadeghi, S. A. Ghasemi, B. Schaefer, S. Mohr, M. A. Lill and S. Goedecker, *J. Chem. Phys.*, 2013, **139**, 184118.

[135] J. S. Richardson and D. C. Richardson, *Trends Biochem. Sci.*, 1989, **14**, 304–309.

[136] S. Auer, C. M. Dobson and M. Vendruscolo, *HFSP J.*, 2007, **1**, 137–146.

[137] S. Auer, F. Meersman, C. M. Dobson and M. Vendruscolo, *PLOS Comput. Biol.*, 2008, **4**, e1000222.

[138] A. Zlotnick, R. Aldrich, J. M. Johnson, P. Ceres and M. J. Young, *Virology*, 2000, **277**, 450–456.

[139] J. E. Baschek, H. C. R Klein and U. S. Schwarz, *BMC Biophys.*, 2012, **5**, 22.

[140] K. V. Gothelf, *Science*, 2012, **338**, 1159–1160.

[141] P. W. K. Rothemund and E. S. Andersen, *Nature*, 2012, **485**, 584–585.

[142] S. Whitelam and R. L. Jack, *Annu. Rev. Phys. Chem.*, 2015, **66**, 143–163.

[143] *GLPK (GNU Linear Programming Kit)*, `https://www.gnu.org/software/glpk/`, (accessed September 2017).

[144] *PuLP*, `https://github.com/coin-or/pulp`, (accessed September 2017).

[145] W. B. Rogers and V. N. Manoharan, *Science*, 2015, **347**, 639–642.

[146] I. Coluzza, P. D. J. van Oostrum, B. Capone, E. Reimhult and C. Dellago, *Soft Matter*, 2012, **9**, 938–944.

[147] I. Coluzza and D. Frenkel, *Biophys. J.*, 2007, **92**, 1150–1156.

[148] B. Srinivasan, T. Vo, Y. Zhang, O. Gang, S. Kumar and V. Venkatasubramanian, *Proc. Natl. Acad. Sci. U.S.A.*, 2013, **110**, 18431–18435.

[149] J. Qin, G. S. Khaira, Y. Su, G. P. Garner, M. Miskin, H. M. Jaeger and J. J. de Pablo, *Soft Matter*, 2013, **9**, 11467.

[150] M. Z. Miskin, G. Khaira, J. J. de Pablo and H. M. Jaeger, *Proc. Natl. Acad. Sci. U.S.A.*, 2016, **113**, 34–39.

[151] B. A. Lindquist, R. B. Jadrich and T. M. Truskett, *J. Chem. Phys.*, 2016, **145**, 111101.

[152] E. I. Shakhnovich, *Phys. Rev. Lett.*, 1994, **72**, 3907–3910.

[153] I. Coluzza, *J. Phys. Cond. Matt.*, 2017, **29**, 143001.

[154] F. DiMaio, A. Leaver-Fay, P. Bradley, D. Baker and I. André, *PLOS ONE*, 2011, **6**, e20450.

[155] N. P. King, W. Sheffler, M. R. Sawaya, B. S. Vollmar, J. P. Sumida, I. André, T. Gonen, T. O. Yeates and D. Baker, *Science*, 2012, **336**, 1171–1174.

[156] N. P. King, J. B. Bale, W. Sheffler, D. E. McNamara, S. Gonen, T. Gonen, T. O. Yeates and D. Baker, *Nature*, 2014, **510**, 103–108.

[157] I. N. Berezovsky, K. B. Zeldovich and E. I. Shakhnovich, *PLOS Computat. Biol.*, 2007, **3**, e52.

[158] R. Fletcher and C. M. Reeves, *Comput. J.*, 1964, **7**, 149–154.

[159] T. E. Ouldridge, A. A. Louis and J. P. K. Doye, *J. Phys. Condens. Matter*, 2010, **22**, 104102.

[160] Z. Zeravcic and M. P. Brenner, *Proc. Natl. Acad. Sci. U.S.A.*, 2014, **111**, 1748–1753.

[161] D. D. Boehr and P. E. Wright, *Science*, 2008, **320**, 1429–1430.

[162] G. R. Lazaro and M. F. Hagan, *J. Phys. Chem. B*, 2016, **120**, 6306–6318.

[163] S. Whitelam, *Soft Matter*, 2015, **11**, 8225–8235.

[164] S. Ni, J. Leemann, H. Wolf and L. Isa, *Farady Discuss.*, 2015, **181**, 225–242.

[165] S. Ni, J. Leemann, I. Buttinoni, L. Isa and H. Wolf, *Sci. Adv.*, 2016, **2**, e1501779.