Controlled formation of nanostructures with desired geometries: Robust dynamic paths to robust desired structures

by

Earl Osman P. Solis

B.S., Carnegie Mellon University (2003)
M.S., Massachusetts Institute of Technology (2004)

Submitted to the Department of Chemical Engineering
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2009

Author .....
Department of Chemical Engineering
28 May 2009

Certified by .....
George Stephanopoulos
Arthur D. Little Professor of Chemical Engineering
Thesis Supervisor

Certified by .....
Paul I. Barton
Lammot du Pont Professor of Chemical Engineering
Thesis Supervisor

Accepted by .....
William M. Deen
Carbon P. Dubbs Professor of Chemical Engineering
Chairman, Department Committee on Graduate Theses

Controlled formation of nanostructures with desired geometries: Robust dynamic paths to robust desired structures

by

Earl Osman P. Solis

Submitted to the Department of Chemical Engineering
on 28 May 2009, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

An essential requirement for the fabrication of future electronic, magnetic, optical and biologically-based devices, composed of constituents at the nanometer length scale, is the precise positioning of the components in the system's physical domain. We introduce the design principles, problems and methods associated with the controlled formation of nanostructures with desired geometries through a hybrid top-down and bottom-up approach: top-down formation of physical domains with externally-imposed controls and bottom-up generation of the desired structure through the self-assembly of the nanoscale constituents, driven by interparticle interactions (short- and long-range) and interactions with the external controls (e.g., electrical, magnetic, chemical). The desired nanoscale structure must be locally stable and robust to a desired level of robustness, and it should be reachable from any initial particle distribution. These two requirements frame the two elements of the design problem: (a) Static Problem: Systematic placement of externally imposed controls and determination of their intensities in order to ensure that the final desired structure is stable with a desired degree of robustness; (b) Dynamic Problem: Time-varying controls in order to ensure that the desired final structure can be reached from any initial particle distribution.

The concept of external controls is realized through point conditions, which introduce attractive or repulsive interaction terms in the system potential energy. The locations of the point conditions are found through the solution of a minimum tiling problem. Given these locations, the Static Problem is solved through the solution of combinatorially-constrained optimization problems. The Dynamic Problem is solved through a genetic algorithm search for the appropriate time-varying system degrees of freedom. Crucial to the achievement of the design goals is the necessity to break the ergodicity of the system phase space and control the subset of system states accessible to the system. More specifically, the static approach requires isolating the desired structure from all competing structures in phase space. The dynamic approach in-

volves a multiresolution view of the system particle number, where we successively restrict the accessible volume of phase space based on coarse-grained particle number (i.e., density) specifications. We illustrate the design problems and solution methods through 1- and 2-dimensional lattice models and simulate the self-assembly process with a dynamic Monte Carlo method.

Thesis Supervisor: George Stephanopoulos
Title: Arthur D. Little Professor of Chemical Engineering

Thesis Supervisor: Paul I. Barton
Title: Lammot du Pont Professor of Chemical Engineering

Acknowledgments

Six years went by. In this time, a young student whose expertise was mainly in solving "textbook" problems learned how to approach and tackle more open-ended and pertinent engineering problems (or so he hopes). The transition from learning the theory and tools needed in the chemical engineering discipline to applying them to the challenging questions posed by today's scientific community is not a smooth one, and one must acknowledge the guidance and support of mentors, colleagues, friends and family. To this end, here I go...

I begin with Professor George Stephanopoulos. His support and encouragement pushed me to be more aware, creative and productive. His guidance and patience, especially during the challenging periods of scientific discovery, have undoubtedly molded me as a researcher. Professor Stephanopoulos provided me with a unique graduate education, guiding me through an open-ended research project, while also giving me the freedom to try different approaches and learn from mistakes. I will always remember the brainstorming sessions on the blackboard in Professor Stephanopoulos' office.

Professor Paul Barton played a significant role in educating me about the practical sides of research, e.g., determining whether a problem is solvable, using the appropriate solution methods. His guidance with optimization problem formulations and solution methods were crucial to the completion of my thesis work. His attention to detail and constructive feedback is a testament to his success as an academic researcher and advisor.

I would also like to thank the other committee members, Professors Klavs Jensen and Arup Chakraborty, for their feedback throughout the development of my thesis project.

This work could not have been done without the computational support of both the Process Systems Engineering Laboratory (PSEL), under the guidance of Professor Barton, and the Tester Group. I would also like to thank Christopher Marton for his help with the initial investigations of the Dynamic Problem.

The many hours spent in Building 66 undoubtedly forges strong friendships within the graduate student body of the Chemical Engineering Department. I would like to thank Scott and Yumi Paap, Mahriah Alf, Ingrid Berkelmans Fox, Lily Tong, and my roommates, Andrew Peterson and Sandeep Sharma, for their encouragement and friendship.

During my six years at MIT, I was involved in few activities outside of research, which undeniably salvaged my sanity. The Thirsty Ear Pub was my office outside the office. I would like to thank the staff, especially Sara Cinnamon and Michael Grenier, for their support and friendship. Thursday nights will never be the same without karaoke and red wine in the basement of a graduate student dormitory. I would also like to thank the members of MIT's Dance Troupe for providing me a creative outlet for my love of dance. Specifically, I would like to thank the dancers who worked with me during my 3 choreography attempts. "This is love..."

I would not be here without 27+ years of love and support from my parents, Achilles and Carmelita Solis, and sister, Mahalia Solis Ong.

Finally, I would like to thank the Mitsubishi Chemical Holding Corporation for financially supporting my thesis research.

ES

28 May 2009

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The fabrication of structures with desired nanoscale geometric features is a core requirement for the manufacturing of future electronic, magnetic, and optical devices, composed of nanoscale particles or blocks of particles, e.g., nanoelectronic circuits, high-sensitivity sensors, molecular computers, molecular-scale factories, synthetic cells, adaptive devices (e.g., artificial tissues and sensorial systems, scalable plasmonic devices, chemico-mechanical processing, nanodevices and targeted cell therapy, human-machine interfaces at the tissue and nervous system level)[1]. The theory and practice of forming closely-packed 2-dimensional films and 3-dimensional materials with desired periodic nanoscale geometries in an essentially infinite domain have advanced significantly during the last 10-15 years. For example, a large variety of self-assembled monolayers, leading to highly structured films on surfaces that provide biocompatibility, control of corrosion, friction, wetting, and adhesion, have been experimentally synthesized and theoretically analyzed[2]. These films are viewed as possible precursors to nanometer-scale devices for use in organic microelectronics. However, their geometries are essentially periodic, which could constitute an important limitation. The features of phase-separated regions of block copolymers and blends are often of nanometer scale, periodic and dense, and can be rationally designed through the judicious selection of the monomers, and the length and frequency of blocks in the polymeric chain(s)[3, 4]. Furthermore, templated self-assembly techniques have been extensively developed to produce nanostructures with

desired geometries through the judicious selection of nanoparticles and features of the system environment. Examples include crystallization on template surfaces that determine the morphology of the resulting crystals[5] and crystallization of colloids in optical fields [6]. The templates could be physical (capillary forces, spin-coating, surface steps, and others), molecular (patterned self-assembled monolayers with specific chemical functionalization of terminal groups), or electrostatic (localized charges on the surface of a substrate). The resolution of the structural features that can be achieved by template-based approaches depends on the spatial resolution of templates. DNA-programmed placement using 2-dimensional DNA crystals as scaffolds, placement using electrophoresis, and focused placement, which uses focusing mechanisms to guide the nanoscale particles to specific locations in the physical domain that are smaller in scale than the template guiding them, are additional approaches that have been extensively reviewed by Koh[7].

Major challenges remain in two areas: (i) the fabrication of non-close-packed materials with asymmetric structures, and (ii) creation of composite systems through the precise positioning of individual functional units. In both cases the fundamental problem is how to place the individual elements (e.g., nanoparticles, nanowires, nanotubes, fragments of DNA, oligomers, proteins) at precise positions in a physical domain so that they are connected to each other to form a complex structure of desired geometry and are also connected to the outside world with which they interact. As an example, let us use a lattice model to represent a system of N particles within a system volume V. Figure 1-1 shows different categories of desired self-assembled structures for such a model system. Judiciously designed nanoparticles can self-assemble and form the periodic structures in Figure 1-1(a), but to form non-periodic nanostructures like those in Figure 1-1(b) external controls are needed to guide the self-assembly process. The size of the physical domains may be defined through a top-down fabrication technique such as photolithography[8, 9] or nanoimprinting[10, 11, 12] and could produce domains with dimensions of ∼50nm or larger. External controls, which may include electric or magnetic fields, chemical functionalizations, could be positioned through various top-down fabrication techniques,

e.g., 2 nm diameter electrodes made with electron beam lithography[13], carbon nanotubes as electrodes (1-5 nm diameter)[14], etc. The particles may have charges, dipoles, long-range and/or short-range interactions with each other, specific affinities (e.g., "patchy" particles[15], DNA molecules[16, 17, 18]), etc. The resolution in the distribution of controls is limited by the physics of the fabrication technique. For example, electron beam lithography can generate arrays of electrodes 2 nm in diameter with about 20-50 nm between electrodes[13]. Thus, in selecting the location of external controls we must always conform with these physical constraints.

New approaches are needed to design and construct non-periodic and non- closed-packed nanostructures systematically, and this need defines the scope of the work presented in this thesis. More specifically, the following design questions are addressed: (a) *Static Problem*: what are the optimal external controls so that the desired nanostructure is stable with a desired degree of robustness? (b) *Dynamic Problem*: how do we change the external controls over time in order to ensure that the desired final structure can be reached from any initial distribution of the particles in the physical domain? The first problem is addressed in Chapter 3, and the second problem in Chapter 4. However, before we can address these problems, we provide some background on statistical mechanics considerations (Section 1.1), lattice models for self-assembling particle systems (Chapter 2), and Monte Carlo (MC) simulation techniques (Section 2.2).

1.1 Statistical mechanics

Statistical mechanics is a probabilistic approach to the thermodynamic principles of chemical systems and is traditionally considered a molecular understanding of systems in the thermodynamic limit, i.e., $N \to \infty, V \to \infty, \frac{N}{V} = \nu$, where the specific volume ν is finite. We cite here a few statistical mechanics textbooks[19, 20, 21, 22] that provide an excellent background on the principles of this field.

The first law of thermodynamics states that both work and heat are forms of energy, and that the total energy is conserved. This is summarized by the fundamental

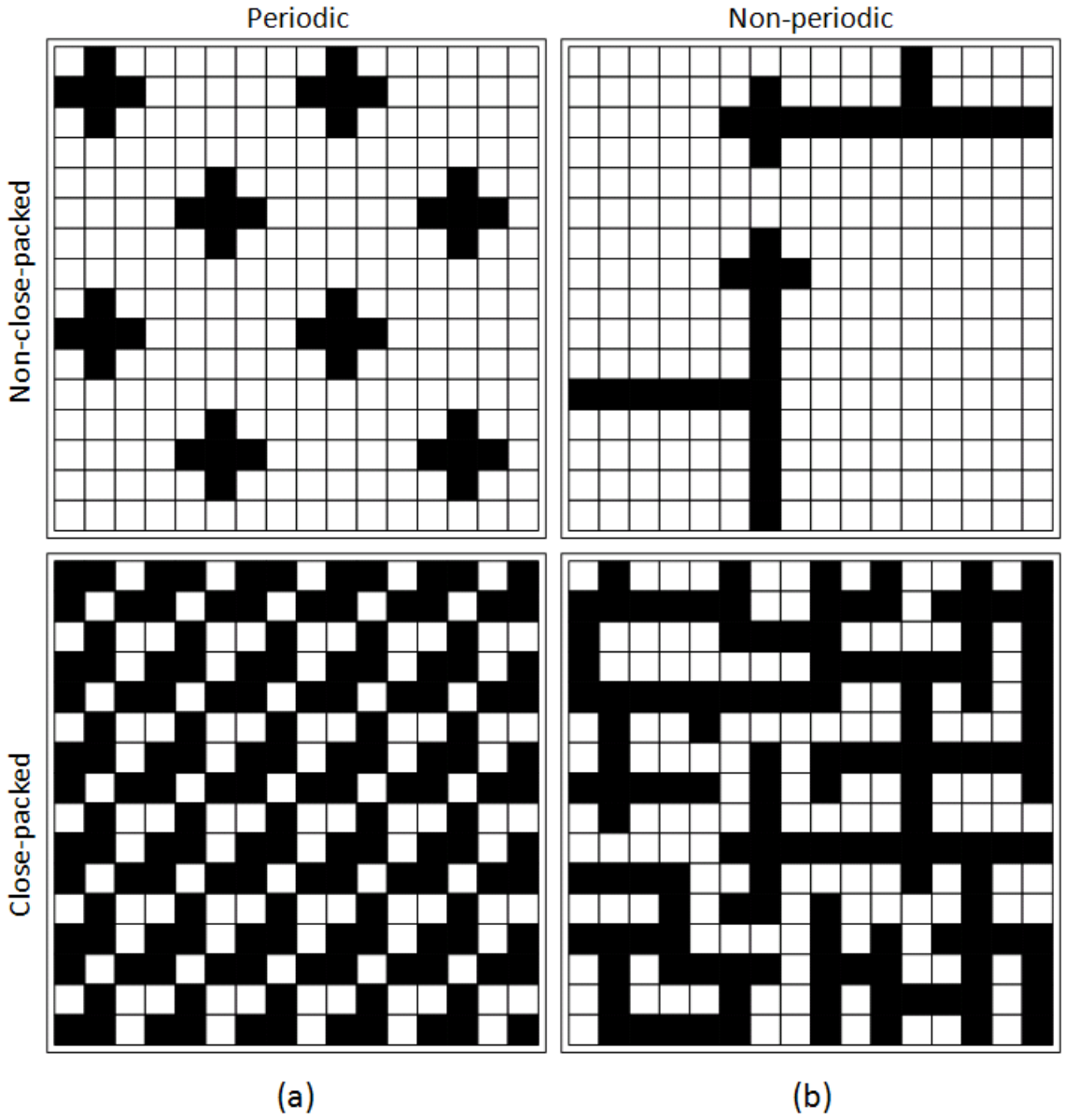

Figure 1-1: The periodic nanostructures in (a) can be achieved through self-assembly of judiciously designed nanoparticles, but the non-periodic structures in (b) require external controls for guided self-assembly.

equation:

$$E = TS + \mathbf{J} \cdot \mathbf{x} + \mu \cdot \mathbf{N}, \tag{1.1}$$

where $\mathbf{J}$ represents generalized forces, e.g., pressure $(-P)$, and $\mathbf{x}$ represents generalized displacements, e.g., volume (V), and μ is the chemical potential. This equation tell us that a system is fully specified by 3 variables, one from each term in the above sum. The most common system types, or ensembles, are the following: (1) N, V, E (microcanonical); (2) N, V, T (canonical); (3) μ, V, T (grand canonical); (4) N, P, T (Gibb's canonical). These systems are equivalent in the thermodynamic limit, but this equivalency breaks down for finite systems (see Section 1.1.1). Hence, for nanoscale systems with finite N and V, each system type must be understood separately.

Thus far, we have described energy as a thermodynamic quantity. At the microscopic level, we know that systems are composed of smaller constituents (e.g., particles, molecules, atoms), whose interactions and dynamics are reasonably well-understood in terms of more fundamental theories. At any time, t, the microstate of a system of N particles is described by specifying the position, $\mathbf{q}(t)$, and momentum, $\mathbf{p}(t)$, of all the system constituents. Thus, the microstate, $m(t) = \prod_{i=1}^{N} \{q_i(t), p_i(t)\}$. The classical energy function for a particle system at a particular microstate can be divided into separable potential and kinetic energy terms:

$$E(m) = E_{PE}(\mathbf{q}) + E_{KE}(\mathbf{p}), \tag{1.2}$$

where it is assumed that the potential energy is only a function of position and the kinetic energy is only a function of momentum.

The model systems described in Chapter 2 specify N, V and T, and are therefore canonical in nature. The probability distribution function (pdf) of a particular microstate, m_j, in the canonical prescription is given by the Boltzmann equation:

$$p(m_j) = \frac{e^{-\beta E(m_j)}}{\sum_k e^{-\beta E(m_k)}} = Z^{-1} e^{-\beta E(m_j)}, \tag{1.3}$$

where Z is called the normalizing partition function:

$$Z = \sum_k e^{-\beta E(m_k)}. \tag{1.4}$$

Equation 1.3 is an ensemble probability, i.e., $p(m_j)$ represents the fraction of an ensemble of systems that are in microstate m_j. It also represents the probability of finding a representative system in state m_j after the system has reached equilibrium. The energy parameter, β, essentially determines the "flow" of the system through phase space. The smaller the β value, the more accessible all states are to the system. Traditionally, $\beta = 1/k_B T$. Because we assume that the kinetic and potential energy terms are independent from each other in Equation 1.2, we can split the Boltzmann pdf:

$$p(m) = p(\mathbf{p}) \cdot p(\mathbf{q}) = Z_p^{-1} e^{-\beta E_{KE}(\mathbf{p})} \cdot Z_q^{-1} e^{-\beta E_{PE}(\mathbf{q})}, \tag{1.5}$$

where Z_p and Z_q are given by

$$Z_p = \sum_k e^{-\beta E_{KE}(\mathbf{p}_k)}, \tag{1.6}$$

$$Z_q = \sum_k e^{-\beta E_{PE}(\mathbf{q}_k)}. \tag{1.7}$$

Analyzing only the Boltzmann pdf, we see that the most probable configuration(s) minimizes the energy $E(m)$; configurations of equal energy are equally probable. Changing the random variable from microstate m to energy, $E(m) = \epsilon$:

$$p(\epsilon) = \sum_k p(m_k)\delta(E(m_k) - \epsilon) = Z^{-1} e^{-\beta\epsilon} \sum_k \delta(E(m_k) - \epsilon) = Z^{-1}\Omega(\epsilon) e^{-\beta\epsilon}, \tag{1.8}$$

where $\Omega(\epsilon)$ is simply the number of microstates with energy ϵ. We define entropy as

$$S(\epsilon) = k_B \ln \Omega(\epsilon). \tag{1.9}$$

Given this definition, Equations 1.8 and 1.9 can be used to give

$$p(\epsilon) = Z^{-1} e^{-\beta(\epsilon - TS(\epsilon))} = Z^{-1} e^{-\beta F(\epsilon)}, \tag{1.10}$$

where $F(\epsilon) = \epsilon - TS(\epsilon)$ is the Helmholtz free energy. Therefore, if we work in the energy space, it is the free energy that is minimized at equilibrium and the entropy that informs us of the degeneracy of a particular energy state.

The other ensembles are also important to analyze, especially since finite systems do not allow for ensemble equivalency. For example, one can imagine an open system with a particular volume, V, and temperature, T, but does not maintain the same particle number, N, in time. This grand canonical system prescription has important applications in open particle systems.

1.1.1 Thermodynamics of small systems

The fundamental laws of thermodynamics and their derivations from statistical mechanics require that the system be in the thermodynamic limit. Under these circumstances, the system parameters typically do not exhibit significant fluctuations. The following derivation portrays this principle.

Using the *maximum-term method*[19], one can simplify Equation 1.4:

$$Z = \sum_k e^{-\beta E(m_k)} = \sum_{\{\epsilon\}} e^{-\beta F(\epsilon)} \approx e^{-\beta F(\epsilon^*)}, \tag{1.11}$$

where ϵ^* is the energy state that minimizes $F(\epsilon)$, i.e., the most probable energy state. This approximation holds true for systems in the thermodynamic limit (see Appendix A for the proof). The average energy can be computed as follows:

$$\bar{E} = \sum_k E(m_k) Z^{-1} e^{-\beta E(m_k)} = -\frac{\partial \ln Z}{\partial \beta}. \tag{1.12}$$

If we define an average free energy, $\bar{F} = \bar{E} - T\bar{S}$, where both energy and entropy are

also averages, then we can also use the following equation for the average energy:

$$\bar{E} = \bar{F} + T\bar{S} = \bar{F} - T\frac{\partial \bar{F}}{\partial T} = -T^2 \frac{\partial}{\partial T}\left(\frac{\bar{F}}{T}\right) = \frac{\partial(\beta \bar{F})}{\partial \beta}. \tag{1.13}$$

Equations 1.12 and 1.13 tell us that

$$\bar{F} = -k_B T \log Z. \tag{1.14}$$

In the thermodynamic limit, Equation 1.11 tells us that

$$F(\epsilon^*) = -k_B T \log Z. \tag{1.15}$$

Hence, in the thermodynamic limit, the average energy state is also the most probable energy state. We can get an idea of how true this is by analyzing the variance in energy, $var(E) = \overline{E^2} - \bar{E}^2$:

$$\begin{aligned} var(E) &= \overline{E^2} - \bar{E}^2 = Z^{-1} \sum_k E^2 e^{-\beta E} - Z^{-2} \left(\sum_k E e^{-\beta E} \right)^2, \\ &= \frac{\partial^2 \ln Z}{\partial \beta^2} = -\frac{\partial \bar{E}}{\partial \beta} = k_B T^2 \frac{\partial \bar{E}}{\partial T} = k_B T^2 \cdot C, \end{aligned} \tag{1.16}$$

where C is the heat capacity. The relative root mean square difference of the energy is given by

$$rms(E) = \frac{\sqrt{var(E)}}{\bar{E}} = \frac{\sqrt{k_B T^2 \cdot C}}{\bar{E}}. \tag{1.17}$$

The root mean square difference essentially is the average energy difference from the mean normalized by the average energy. Having a small value for this ratio means that the energy does not deviate much from its mean value. In the thermodynamic limit, the heat capacity and the system energy are extensive, i.e., $(E, C) = O(N^1)$,

which means

$$rms(E) = O\left(\frac{1}{\sqrt{N}}\right). \tag{1.18}$$

Therefore, for a system in the thermodynamic limit, the root mean square difference is essentially nonexistent (for $N = O(10^{23}), rms(E) = O(10^{-12})$), and the most probable energy is the average energy. However, as we decrease N, as in biological and nanotechnological systems, fluctuations become significant.

Researchers are currently working on new theories, e.g., thermodynamics of small systems[23, 24], nonextensive thermodynamics[25], which account for phenomena that counter the classical laws due to finite system sizes. For instance, it has been shown that finite systems may violate the second law of thermodynamics[26], the principles of extensivity and intensivity[27], and ensemble equivalency[28]. Mohazzabi and Mansoori[27] have shown that finite systems exhibit subextensivity, i.e., $E, C \sim O(N^{\lambda})$, where $\lambda > 1$. We can generalize Equations 1.17 and 1.18 as follows:

$$rms(E) = \frac{\sqrt{k_B T^2 \cdot C}}{\bar{E}} \sim \frac{\sqrt{O(N^{\lambda})}}{O(N^{\lambda})} = O\left(\frac{1}{\left(\sqrt{N}\right)^{\lambda}}\right) = O\left(N^{-\frac{\lambda}{2}}\right). \tag{1.19}$$

Ensemble equivalency has often been applied to problems which are difficult to solve under one set of thermodynamic conditions but easily solved under another set of conditions. For finite systems, such methods are no longer available. Statistical mechanics, however, is not restricted to infinite systems and we utilize its basic principles that do not utilize the thermodynamic limit assumption.

1.1.2 Ergodicity considerations

Equations 1.3 and 1.10 both assume that the "flow" through phase space allows eventual access to any other state from any particular state. This is the *ergodic hypothesis*, and is an important consideration. For the controlled self-assembling processes we discuss in Chapters 3 and 4, the externally imposed controls offer degrees

of freedom that can be used to decrease the volume of phase space accessible to the system, i.e., decrease the number of microstates accessible from any given microstate. Systems that exhibit such *nonergodicity*[29] are known as glassy systems[30, 31] and characteristically have rugged energy landscapes. As a result, transitioning between two ergodic subsets of phase space, separated by a large energetic barrier, is not very probable and requires either a sufficiently small β value or a sufficiently long period of time.

In questioning whether the behavior of a particular system is ergodic, one always needs to consider two aspects: (1) the time-scale of interest, and (2) the temperature of the system. Given a finite measurable time, many systems are effectively nonergodic and are therefore not properly described by the Boltzmann equation. Also, as one lowers the system temperature, the system may become trapped within energy wells, breaking system ergodicity. Spin glasses[30] have become the model system for the study of nonergodic behavior, which has found applicability in areas of research such as protein folding[32, 33].

A system with competing interactions exhibits "frustration". This phenomena describes the system's inability to satisfy all competing internal interactions, making a rough energy landscape with many local minima separated by high energy barriers. Given a sufficiently low temperature, i.e., below the *glass transition temperature*, or a finite time-scale, a system exhibits nonergodic behavior. However, one can analyze a nonergodic system's phase space Ω, as a set of ergodic subsets, called *components*, where

$$\Omega = \cup_\alpha \Omega_\alpha. \tag{1.20}$$

A system which is in equilibrium within but not between components is referred to as being in *quasiequilibrium*, which describes a system trapped in a local minimum well. Because of the ergodicity within a component, one may use the Boltzmann equation

within each component:

$$p_\alpha(m_i) = \frac{e^{-\beta E(m_i)}}{\sum_{m_k \in \Omega_\alpha} e^{-\beta E(m_k)}} = Z_\alpha^{-1} e^{-\beta E(m_i)}, \tag{1.21}$$

where Z_α is now the partition function of the ergodic component, α, and the sum over k is over all states in α with state m_i being in α.

In the Static Problem, we require that the desired configuration be robust, i.e., once the system reaches the desired configuration, it stays in it. From a statistical mechanics perspective, this requirement implies that we want to maximize the probability of the system being in the desired state and therefore minimize the desired state's energy compared to energies of all other accessible states. Under such conditions, the partition function is dominated by the desired state, leading to its increased probability, i.e., $p(m_{desired}) \to 1$. This essentially means we have created a nonergodic system where the desired configuration is the only probable state in the accessible system component. These considerations form the basis of the problem formulation for the Static Problem (Chapter 3). Nonergodicity is also a desirable trait in the self-assembly process, where the desired configuration is the system's final equilibrium state, because it reduces the total number of undesirable competing states. In Chapter 4, we propose a systematic restriction of phase space, starting with an ergodic system where all of phase space is accessible. At all times, the desired state is a member of the component the system is trapped in.

Chapter 2

Model systems

The model system we use to demonstrate our self-assembly design strategies is an isomorph of the Ising model[34], the workhorse of statistical mechanics. We assume that the system particles do not rotate but simply translate throughout the discrete system volume. We also assume that the particles are indistinguishable and negatively charged with a charge equal to -1. Analyzing only the positional pdf, the potential energy function (note that we have dropped the subscript) is given by

$$E(\mathbf{z}) = E_{ext}(\mathbf{z}) + E_{int}(\mathbf{z}) = \sum_{i=1}^{V}\sum_{k=1}^{N_d} z_i H_{i,k} s_k + \sum_{i<j} z_i J_{i,j} z_j = \mathbf{z}^{\mathrm{T}}\mathbf{H}\mathbf{s} + \mathbf{z}^{\mathrm{T}}\mathbf{J}\mathbf{z}. \quad (2.1)$$

The binary vector, $\mathbf{z}$, represents the system configuration, where $z_i = 0$ represents an empty lattice site and $z_i = 1$ represents the presence of a particle. V is the system volume, i.e., the number of lattice sites, and N_d is the number of external fields (controls). The first sum, E_{ext}, is the total energy imparted on the system by external fields, and the second sum, E_{int}, accounts for binary interactions between the system constituents (particles). Higher-order interaction terms may be included, if necessary. Because we are using a phenomenological model, we tend to simplify the system and focus on the most important contributors to the system behavior. The parameter s_k is the strength of external field k.

Many binary interaction potential energy models have the following form:

$$E_{int}(\mathbf{z}) = \sum_{i,j} z_i J_{i,j} z_j = \sum_{i,j} z_i z_j \sum_p c_{i,j}^{(p)} f^{(p)}(r_{i,j}) = \sum_{i,j} z_i z_j \sum_p \frac{c_{i,j}^{(p)}}{(r_{i,j})^{m_p}}, \tag{2.2}$$

where $r_{i,j}$ is the positional distance between constituents i and j, and $c_{i,j}^{(p)}$ is typically a fitted parameter used to match the model to experimental results. The larger the exponent, m_p, the shorter the interaction range. For instance, the classic Lennard-Jones potential has an attraction interaction exponent, $m = 6$, whereas the Coulombic potential has an interaction exponent, $m = 1$. Figure 2-1 shows that Coulombic interactions are more long-range compared to the Lennard-Jones model for van der Waals attraction. Of course the total interaction energy is the sum of all short- and long-range contributions, shown in the above equation as a sum with the index p. Both types of interactions are important considerations in a self-assembly process; long-range interactions may be used to attract or repel system constituents within the system volume and short-range interactions may be used to define the local geometry of neighboring constituents. In this body of work, we consider particles interacting with both long- and short-range interactions.

The most basic external field potential function takes the following form:

$$E_{ext}(\mathbf{z}) = \sum_{i,k} z_i H_{i,k} s_k = \sum_{i,k} z_i c_{i,k} s_k, \tag{2.3}$$

where $c_{i,k}$ is simply a constant. We may also introduce external fields that can take a basic form similar to the binary interaction potential above:

$$E_{ext}(\mathbf{z}) = \sum_{i,k} z_i H_{i,k} s_k = \sum_{i,k} z_i s_k \sum_p c_{i,k}^{(p)} f^{(p)}(r_{i,k}) = \sum_{i,k} z_i s_k \sum_p \frac{c_{i,k}^{(p)}}{(r_{i,k})^{m_p}}. \tag{2.4}$$

Here, we introduce a distance parameter, $r_{i,k}$, which represents the distance between lattice site i and external control k. We call this type of external field a *point condition* because the field originates at a point that is a certain distance away from each lattice site and influences the lattice site through a potential that is a function of

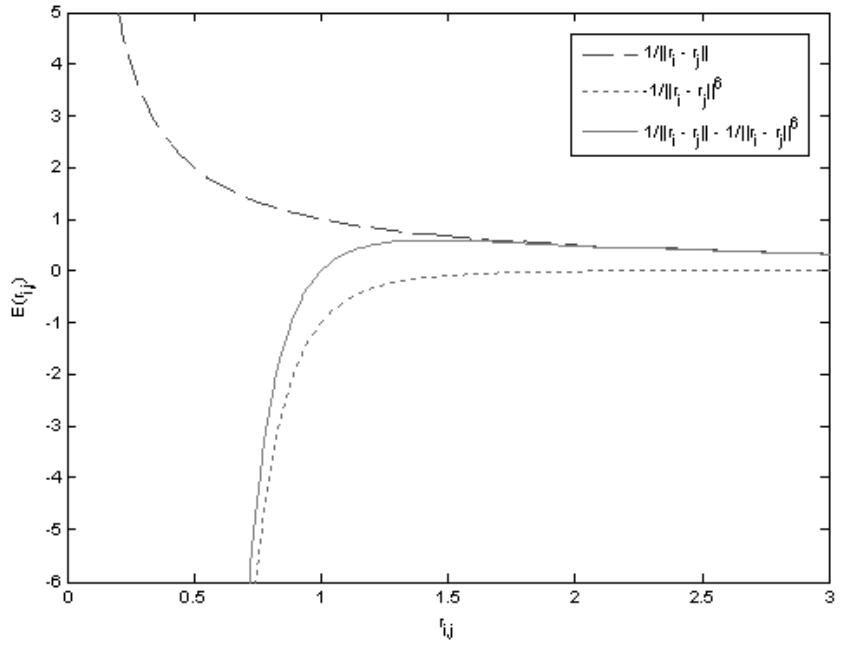

Figure 2-1: The long-range repulsive and short-range attractive interactions among the self-assembling particles.

that distance. Each point condition has a positional array and can be located within (*internal* point condition) or outside (*boundary* point condition) the system volume. The point condition locations and strengths are determined by solving the Static Problem (see Chapter 3). When s_k is positive, the point condition k, in general, produces an energy well, and when s_k is negative, k produces an energy barrier.

2.1 Potential energy considerations

In all of our examples, we utilize point conditions as a means of controlling the features of the potential energy landscape. More specifically, we use the external potential energy function in Equation 2.4 with $p = 1$, $c_{i,k}^{(1)} = -1$ and $m_1 = 1$. This is a simple phenomenological model for Coulombic charged interactions. The negative constant value simply represents the fact that we specify negatively charged particles. We take a more complicated form for the interparticle interaction potential energy

function, using Equation 2.2 with $p = 1, 2$; $(c_{i,j}^{(1)}, c_{i,j}^{(2)}) = (1, -1)$; and $(m_1, m_2) = (1, 6)$. These parameter values represent a simple phenomenological model for long-range Coulombic charged repulsion and short-range van der Waals attraction. More specifically, we utilize the following overall potential energy function:

$$E = E_{ext} + E_{int} = -\sum_{i,k} z_i \frac{s_k}{r_{i,k}} + \sum_{i,j} z_i z_j \left(\frac{1}{r_{i,j}} - \frac{1}{r_{i,j}^6} \right). \tag{2.5}$$

For real systems, more complex interaction energy models will be required, but the above model serves as a phenomenological one.

2.2 Simulation techniques

Because we are working with a lattice model system and we are only analyzing the potential energy of a specific microstate, i.e., we are only looking at the system configuration space, we utilize Monte Carlo techniques[34, 35] to simulate our system. For a particle system of N particles in a finite volume V operated at a specific temperature T, one Monte Carlo simulation takes a representative system from an initial configuration to its stable equilibrium state. At equilibrium, the probability of being in a particular configuration state arrives at a limiting value, called the equilibrium probability. The evolution of the probability distribution in time is represented by the master equation:

$$\frac{d}{dt} p(\mathbf{z}_i, t) = \sum_{i \neq j} \left[p(\mathbf{z}_j, t) W_{j \to i} - p(\mathbf{z}_i, t) W_{i \to j} \right], \tag{2.6}$$

where $p(\mathbf{z}_i, t)$ is the probability of being in state i at time t. The first term in the summation is the rate that the system arrives at state i from state j; $W_{j \to i}$ is the probability of transitioning from state j to state i. The second term is the rate that the system leaves state i. As the system approaches equilibrium, the above equation reaches a steady state, i.e., $p(\mathbf{z}_i, t) \to p(\mathbf{z}_i)$. There are many ways that this equation can equal zero, but the most common approach is to impose detailed balance on the

system:

$$p(\mathbf{z}_j)W_{j\to i} = p(\mathbf{z}_i)W_{i\to j}, \forall(i,j), i \neq j. \tag{2.7}$$

There are many choices for $W_{i\to j}$ that satisfy the above equation, the most famous of which is the Metropolis Monte Carlo scheme[36]:

$$W_{i\to j} = \min\left\{1, \frac{p(\mathbf{z}_j)}{p(\mathbf{z}_i)}\right\}. \tag{2.8}$$

Using the Boltzmann distribution equation for an ergodic phase space or an ergodic subset of phase space, this equation can be expanded:

$$W_{i\to j} = \min\left\{1, e^{-\beta(E(\mathbf{z}_j)-E(\mathbf{z}_i))}\right\}. \tag{2.9}$$

The right-hand side of the above equation comes from comparing the two energy values, $E(\mathbf{z}_i)$ and $E(\mathbf{z}_j)$. If $E(\mathbf{z}_i) \geq E(\mathbf{z}_j)$, then the minimum value is unity. If, however, $E(\mathbf{z}_i) < E(\mathbf{z}_j)$, then the minimum value is the exponential term. To prove detailed balance, if $E(\mathbf{z}_i) \geq E(\mathbf{z}_j)$,

$$\begin{aligned}
W_{i\to j} &= 1, \\
p(\mathbf{z}_i)W_{i\to j} &= p(\mathbf{z}_i), \\
&= e^{-\beta E(\mathbf{z}_i)}; \\
W_{j\to i} &= e^{-\beta(E(\mathbf{z}_i)-E(\mathbf{z}_j))}, \\
p(\mathbf{z}_j)W_{j\to i} &= p(\mathbf{z}_j)e^{-\beta(E(\mathbf{z}_i)-E(\mathbf{z}_j))}, \\
&= e^{-\beta E(\mathbf{z}_i)}.
\end{aligned} \tag{2.10}$$

As can be seen, detailed balance is satisfied. If $E(\mathbf{z}_i) < E(\mathbf{z}_j)$, we can perform a similar analysis:

$$\begin{aligned} W_{i\to j} &= e^{-\beta(E(\mathbf{z}_j)-E(\mathbf{z}_i))}, \\ p(\mathbf{z}_i)W_{i\to j} &= p(\mathbf{z}_i)e^{-\beta(E(\mathbf{z}_j)-E(\mathbf{z}_i))}, \\ &= e^{-\beta E(\mathbf{z}_j)}; \\ W_{j\to i} &= 1, \\ p(\mathbf{z}_j)W_{j\to i} &= p(\mathbf{z}_j), \\ &= e^{-\beta E(\mathbf{z}_j)}. \end{aligned} \tag{2.11}$$

This also satisfies the detailed balance criteria.

In standard Monte Carlo simulations of particle systems, the system is initialized at a particular state, $\mathbf{z}_0$, and a sequence of one-step moves of individual particles are proposed and accepted using the acceptance probability in Equation 2.9, leading the system to equilibrium. The transition from the initial state to the equilibrium distribution of states may also serve as an approximation to the system's nonequilibrium dynamics. However, if the system exhibits strong interaction potentials, defined by U_{int} and U_{ext}, we may find that the system gets trapped in unphysical kinetic traps. For example, if two particles are located in neighboring lattice sites and there is a strong short-range attractive potential between them, then the likelihood of one particle moving one site away from its current location is very low. In essence, the system is trapped in an unphysical potential energy well. It is unphysical because a real system's potential energy surface would not include such a well; the two particles would simply move together in concert. The problem with the standard Monte Carlo approach therefore lies in the fact that it does not allow for movement of clusters of particles that are interacting through strong-range attractive potentials. Therefore, the Monte Carlo method we use throughout the self-assembly studies in Chapters 3 and 4 utilize the "virtual-move" Monte Carlo (vmmc) algorithm, developed by Whitelam and Geissler[37].

2.2.1 Dynamic Monte Carlo

Given a system of particles, the vmmc algorithm chooses a seed particle, i, and creates a cluster of linked particles, C_i, that moves together in concert. The same cluster of particles can be formed and moved in concert using multiple linkages with different seed particles. Similar to the detailed balance criteria above, the vmmc method satisfies superdetailed balance through the following equilibrium requirement:

$$p(\mathbf{z}_j)W_{j\to k|C_i} = p(\mathbf{z}_k)W_{k\to j|C_i}, \forall(i,j,k), j \neq k. \tag{2.12}$$

It is called superdetailed balance because it is specified over a specific linking between particles. More specifically,

$$W_{j\to k|C_i} = W^{gen}_{j\to k|C_i} W^{acc}_{j\to k|C_i}, \tag{2.13}$$

where W^{gen} is the probability of proposing (i.e., generating) and moving from state $\mathbf{z}_j$ to state $\mathbf{z}_k$, and W^{acc} is the probability of accepting the move from state $\mathbf{z}_j$ to $\mathbf{z}_k$. We can break down the first term as follows:

$$W^{gen}_{j\to k|C_i} \quad = \quad p_{seed}(\mathbf{z}_j)p_{displace}(C_i;\mathbf{z}_j \to \mathbf{z}_k), \tag{2.14}$$

where $p_{seed}(\mathbf{z}_j)$ is the probability of choosing a seed particle in state $\mathbf{z}_j$ and $p_{displace}(C_i;\mathbf{z}_j \to \mathbf{z}_k)$ is the probability, given a seed particle, of building C_i and moving it from state $\mathbf{z}_j$ to state $\mathbf{z}_k$. This latter term can be broken down further to two factors:

$$p_{displace}(C_i;\mathbf{z}_j \to \mathbf{z}_k) = \prod_{\{mn\}_{nl}} (1 - p_{mn}(\mathbf{z}_j \to \mathbf{z}_k)) \prod_{\{mn\}_l}^{C_i} p_{mn}(\mathbf{z}_j \to \mathbf{z}_k). \tag{2.15}$$

The first product is the probability of not forming links between all the particles within C_i and all the particles that do not belong to C_i. Hence, the product over $\{mn\}_{nl}$ is the product over all particle pairs, m and n, that must not form in order to move from state $\mathbf{z}_j$ to $\mathbf{z}_k$. The second product is the probability of forming a specific

set of links between the particles that form the C_i cluster.

From Equation 2.12, superdetailed balance now states that

$$p(\mathbf{z}_j)W^{gen}_{j\to k|C_i}W^{acc}_{j\to k|C_i} = p(\mathbf{z}_k)W^{gen}_{k\to j|C_i}W^{acc}_{k\to j|C_i}, \forall(i,j,k), j \neq k. \tag{2.16}$$

This can be rearranged as follows:

$$\frac{W^{acc}_{j\to k|C_i}}{W^{acc}_{k\to j|C_i}} = \frac{p(\mathbf{z}_k)}{p(\mathbf{z}_j)}\frac{W^{gen}_{k\to j|C_i}}{W^{gen}_{j\to k|C_i}}. \tag{2.17}$$

There are many choices for $W^{acc}_{j\to k|C_i}$ that satisfy the above equation. Whitelam and Geissler use the following form:

$$\begin{aligned} W^{acc}_{j\to k|C_i} \quad = \quad & \min\left\{1, \frac{p_{seed}(\mathbf{z}_k)}{p_{seed}(\mathbf{z}_j)}e^{-\beta(E(\mathbf{z}_k)-E(\mathbf{z}_j))}\right. \\ & \left.\cdot\frac{\prod_{\{mn\}^{k\to j}_{nl}} 1-p_{mn}(\mathbf{z}_k\to\mathbf{z}_j)}{\prod_{\{mn\}^{j\to k}_{nl}} 1-p_{mn}(\mathbf{z}_j\to\mathbf{z}_k)}\cdot\prod_{\{mn\}_l}^{C_i}\frac{p_{mn}(\mathbf{z}_k\to\mathbf{z}_j)}{p_{mn}(\mathbf{z}_j\to\mathbf{z}_k)}\right\}. \end{aligned} \tag{2.18}$$

We can choose the seed particle from a uniform distribution, and therefore $p_{seed}(\mathbf{z}_j) = p_{seed}(\mathbf{z}_k)$. The probability $p_{mn}(\mathbf{z}_j \to \mathbf{z}_k)$ depends on the energy difference of the bond between particles m and n before and after the seed particle is moved:

$$p_{mn}(\mathbf{z}_j \to \mathbf{z}_k) = \max\left\{0, 1 - e^{-\beta(E_I(m,n)-E_C(m,n))}\right\}. \tag{2.19}$$

The energy $E_C(m,n)$ is the energy of the bond between a concerted move of particles m and n. Because the particles do not move relative to each other, this bond energy is identical at the beginning and end of the move. The energy $E_I(m,n)$ is the bond energy following an individual move of particle m without moving particle n.

Using the above derivations, the vmmc algorithm proceeds as follows:

Step 0: Start in state $\mathbf{z}_j$. Define the number of MC steps for the simulation.

Step 1: Select from a uniform distribution a seed particle, m, and a proposed translation.

Step 2: Find the particles, $\{n\}$, that interact with the seed particle. Link a given particle, n, to the seed particle with probability, $p_{mn}(\mathbf{z}_j \rightarrow \mathbf{z}_k)$.

Step 3: Calculate $p_{mn}(\mathbf{z}_k \rightarrow \mathbf{z}_j)$.

Step 4: Perform Steps 2 and 3 recursively for each linked particle to form new links with each linked particle acting as the "seed".

Step 5: After no more possible links can be evaluated, update the position of the formed cluster using the proposed translation defined in Step 1.

Step 6: Evaluate the acceptance probability, and update the state of the system accordingly.

Step 7: Perform Steps 1-6 for the remaining MC steps.

All simulations in Chapters 3 and 4 use the vmmc algorithm to simulate both system equilibrium and nonequilibrium dynamics.

Chapter 3

The static problem

The static problem deals with maintaining the desired structure once it has been reached by the dynamic process. In essence, we assume that we have attained the desired geometry and our concern is to make this state robust. We are creating an ergodic component that consists of the just the desired state, and sufficiently large energy barriers keep the system inside this component. In order to shape the energy landscape for the creation of this ergodic component, we introduce system degrees of freedom in the form of externally-controlled point conditions. Each point condition has two parametric degrees of freedom: the location of the point conditions and their strength. The qualitative features of the energy landscape are defined by the location of the point conditions and their energetic characteristics, e.g., attractive or repulsive, while the quantitative features (the robustness) are determined by the point condition strengths.

We accordingly divide the Static Problem into two subparts: (a) the qualitative definition of the energy landscape through the specification of the point condition locations; (b) quantifying the robustness of the desired structure through the specification of the strengths of the point conditions. Section 3.1 details how to systematically place point conditions in order to qualitatively shape the energy landscape, introducing the minimum tiling algorithm for attractive and repulsive point conditions. Section 3.2 details the problem formulation and solution methodology for defining the strengths of the point conditions. To find the strengths we define and

solve the Energy-gap Maximization Problem (EMP), a combinatorially-constrained mixed-integer quadratic optimization problem. Section 3.4 provides examples that illustrate the qualitative and quantitative solutions of the Static Problem. Given the set of point conditions used, the system robustness can be quantified and the appropriate operating system temperature can be found through simulation techniques. If the system robustness is such that the necessary operating temperature is impractically low, we must introduce additional degrees of freedom to increase robustness. Section 3.3 details how we can use the EMP output to find the constraining features of the desired configuration and introduce the necessary degrees of freedom that energetically influence these features.

3.1 Qualitatively shaping the energy landscape: the minimum tiling approach

The qualitative features of the energy landscape are defined by the location of the degrees of freedom. The degrees of freedom we use are isotropic point conditions, which are attractive or repulsive in nature. Attractive point conditions introduce wells in the energy landscape, attracting the particles; repulsive point conditions introduce barriers, deflecting particles. Because the presence of two wells forms a barrier and vice versa, only one of these types (well- or barrier-forming) is needed to define the qualitative features of the energy landscape. However, when solving EMP, we may find that additional point conditions are needed to maintain a certain level of robustness. It is therefore necessary to solve EMP with both types of point conditions to see which is optimal in terms of two factors: (1) the total number of degrees of freedom needed, (2) the minimum distance between the degrees of freedom. Based on fabrication limitations, one can select which set of point conditions is more practical, while still maintaining the desired level of robustness for the desired structure.

To find the minimum number of well- and barrier-forming point conditions, we have formulated the minimum tiling algorithm below. For each point condition we

assign a tile that encompasses the local area that the point condition is intended to influence. Thus, the minimum number of tiles needed to cover the well- and barrier-forming regions determines the minimum number of attractive and repulsive point conditions, respectively.

In our example systems, we restrict the point conditions to 3 types with respect to their locations. As shown in Figure 3-1, the point conditions can be on a lattice site, between two lattice sites, or between four lattice sites. The first two types apply to both the 1- and 2-D example systems while the third type applies only to the 2-D examples. One can consider the possible point condition locations as another system degree of freedom. The more types of point conditions (or, more generally, degrees of freedom) we provide the system, the less the number of point conditions we need, and vice versa. It is straightforward to see that for 1-D systems ($V = L$) the number of possible point condition locations can be calculated for each type as follows:

$$\begin{aligned} N_{pc}^{a} &= L, \\ N_{pc}^{b} &= L + 1, \\ N_{pc} &= N_{pc}^{a} + N_{pc}^{b} = 2L + 1. \end{aligned} \tag{3.1}$$

In the same fashion, the number of possible point condition locations for 2-D square-lattice systems ($V = L^2$) can be calculated as follows:

$$\begin{aligned} N_{pc}^{a} &= L^2, \\ N_{pc}^{b} &= 2(L+1)L = 2L^2 + 2L, \\ N_{pc}^{c} &= (L+1)^2 = L^2 + 2L + 1, \\ N_{pc} &= N_{pc}^{a} + N_{pc}^{b} + N_{pc}^{c} = 4L^2 + 4L + 1. \end{aligned} \tag{3.2}$$

We only consider isotropic point conditions, and therefore each point condition's area of influence grows radially in all directions. The discretization of the system volume into lattice sites and the different locations of each point condition type cause their respective areas of influence to form different tile shapes. Figure 3-2 shows how

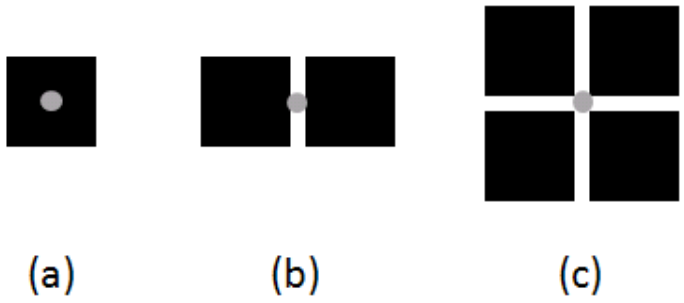

Figure 3-1: The three types of point conditions used in the example systems: (a) on a lattice site, (b) between two lattice sites, (c) between four lattice sites.

each type of point condition develops different tile shapes of increasing sizes. As a point condition's strength increases so does the area of influence it possesses. A point condition's tile can only include lattice sites of one particular type, i.e., occupied sites for well-forming point conditions and unoccupied sites for barrier-forming point conditions. Therefore, when an incremental growth in the radius of a point condition's area of influence causes the inclusion of a lattice site of the opposing type, we know that the tile has reached its maximum size.

Given a desired lattice structure, the minimum tiling algorithm proceeds as follows:

Step 1: Locate all possible point conditions of a particular type (well- or barrier-forming). $N_{pc} = O(V)$.

Step 2: Grow each point condition's tile (defining the area of influence) to its maximum size. In other words, increment the radius of influence of each point condition until any further increase would include a lattice site of the opposing type.

Step 3: Eliminate all point conditions with null areas of influence. These point conditions are located in areas where the opposite point condition type is needed.

Step 4: Eliminate all point conditions with areas of influence completely subsumed by another point condition's area of influence. If the set of lattice sites

in the area of influence of point condition i is S_i, we eliminate any point condition j where $S_j \subseteq S_i$. If $S_j = S_i$, then one point condition is chosen, which therefore allows for multiple solution sets of point conditions. We may eliminate multiple solutions by choosing the location that optimizes a particular system characteristic, e.g., maximizes the distance between point condition locations.

Step 5: Solve a set cover optimization problem with the remaining point conditions:

$$\begin{aligned} \min_{\mathbf{x}} \quad & \sum_i x_i \\ s.t. \quad & \sum_i a_{ij} x_i \geq 1, \forall j, \\ & x_i \in \{0, 1\}, \end{aligned} \tag{3.3}$$

where x_i is a binary variable that takes a value of 1 if the point condition is kept and 0 if it is discarded. The parameter a_{ij} is a binary parameter that equals 1 if lattice site j is within the area of influence of point condition i, i.e., j is a member of i's tile, and 0 otherwise. The set of point conditions with $x_i = 1$ represents the minimum set that covers all the lattice sites. After Step 4, the resulting point condition set is essentially all point conditions with independent sets of lattice sites in their areas of influence. Solving the set cover problem accounts for the fact that the union of multiple point condition tiles may subsume another point condition's area of influence, e.g., $S_i \subset \cup_{\{j, i \neq j\}} S_j$.

Step 6: Repeat Steps 1-5 for the other point condition type.

Figure 3-3 depicts the logical flowchart for Steps 1-5 of the minimum tiling algorithm.

The set cover optimization problem in Step 5 is an NP-hard problem with an NP-complete decision problem equivalent. If the number of point conditions considered is small enough, the set cover optimization problem can be solved for the minimum set. However, if the number of point conditions is too large, one must use an approximate algorithm, e.g., the greedy cover algorithm[38]. The greedy cover algorithm sequentially chooses the point condition that covers the most number of lattice sites not yet

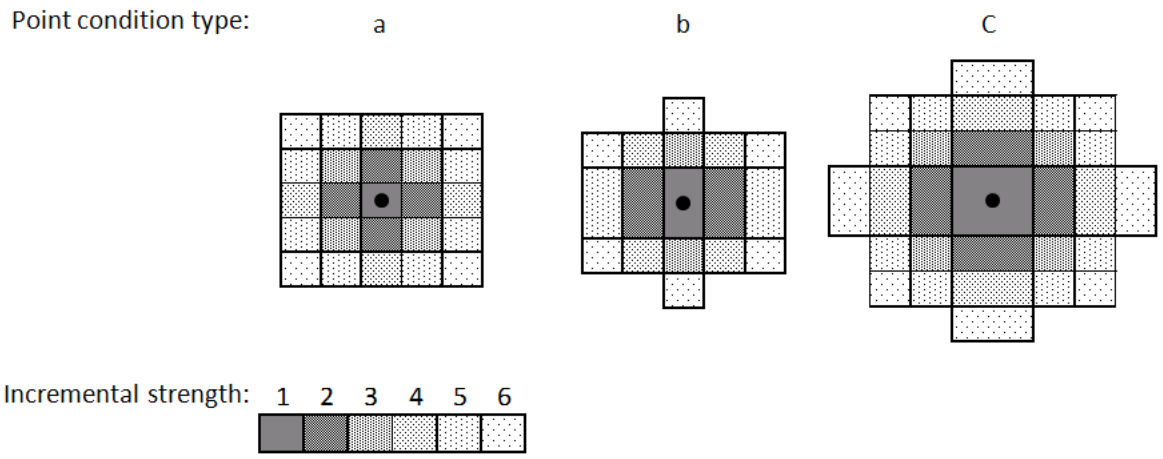

Figure 3-2: The geometry of tiles generated from the three types of point conditions defined in Figure 3-1.

covered by another chosen point condition.

The minimum tiling algorithm will output two sets of point conditions, the set representing the necessary well-forming point conditions to cover the occupied lattice sites and the set representing the necessary barrier-forming point conditions to cover the unoccupied lattice sites. These sets of point condition locations can be used to quantitatively shape the energy landscape, as described in Section 3.2.

3.2 Quantitatively shaping the energy landscape: the energy-gap maximization problem (EMP)

Given the number of point conditions and their locations generated by the minimum tiling algorithm (Section 3.1), we must now guarantee that the desired configuration is robust, i.e., we want to maintain the desired structure given that the dynamic process allows us to reach the desired state. To do this we utilize the point condition strengths as our degrees of freedom. From Section 1.1, we may formulate this problem for a system in the Canonical prescription by simply maximizing its Boltzmann probability:

$$\max_{\mathbf{s}\in\mathbf{S}} p(\mathbf{z}_d, \mathbf{s}) = \max_{\mathbf{s}\in\mathbf{S}} \frac{e^{-\beta E(\mathbf{z}_d,\mathbf{s})}}{\sum_{\mathbf{z}_j\in\Omega_\alpha} e^{-\beta E(\mathbf{z}_j,\mathbf{s})}} = \max_{\mathbf{s}\in\mathbf{S}} Z_\alpha(\mathbf{s})^{-\mathbf{1}} \mathbf{e}^{-\beta \mathbf{E}(\mathbf{z_d},\mathbf{s})}, \tag{3.4}$$

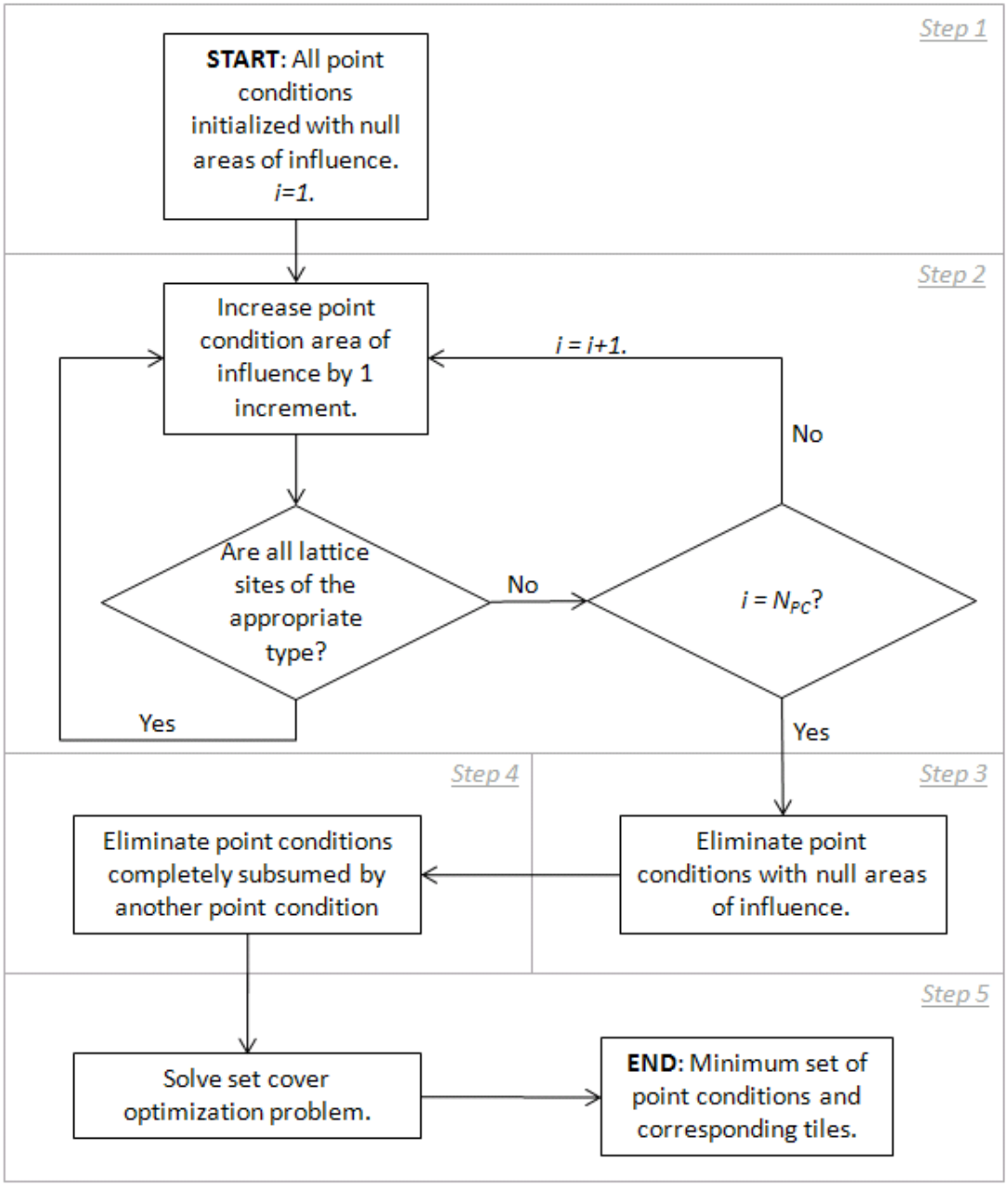

Figure 3-3: The flowchart of the minimum tiling algorithm that generates the minimum number of well- or barrier-forming point conditions.

where $\mathbf{s}$ is the set of point condition strengths needed to calculate the system energy. We specify the partition function as that of the ergodic component, α, and the sum over j is over all configurations in α. Configuration $\mathbf{z}_d$ is in α. From Section 1.1, we are also familiar with the ratio:

$$\frac{p(\mathbf{z}_i, \mathbf{s})}{p(\mathbf{z}_j, \mathbf{s})} = e^{-\beta(E(\mathbf{z}_i,s)-E(\mathbf{z}_j,s))}. \tag{3.5}$$

If $E(\mathbf{z}_i, \mathbf{s}) >> E(\mathbf{z}_j, \mathbf{s})$, this ratio approaches zero, and we know that state $\mathbf{z}_j$ has a high probability. If $E(\mathbf{z}_i, \mathbf{s}) << E(\mathbf{z}_j, \mathbf{s})$, this ratio approaches infinity, and we know $\mathbf{z}_i$ has a higher probability. Using this analysis, we see that we may analyze the equilibrium probability of being in a particular state versus any accessible competing states simply by looking at energy differences. We would like to minimize the energy of the desired state with respect to all competing configurations. To minimize the complexity of the problem, we may consider simply the competing configuration of minimum energy and the energy difference between this configuration and the desired configuration, see Figure 3-4. Using this approach, we may recast Equation 3.4 as a bilevel optimization problem:

$$\begin{aligned} \max_{\mathbf{s}\in\mathbf{S}} \quad & E^*(\mathbf{s}) - E(\mathbf{z}_d, \mathbf{s}) \\ s.t. \quad & E^*(\mathbf{s}) = \min_{\mathbf{z}\in\Omega_\alpha\setminus\{\mathbf{z}_d\}} E(\mathbf{z}, \mathbf{s}). \end{aligned} \tag{3.6}$$

In this optimization formulation, which we call the Energy-gap Maximization Problem (EMP), the inner problem finds the configuration in the ergodic component, excluding the desired configuration, that minimizes the system energy. This configuration is then passed to the outer problem, where we maximize the energy difference between our desired configuration and the inner problem solution through the modification of the point condition strengths. A negative objective function value means that, given the set of point condition locations, the desired configuration cannot be the energy minimum and therefore is not the configuration within component α with the highest probability. In other words, there is at least one configuration with lower

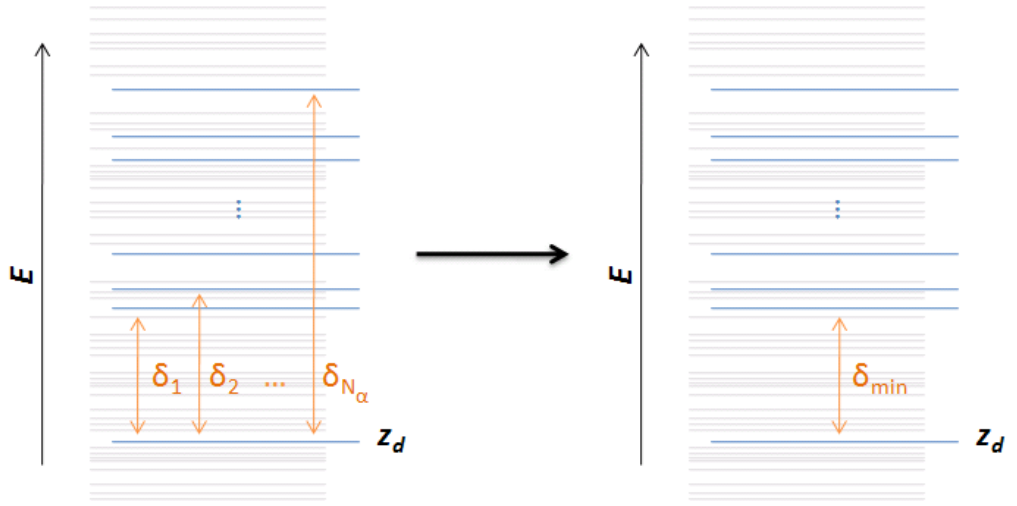

Figure 3-4: To decrease the complexity of the problem, we transition from viewing the system as a set of energy differences between the desired configuration and all N_α competing configurations to considering only the energy gap between the desired configuration and the minimum-energy competing configuration.

energy, and the actual energy minimum is the energy of the desired configuration plus the objective function value. If the objective function value is zero, there is an energetically degenerate configuration that will have the same probability as the desired configuration. However, if the value of the objective function is positive, our desired configuration has the highest probability within component α.

The problem formulation in Equation 3.4 is difficult to solve because calculating the partition function may require considering a combinatorially large number of configurations. Also, it is a $0-1$ nonconvex non-linear problem (NLP), and a global solution is not guaranteed. The reformulated optimization problem is an NLP with an embedded $0-1$ quadratic problem (QP); the outer and inner problems cannot be decoupled and solved separately. We can recast this bilevel optimization problem as a combinatorially-constrained linear problem as follows:

$$\begin{aligned} \max_{\mathbf{s}\in\mathbf{S},\delta} \quad & \delta \\ s.t. \quad & E(\mathbf{s},\mathbf{z}) - E(\mathbf{s},\mathbf{z}_d) \geq \delta, \forall \mathbf{z} \in \Omega_\alpha \backslash \{\mathbf{z}_d\}, \end{aligned} \tag{3.7}$$

where δ is the minimum energy difference between the desired configuration and the

competing configuration states. The combinatorial number of constraints in Equation 3.7 is directly related to the system volume and particle number. In Section 3.2.2 we discuss how one can implicitly solve this problem without realizing all the constraints. In Section 3.2.3, we discuss how a binary integer quadratic problem can be linearized for faster computation times and guaranteed global solutions.

The minimum-energy competing configuration gives us a lower bound to the probability of the desired structure as follows:

$$p_\alpha^{LB}(\mathbf{z}_d, \mathbf{s}) = \frac{e^{-\beta E(\mathbf{z}_d,\mathbf{s})}}{e^{-\beta E(\mathbf{z}_d,\mathbf{s})} + \sum_{\mathbf{z}_j \in \Omega_\alpha \backslash \{\mathbf{z}_d\}} e^{-\beta(E(\mathbf{z}_d,\mathbf{s})+\delta)}} = (Z_\alpha^{LB}(\mathbf{s}))^{-\mathbf{1}} \mathbf{e}^{-\beta \mathbf{E}(\mathbf{z_d},\mathbf{s})}. \quad (3.8)$$

The following inequality holds true for all configurations in $\Omega_\alpha \backslash \{\mathbf{z}_d\}$:

$$E(\mathbf{z}_j) \geq E(\mathbf{z}_d) + \delta. \quad (3.9)$$

Because of this inequality,

$$Z_\alpha^{LB}(\mathbf{s}) \geq Z_\alpha(\mathbf{s}), \quad (3.10)$$

$$p_\alpha^{LB}(\mathbf{z}_d, \mathbf{s}) \leq p_\alpha(\mathbf{z}_d, \mathbf{s}). \quad (3.11)$$

If solving EMP provides us with a lower bound on the probability of the desired configuration, we know that the actual probability is at least that value, and therefore the system will be more robust than the EMP output specifies. To find the actual probability of the desired configuration or a tighter lower bound to the actual probability, we must have more information about all other configurations' energies.

3.2.1 Defining the phase space component Ω_α

The ergodic component that includes the desired configuration and other accessible competing configuration states can be defined in multiple ways. If we consider a high-temperature system, all configurations in phase space are accessible and therefore make up Ω_α. We call this problem the Phase Space Energy-gap Maximization

Problem (PSEMP). If, however, the system temperature is sufficiently low or the timeframe we are considering is short, Ω_α is only a subset of the phase space. This subset is not always easy to define, but in Section 3.4 we provide an example where the barrier-forming point conditions define the competing configurations, in that the energetic peaks trap particles in localized subvolumes of the system. In other words, we restrict the system to specific particle number specifications. We call this problem the Component Energy-gap Maximization Problem (CEMP).

In many cases, the definition of the competing component configurations is difficult or impractical. In these cases, we propose a "one step" approach, where we consider all one-particle movements from the desired configuration. We call this problem the One Step Energy-gap Maximization Problem (OSEMP). Using this method of defining the competing configurations, the number of constraints becomes $O(N)$.

3.2.2 Reducing the number of constraints needed to solve EMP

Solving EMP requires considering a combinatorially large number of constraints. To solve such problems, we utilize a cutting-plane method. Cutting-plane methods allow us to solve optimization problems with a very large, sometimes infinite, number of constraints with only a subset of the constraints. More specifically, in our examples in Section 3.4, we utilize a cutting-plane method devised by Blankenship and Falk[39], where we start by solving the optimization problem subject to a finite subset of the combinatorial number of constraints and enlarge this subset in order to obtain better solutions to the problem. Therefore, we are solving the following problem:

$$\begin{aligned} \max_{\mathbf{s}\in\mathbf{S},\delta^i} \quad & \delta^i \\ s.t. \quad & E(\mathbf{s},\mathbf{z}) - E(\mathbf{s},\mathbf{z}_d) \geq \delta, \forall \mathbf{z} \in \Omega_\alpha^i \backslash \{\mathbf{z}_d\}, \end{aligned} \tag{3.12}$$

where Ω_α^i is a finite subset of Ω_α and i is the iteration number. As the iteration number increases, so does the accuracy of the solution. The approach to the solution

is an upper bound to the EMP global solution:

$$\delta^i \geq \delta^{EMP}, \forall i \in \mathbb{N}. \tag{3.13}$$

Clearly, if i reaches the combinatorial number of constraints defined by the subset of configurations in Ω_α, we are then solving the original EMP. However, we would like to reach δ^{EMP} before we consider the full set of constraints.

The algorithm proposed by Blankenship and Falk involves the four following steps.

Step 1. Initialize: set $i = 0$ and choose $\Omega_\alpha^i \subset \Omega_\alpha$.

Step 2. Solve the ith outer problem: compute $\mathbf{s}^i \in \mathbf{S}$ and δ^i that solves the linear problem defined in Equation 3.12.

Step 3. Solve the ith inner problem: compute $\mathbf{z}^i \in \Omega_\alpha$ that solves the following quadratic $0-1$ problem using the outer problem solution $\mathbf{s}^i$ and δ^i:

$$\min_{\mathbf{z} \in \Omega_\alpha} E(\mathbf{z}, \mathbf{s}^i) - E(\mathbf{z}_d, \mathbf{s}^i) - \delta^i. \tag{3.14}$$

Step 4. Check termination criteria and update Ω_α^i, if necessary: If $E(\mathbf{z}^i, \mathbf{s}^i) - E(\mathbf{z}_d, \mathbf{s}^i) - \delta^i \geq 0$, then iteration i has generated the global solution. Otherwise, return to Step 2 with an updated subset of constraints: $\Omega_\alpha^{i+1} = \Omega_\alpha^i \cup \{\mathbf{z}^i\}$.

The inner problem essentially finds the configuration that maximally violates the EMP energy difference constraint. If the solution to the inner problem however still satisfies the EMP constraint, then we know all other configurations in $\Omega_\alpha^i \backslash \{z_d\}$ also satisfy the constraint and we have the global solution at the current iteration.

Expanded EMP formulation

Thus far, the EMP formulation has been given in a concise form. The outer problem is an LP with variables $\mathbf{s}$ and δ, the output values of which are passed to the inner problem (Equation 3.14), which can be expanded as follows (to the form that is

actually solved) to find the minimum energy competing configuration:

$$\min_{\mathbf{z}} \quad E(\mathbf{z}, \mathbf{s}^i) - E(\mathbf{z}_d, \mathbf{s}^i) - \delta^i \tag{3.15}$$

$$s.t. \quad z_j \geq 0, \forall j = 1, 2, \cdots, V; \tag{3.16}$$

$$z_j \leq 1, \forall j = 1, 2, \cdots, V; \tag{3.17}$$

$$z_j \in \{0, 1\}, \forall j = 1, 2, \cdots, V; \tag{3.18}$$

$$\sum_{j=1}^{V} z_j = N; \tag{3.19}$$

$$N - \sum_{z_j \in \Theta} z_j \geq 1; \tag{3.20}$$

$$\sum_{z_j \in \upsilon} z_j = N_\upsilon, \forall \upsilon \in \Upsilon. \tag{3.21}$$

The variable z_j is simply the occupancy of lattice site j, where $j = 1, 2, \cdots, V$. Constraints 3.16-3.18 define z_j as a binary integer variable. Constraint 3.19 tells us that our system must have N particles. In Constraint 3.20, Θ is the set of all occupied lattice sites in the desired configuration, i.e., $z_{d,i} = 1$. This constraint ensures that $\mathbf{z} \neq \mathbf{z}_d$ by requiring at least one difference in the occupancies of $\mathbf{z}_d$ and $\mathbf{z}$. If we are solving PSEMP, only Constraints 3.16-3.20 are required. Constraint 3.21 is a particle number constraint needed for CEMP. The component α of competing configurations in CEMP is made up of configurations that have a certain number of particles, N_υ, trapped inside the system subvolume, $\upsilon \in \Upsilon$, where Υ represents the set of all defined system subvolumes. Hence, υ represents a system subvolume with an energetically favorable well between energetic barriers that are sufficiently large to isolate the particles within the corresponding subvolume, υ, of the system volume. An example using this type of constraint can be found in Section 3.4.1.

Solving this optimization problem involves iterating between solutions of an LP and a QIP. To decrease computational times and guarantee a global solution, QIP's with binary variables can be linearized, which is discussed in Section 3.2.3.

3.2.3 Linearization of a $0-1$ quadratic problem

Quadratic functions with binary variables can easily be linearized in the following manner. Consider the general potential energy equation:

$$E(\mathbf{z},\mathbf{s}) = \mathbf{s}^{\mathrm{T}}\mathbf{H}\mathbf{z} + \mathbf{z}^{\mathrm{T}}\mathbf{J}\mathbf{z} = \sum_{k=1}^{N_{pc}}\sum_{i=1}^{N} s_k H_{k,i} z_i + \frac{1}{2}\sum_{i,j} z_i J_{ij} z_j. \tag{3.22}$$

When you multiply two binary variables with each other, there are 4 possible outcomes:

$$0 \cdot 0 = 0, \tag{3.23}$$

$$0 \cdot 1 = 0, \tag{3.24}$$

$$1 \cdot 0 = 0, \tag{3.25}$$

$$1 \cdot 1 = 1. \tag{3.26}$$

We can therefore create a new binary variable, z_{ij} for each pair of the original binary variables, z_i and z_j, where $z_{ij} = 0$ if one or both lattice sites i and j are unoccupied and $z_{ij} = 1$ if both lattices sites are occupied. Using this new variable, Equation 3.22 can be recast as follows:

$$E(\mathbf{z},\mathbf{s}) = \sum_{k=1}^{N_{pc}}\sum_{i=1}^{N} s_k H_{k,i} z_{ii} + \frac{1}{2}\sum_{i,j} J_{ij} z_{ij}, \tag{3.27}$$

where

$$z_{ij} \leq z_{ii}, \quad 1 \leq i < j \leq V; \tag{3.28}$$

$$z_{ij} \leq z_{jj}, \quad 1 \leq i < j \leq V; \tag{3.29}$$

$$z_{ij} \geq z_{ii} + z_{jj} - 1, \quad 1 \leq i < j \leq V; \tag{3.30}$$

$$z_{ij} \geq 0, \quad 1 \leq i < j \leq V; \tag{3.31}$$

$$z_{ii} \in \{0,1\}^V, \quad \forall i = 1, 2, \cdots, V. \tag{3.32}$$

In this formulation, z_{ij}, $i \neq j$, is a continuous variable, and z_{ii} is a binary variable. Hence, the number of binary variables remains the same, but new continuous variables and the above constraints are introduced. Due to these additions, recasting the problem does not guarantee a faster solution time. For example, if we have an 8x8 2-D square lattice system or a 64-site 1-D system, the original problem would include 64 binary variables. If we linearize this problem, we would maintain the 64 binary variables but also add 2016 new continuous variables, i.e.,

$$N_{bin}^{new} = N_{bin}^{old}, \tag{3.33}$$

$$N_{cont}^{new} = \frac{N_{bin}^{old}(N_{bin}^{old}-1)}{2}, \tag{3.34}$$

where N_{cont} is the number of continuous variables and N_{bin} is the number of binary variables. The number of additional constraints is

$$N_{const} = 4 \cdot \begin{pmatrix} V \\ 2 \end{pmatrix}. \tag{3.35}$$

Because of this increase in number of variables and constraints, many have researched what additional constraints are needed in order to make linearization a favorable strategy for the solution of quadratic binary integer problems. Adams and Sherali[40] propose one such linearization method for quadratic binary problems with linear constraints.

Let us define a general quadratic integer program with a quadratic objective function and linear constraints:

$$\min_{x} \quad \sum_{i=1}^{m} c_i x_i + \sum_{i,j} D_{ij} x_i x_j \tag{3.36}$$

$$s.t. \quad x \in X, \tag{3.37}$$

$$x \in \{0,1\}, \tag{3.38}$$

where $X \equiv X_1 \cap X_2 \cap X_3$ is the set of constraints:

$$X_1 = \sum_{i=1}^{m} a_{ki} x_i = b_k, k = 1, 2, \cdots, K; \tag{3.39}$$

$$X_2 = \sum_{i=1}^{m} g_{li} x_i \geq h_l, l = 1, 2, \cdots, L; \tag{3.40}$$

$$X_3 = 0 \leq x_i \leq 1, i = 1, 2, \cdots, m. \tag{3.41}$$

X_1 defines the set of equality constraints, X_2 defines the set of inequality constraints, and X_3 bounds the value of x_i to between zero and unity. Adams and Sherali linearize this general quadratic integer program by performing the following operations:

Step 1: Form $m \cdot K$ new constraints by multiplying the K equality constraints in X_1 by each x_j, $j = 1, 2, \cdots, m$.

Step 2: Form $m \cdot L$ new constraints by multiplying the L inequality constraints in X_2 by each x_j, $j = 1, 2, \cdots, m$.

Step 3: Form $m \cdot L$ new constraints by multiplying the L inequality constraints in X_2 by each $(1 - x_j)$, $j = 1, 2, \cdots, m$.

Step 4: Form $3m(m-1)/2$ new constraints by multiplying the m $x_i \geq 0$ inequality constraints in X_3 by each $(1 - x_j)$, $j = 1, 2, \cdots, m, j \neq i$, and by multiplying, for $i = 1, 2, \cdots m - 1$, the $x_i \leq 1$ inequality constraints in X_3 by each $(1 - x_j)$, $j = i + 1, i + 2, \cdots, m$.

Step 5: Substitute $w_{ij} \equiv x_i x_j, \forall (i, j)$, $w_{ij} \geq 0$. Note that w_{ij} needs to only be defined for $i = 1, 2, \cdots, m-1, j = i, i+1, \cdots, m$, reducing the total number of new variables introduced to the problem. Also, $x_i^2 \equiv w_{ii}$.

The linearized general quadratic optimization problem can therefore be written as

follows:

$$
\begin{array}{lll}
\min_w & \sum_{i=1}^{m} c_i w_{ii} + \sum_{i,j} D_{ij} w_{ij} & \\
s.t. & (a_{kj} - b_k) w_{jj} + \sum_{i<j} a_{ki} w_{ij} + \sum_{i>j} a_{ki} w_{ji} = 0, & \forall (j,k); \\
& \sum_i g_{li} w_{ii} - h_l \geq \sum_{i<j} g_{li} w_{ij} + \sum_{i>j} g_{li} w_{ji} + (g_{lj} - h_j) w_{jj} \geq 0, & \forall (j,l); \\
& w_{ii} - w_{ij} \geq 0, & \forall (i,j), \quad i<j; \\
& w_{jj} - w_{ij} \geq 0, & \forall (i,j), \quad i<j; \\
& -w_{ii} - w_{jj} + w_{ij} \geq -1, & \forall (i,j), \quad i<j; \\
& w_{ij} \geq 0, & \forall (i,j), \quad i<j; \\
& \sum_{i=1}^{m} a_{ki} w_{ii} = b_k, & \forall k; \\
& w_{ii} \in \{0,1\}, & \forall i; \\
& w_{ij} \in [0,1], & \forall (i,j) \quad i<j.
\end{array}
\tag{3.42}
$$

The second constraint is a result of performing Steps 2 and 3 on the inequality constraint (Equation 3.40). One can see that this new constraint also includes the old constraint, X_2; therefore, adding the old constraint would be redundant in the linearized problem. Applying this linearization approach to the QIP in the inner problem, we have the following linearized problem:

$$
\begin{array}{lll}
\min_{\mathbf{z}} & E(\mathbf{z}, \mathbf{s}^i) - E(\mathbf{z}_d, \mathbf{s}^i) - \delta^i & \\
s.t. & (1-N) z_{jj} + \sum_{i<j} z_{ij} + \sum_{i>j} z_{ji} = 0, & \forall j; \\
& (N-1) - \sum_{z_{ii} \in \Theta} z_{ii} \geq (N-1) z_{jj} - \sum_{z_{ii} \in \Theta, i \leq j} z_{ij} - \sum_{z_{ii} \in \Theta, i>j} z_{ji} \geq 0, & \forall j; \\
& z_{ii} - z_{ij} \geq 0, & \forall (i,j), \quad i<j; \\
& z_{jj} - z_{ij} \geq 0, & \forall (i,j), \quad i<j; \\
& -z_{ii} - z_{jj} + z_{ij} \geq -1, & \forall (i,j), \quad i<j; \\
& z_{ij} \geq 0, & \forall (i,j), \quad i<j; \\
& \sum_i z_{ii} = N, & \forall i; \\
& z_{ii} \in \{0,1\}, & \forall i; \\
& z_{ij} \in [0,1], & \forall (i,j), \quad i<j; \\
& \sum_{z_{ii} \in \upsilon, i \leq j} z_{ij} + \sum_{z_{ii} \in \upsilon, i>j} z_{ji} - N_\upsilon \cdot z_{jj} = 0, & \forall (j, \upsilon \in \Upsilon); \\
& \sum_{z_{ii} \in \upsilon} z_{ii} = N_\upsilon, & \forall \upsilon \in \Upsilon.
\end{array}
\tag{3.43}
$$

Similar to the original problem formulation, the last two constraints above is used

to define particle number specifications within system subvolumes for CEMP. We have now reduced our problem to two linear problems, the first (outer problem) with continuous variables and the second (inner problem) with binary integer variables.

3.3 System robustness, constraining features and introducing additional degrees of freedom

The value of δ, resulting from optimization problem, is a measure of the desired structure's robustness against statistical fluctuations, a measure of the probability that the nanostructure will remain in the desired geometric configuration. If the value of δ suggests satisfactory robustness, then the Static Problem is solved. If not, then additional system degrees of freedom are needed. Since the initial set of point conditions used thus far is of the same type, i.e., well- or barrier-forming, the additional point conditions will be selected from the opposite type class. The selection of the new point conditions is guided by the solution of the optimization problem.

The computed δ value tells us the minimum energy gap between the competing configurations and the desired configuration. All energy difference values are found in the optimization problem's constraints. The constraints that are active, such that $E(\mathbf{z}, \mathbf{s}) - E(\mathbf{z}_d, \mathbf{s}) = \delta$, represent the configurations that are constraining our objective function from increasing in value. We look at the configurational differences between the constraining configurations and the desired configuration and find the *constraining features* of the desired structure, i.e., the locations in the desired structure where the energetic barriers are easier to overcome, therefore allowing particles to sample out of the desired configuration.

Once the constraining features are found, new point conditions can be added that include the constraining features in their areas of influence (see Section 3.1). Then, the optimization problem can be solved again for an updated δ value. This approach can be repeated until a satisfactory level of robustness is achieved.

3.3.1 Glass transition temperature

Thus far, we have not spoken about the system temperature. Given that we have solved the Static Problem and have computed a δ value, we must determine at what temperature to operate the system, assuming it has reached the desired configuration. We use system simulations at multiple temperatures, all initialized with the desired configuration, to see how the probability of maintaining the desired configuration changes with the system temperature, $k_B T$. The results in Section 3.4 show a second-order phase transition from nonergodic behavior (i.e., the system is trapped in the desired state) and ergodic behavior (i.e., the system follows Boltzmann's probability distribution with all of phase space accessible). This transition from "glassy" to "fluid" is parameterized by a *glass transition temperature*, T_g. This critical temperature is useful in that if we choose an operating temperature, $T_o << T_g$, then we know the system is nonergodic; if we choose $T_o >> T_g$, the system is ergodic; if we choose $T_o \approx T_g$, the system is in transition between ergodic and nonergodic behavior. This transition region simply means that, given an ensemble of system realizations, the temperature is sufficiently high enough for the system to leave the desired state, but the system is not able to fully sample all of phase space. As the system temperature increases, so does the subset of allowable configurations in phase space.

The most widely-used approach to finding the glass transition temperature of a second-order transition involves finding the intersection of the tangents to two "linear" regions of a curve representing the corresponding temperature dependence[41]. From the simulation results for the example systems in Section 3.4, we see second order phase transitions similar to that in Figure 3-5. Using this curve, there are 3 straight-line regions. If we find the intersection of each adjacent pair of straight-line regions to find T_g^+ and T_g^-. These two glass temperature values are different and the region in between them represent the transition region between ergodic and nonergodic behavior.

As a clarifying note, the curve represented in Figure 3-5 does not represent a "cooling" or "heating" process. Classical glass transition temperatures are calculated

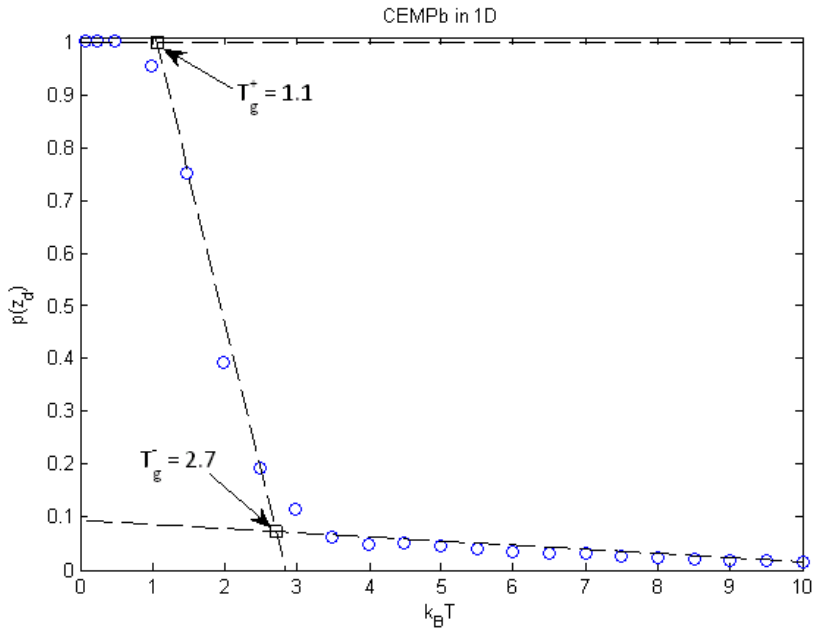

Figure 3-5: The definition of the glass transition temperatures, T_g^+ and T_g^- for the transition between ergodic and nonergodic behavior. The region between the two temperatures is the transition region for the probability of the desired configuration.

from analyzing the system starting from a low (or high) temperature and heating (or cooling) it while monitoring a characteristic parameter that shows a second-order critical transition; this type of study may be a fruitful investigative topic, especially for the Dynamic Problem described in Section 4. However, the glass transition temperature, as described above, is useful for understanding the robustness of the desired structure. In other words, if we operate the system, which has reached the desired structure, at $T_o << T_g^-$, then we know our desired structure is robust due to the system's nonergodic behavior.

3.4 Static problem examples

We show the principles discussed above with 3 example lattice systems, one 1-D system and two 2-D systems. Section 3.4.1 discusses the results of the 1-D example, and Section 3.4.2 discuss the 2-D examples. Other example systems were analyzed (not to the same level of completeness as in the following sections), and these systems

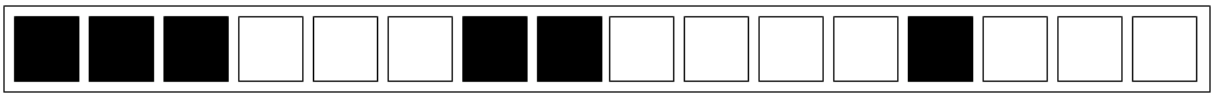

Figure 3-6: The desired configuration for the 1D example system, where $N = 6$ and $V = 16$.

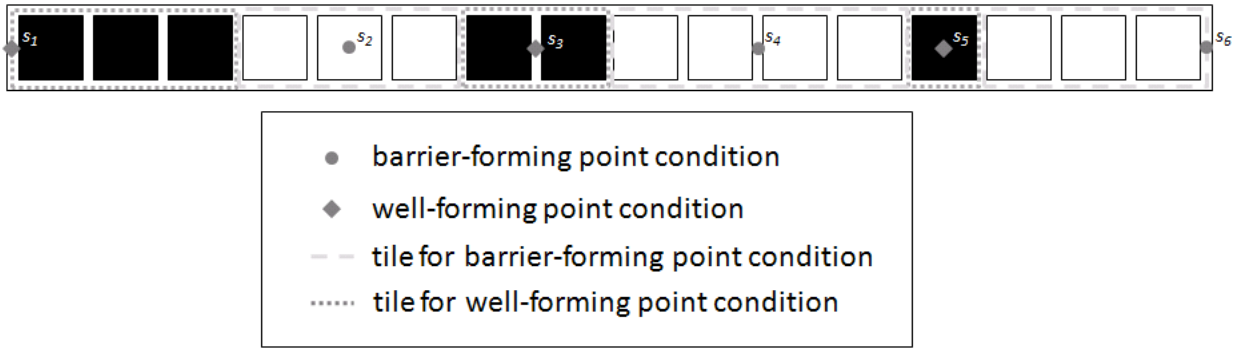

Figure 3-7: The minimum tiling algorithm outputted six point condition locations, three in barrier regions and three in well regions. The tiles represent each point condition's area of influence.

are described in Appendix B.

3.4.1 1D example system

The 1-D example system has particle number $N = 6$ and system volume (i.e., number of lattice sites) $V = 16$. The desired configuration is depicted in Figure 3-6. Using the minimum tiling algorithm, the locations and the respective tiles that define the areas of influence for the well- and barrier-forming point conditions are shown in Figure 3-7.

Given these point condition locations, we solved all variations of the EMP optimization problem. The results are shown in Table 3.1. The strength values were constrained, $-100 \leq s_i \leq 100, \forall i = 1, 2, \cdots, N_{pc}$. The variables $s_{1,3,5}$ are the well-forming point condition strengths, and $s_{2,4,6}$ are the barrier-forming point condition strengths. A strength value of zero simply means that a particular point condition was not considered in the optimization problem. The δ column is the objective function output value. The column with the number of constraints has a different meaning

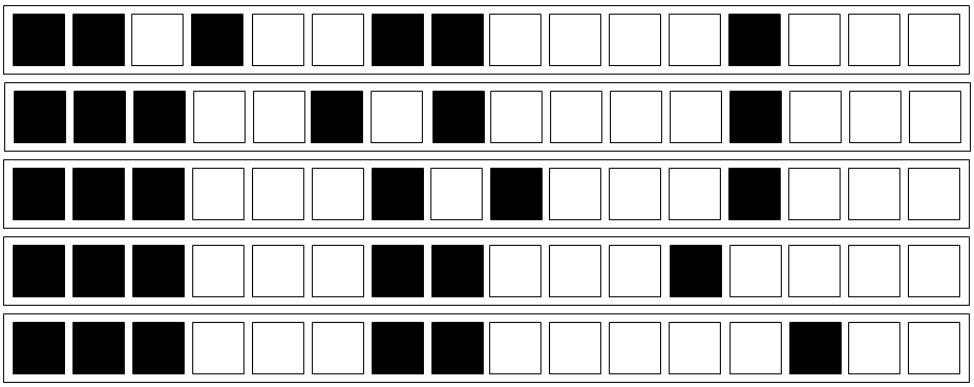

Figure 3-8: The five competing configurations used in the OSEMP formulation, representing configurations that are one step away from the desired configuration.

depending on which specific variation of EMP we are solving. For PSEMP, the number of constraints specifies how many iterations of Blankenship and Falk's method the solver needed to perform before arriving at the global solution. We initialized the outer problem with one constraint (i.e., $\text{card}(\Omega_\alpha^0) = 1$), which has all the particles in lattice sites $1, 2, \cdots, N$ (starting from the left). For OSEMP, the number of constraints is the number of configurations that are one MC step away from the desired configuration. These configurations are depicted in Figure 3-8. For CEMP, the number of constraints represents the subset of configurations within the defined component needed for Blankenship and Falk's method to reach the global optimum. As before we initialized the outer problem with one constraint, which defined a configuration that belonged to the respective component. More specifically, the initial constraint defined a configuration with particles in lattice sites $1, 2, 3, 6, 7$, and 11.

Problems PSEMPw_1D and PSEMPb_1D are the outputs of PSEMP using the well- and barrier-forming point conditions, respectively. As can be seen, the 3 well-forming point conditions provide a larger δ value. Figure 3-9 shows the results of dynamic MC simulations (using the vmmc methods in Section 2.2.1), varying the system temperature with both the strength value outputs of PSEMPw_1D and PSEMPb_1D. The MC simulations were initialized with the desired configuration and were allowed to run for $50,000$ MC steps in order to test the robustness of the desired state. A total of 50 simulation results were averaged to find the average probabilities in

Figure 3-9. The points labeled $E[p^w_{MC}(\mathbf{z}_d)]$ represent the average probability of maintaining the desired configuration using the well-forming point conditions defined in PSEMPw_1D, while the points labeled $E[p^b_{MC}(\mathbf{z}_d)]$ represent the average probability using the barrier-forming point conditions in PSEMPb_1D. We see that the well-forming point conditions provide a more robust desired configuration, evident from the larger δ value, i.e., the system probability decreases from unity at higher temperatures for higher output values of δ.

Figure 3-9 also compares the MC results to the equilibrium Boltzmann probability (Equation 1.3) of the desired configuration, $p_{EQ}(\mathbf{z}_d)$, calculated from a full enumeration of the energies of all system configurations. Looking at the general structure of the MC simulation results, they tend to follow the Boltzmann distribution, i.e., $p_{MC}(\mathbf{z}_d) = p_{EQ}(\mathbf{z}_d) = 1$ at very low temperatures and $p_{MC}(\mathbf{z}_d) \rightarrow p_{EQ}(\mathbf{z}_d)$ at higher temperatures. The fact that the equilibrium probabilities reach unity at low temperatures tells us that our desired configuration is the minimum energy state. The simulated systems are not equilibrium systems as the Boltzmann equation suggests. Because the Boltzmann equation includes all possible system states, it assumes that the system is ergodic, i.e., given a long period of time, $p_{MC}(\mathbf{z}_d) \rightarrow p_{EQ}(\mathbf{z}_d)$. However, the simulated system is non-ergodic at low temperatures and only approaches an ergodic system at very high temperatures. This is the reason why we see differences in the behavior of the probability at intermediate temperatures. This will become more apparent in the OSEMP results.

The transition from non-ergodic to ergodic behavior is a second-order phase transition parameterized by the glass transition temperature. At high temperatures, the system follows the Boltzmann distribution, i.e., the system exhibits ergodic behavior. As you decrease the temperature, a critical temperature is reached where the system begins to exhibit "glassy' behavior, and after the transition, the system is non-ergodic.

The second plot in Figure 3-9 shows the standard deviation of the probability given these 50 realizations of the system, initialized with the desired configuration. We see that within the transition region, there are large deviations between the 50

simulations at each particular temperature. This tells us that the system is becoming trapped in meta-stable system components during the transition region, therefore allowing for different probabilities between different system realizations. At low temperatures, there is low variability because all systems are essentially frozen at the desired configuration. At high temperatures, there is also low variability because the system exhibits ergodic behavior, where all system states are accessible and we are essentially following the Boltzmann distribution.

We solved OSEMP using the well- and barrier-forming point conditions, respectively. The resulting point condition strengths and objective function values are shown in Table 3.1 under OSEMPw1_1D and OSEMPb1_1D. As can be seen, the well-forming point conditions provide the larger δ value. Figure 3-10 shows dynamic MC simulation results for OSEMPw1_1D. Here, $E[p_{MC}^{w1}(\mathbf{z}_d)]$ represent the average probability of maintaining the desired configuration using the well-forming point conditions defined in OSEMPw1_1D, while the points labeled $E[p_{MC}^{b1}(\mathbf{z}_d)]$ represent the average probability using the barrier-forming point conditions in OSEMPb1_1D. Given the results from PSEMP above, we know that the well-forming point conditions will provide a more robust desired structure due to its larger δ value. Again, we compare the simulation results to th Boltzmann equilibrium probability of maintaining the desired configuration, $p_{EQ}(\mathbf{z}_d)$.

We may find that, given a set of point condition locations and strengths, we would like to increase the robustness of our desired configuration, i.e., we would like to increase the probability of maintaining the desired configuration at a specific system temperature. This can be done through the introduction of additional point conditions. Because each point condition provides another system degree of freedom (the point condition strength) within the optimization problem, adding another point condition to a set of point conditions can only increase the system's robustness, i.e.,

$$\delta_{\mathbf{s}_1} \geq \delta_{\mathbf{s}_2 \subseteq \mathbf{s}_1}, \tag{3.44}$$

$$p(\mathbf{z}_d, \mathbf{s}_1) \geq p(\mathbf{z}_d, \mathbf{s}_2). \tag{3.45}$$

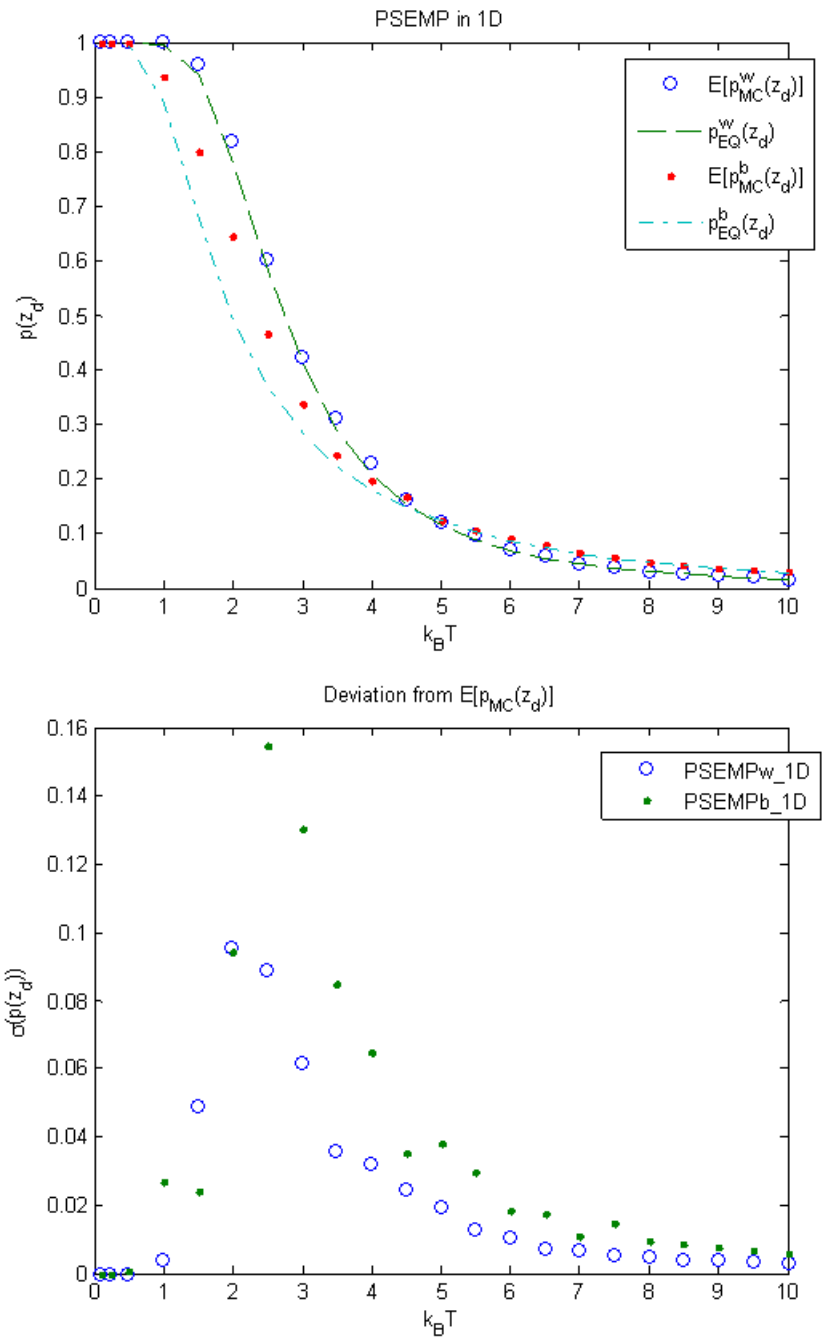

Figure 3-9: Dynamic MC results using the solution to PSEMP for the 1D example system compared to the Boltzmann probability distribution function.

Because OSEMP is an LP, we can look at the optimization problem output and see which additional point conditions will impact the objective function value the most. More specifically, we can analyze the marginal value of all equality and active inequality constraints, where an active inequality constraint simply means that the energy difference is equal to δ. The marginal value specifies how sensitive the objective function value is to the bound or right-hand side of the constraint. The larger the marginal value, the greater the change in the objective function value given a change in the bound. Table 3.2 shows the marginal values of all 5 constraints (representing one step movements away from the desired configuration) for OSEMPw1_1D and OSEMPb1_1D. The column named 'configurational change' provides information on the configurational difference between the desired configuration and the competing configuration being analyzed in each constraint. Non-zero marginal values show us the locations of the constraining features of the desired configuration.

For OSEMPw1_1D, constraints $1, 2$, and 4 are active inequality constraints with marginal values of $0.84, 0.14$, and 0.02, respectively. The constraining features in the desired configuration are the particles in lattice site $3, 7$ and 13. The first constraint provides the largest marginal value. The barrier-forming point condition at lattice site 5 would help decrease the likelihood of the particle in lattice site 3 moving to lattice site 4, i.e., lattice site 4 is in the area of influence of this particular barrier-forming point condition. In fact, both constraints 1 and 2, having positive marginal values, suggest adding the point condition at site 5. Constraints 3 and 4 suggest adding the barrier-forming point condition between sites 10 and 11. Finally, constraint 5 suggests adding the barrier-forming point condition at the right boundary of the system. Problem OSEMPw2_1D has the additional barrier-forming point condition at lattice site 5. The optimized output shows a large increase in the minimum energy gap value, $\delta = 25.27$. Figure 3-10 also shows the MC average probability, $E[p^{w2}_{MC}(\mathbf{z}_d)]$, of maintaining the desired configuration for OSEMPw2_1D. Comparing it to OSEMPw1_1D, we see that it transitions from a non-ergodic to an ergodic system at a higher temperature, i.e., the glass transition temperature for OSEMPw2_1D is greater than that for OSEMPw1_1D. Something to note is that while the Boltzmann

equilibrium probability for OSEMPw1_1D shows that the desired configuration is the phase space minimum energy configuration, this is not the case for the Boltzmann probability, $p_{EQ}^{w2}(\mathbf{z}_d)$, in OSEMPw2_1D. As the temperature decreases, we no longer see an increase in the equilibrium probability, i.e., the minimum energy configuration equilibrium probability goes to unity (assuming the system is ergodic), and this configuration is not the desired configuration. The MC simulation results exhibit non-ergodic behavior by maintaining the desired configuration at low temperatures. As the temperature increases, we are able to sample other configurations. Figure 3-10 also shows the equilibrium probability, $p_{EQ}(\mathbf{z}_d|\Omega_\alpha)$, of maintaining the desired configuration given that the system is sampling within component Ω_α. As can be seen, the desired configuration is the minimum within the component because this probability goes to unity at low temperatures. We also see that as the temperature increases, $p_{MC}(\mathbf{z}_d)$ initially follows $p_{EQ}(\mathbf{z}_d|\Omega_\alpha)$. Then the system temperature increases sufficiently to allow the system to sample beyond Ω_α, and the system eventually approaches ergodic behavior at higher temperatures. Figure 3-10 also shows that, as in PSEMP, deviations in the desired configuration probability also suggest the transition from a non-ergodic system at low temperatures to an ergodic one at high temperatures.

Problem OSEMPw3_1D has the additional point condition between sites 10 and 11, and problem OSEMPw4_1D has the additional point condition at the right boundary. We see that these do not significantly change the objective function value. OSEMPw3_1D has a slightly higher δ value, attributed to the non-zero marginal value of constraint 4. OSEMPw4_1D shows the lowest increase in the objective function due to the fact that constraint 5 is not active. OSEMPw5_1D shows the outcome of using as many point conditions as those needed to account for all the constraining features of the desired configuration.

Performing a similar analysis using OSEMPb1_1D, we see that constraints $2, 4$ and 5 are active, with constraint 5 having the largest marginal value. Both constraints 4 and 5 suggest the addition of a well-forming point condition at lattice site 13. Constraint 2 suggests the addition of the point condition between sites 7 and 8. As

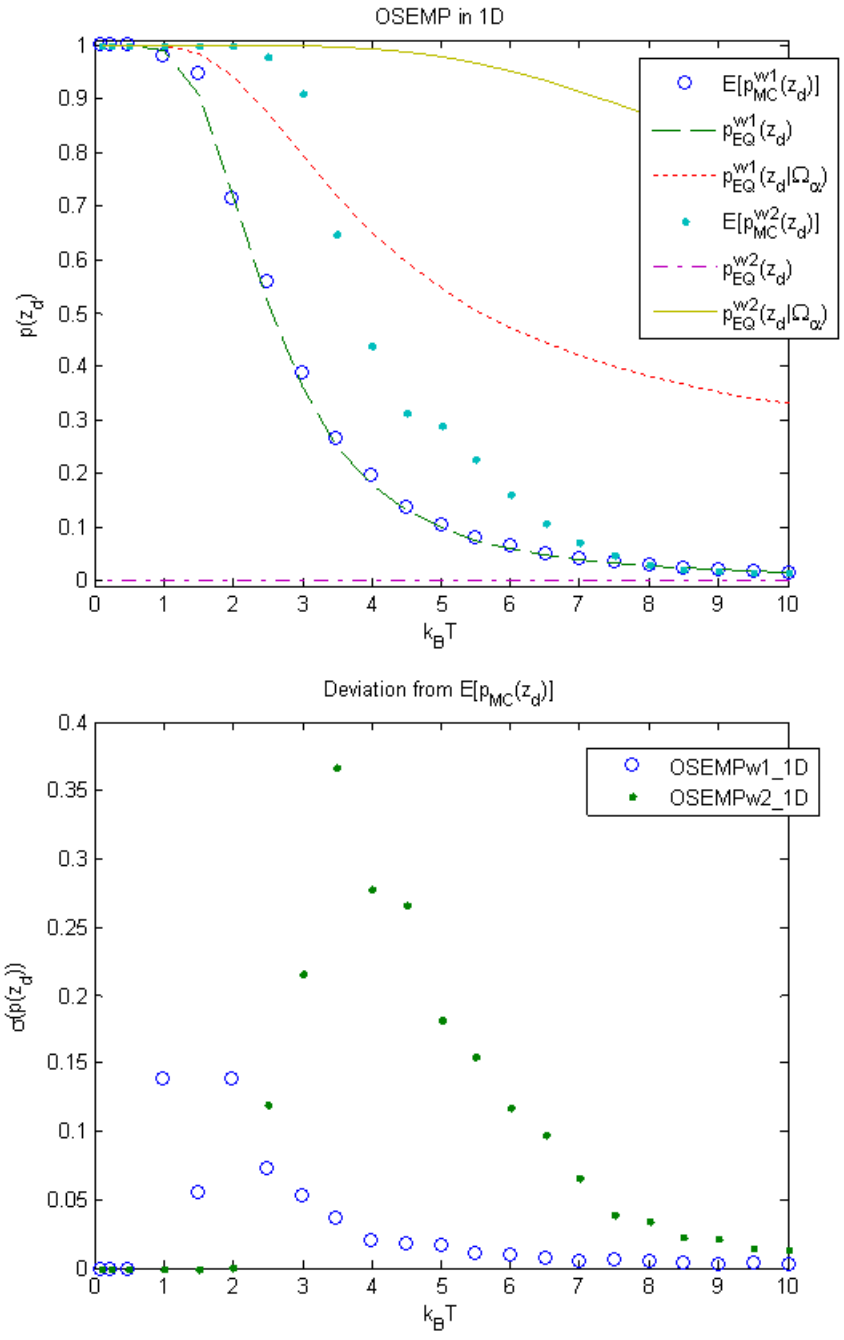

Figure 3-10: Dynamic MC results using the solution to OSEMP for the 1D example system compared to the Boltzmann probability distribution function.

Table 3.1: EMP results for the 1D example system.

Problem	s_1	s_2	s_3	s_4	s_5	s_6	δ	# Constraints
PSEMPw_1D	100	0	30.26	0	33.61	0	7.66	9
PSEMPb_1D	0	-67.15	0	-52.48	0	-100	3.60	7
OSEMPw1_1D	100	0	27.83	0	33.47	0	8.17	5
OSEMPw2_1D	100	-84.49	100	0	96.23	0	25.27	5
OSEMPw3_1D	100	0	45.33	-100	57.45	0	8.88	5
OSEMPw4_1D	100	0	30.02	0	54.29	-100	8.40	5
OSEMPw5_1D	100	-96.24	100	-24.79	100	0	28.77	5
OSEMPb1_1D	0	-29.09	0	-51.27	0	-100	4.24	5
OSEMPb2_1D	0	-69.38	0	-100	47.84	-100	12.99	5
OSEMPb3_1D	0	-30.45	100	-74.29	0	-100	4.68	5
OSEMPb4_1D	100	-40.58	0	-54.55	0	-100	4.35	5
OSEMPb5_1D	0	-100	48.23	-100	86.69	-100	25.30	5
CEMPb_1D	0	-35.84	0	-50.52	0	-100	4.22	7

Table 3.2: Marginal values for the linear constraints using the OSEMP formulation.

Constraint	OSEMPw1 marginals	OSEMPb1 marginals	Configurational change
1	0.84	0	Particle in site 3 moves right.
2	0.14	0.01	Particle in site 7 moves left.
3	0	0	Particle in site 8 moves right.
4	0.02	0.34	Particle in site 13 moves left.
5	0	0.65	Particle in site 13 moves right.

expected, adding the point condition s_5 has a larger impact on the δ value output.

In terms of defining Ω_α, PSEMP uses the entire phase space. However, the combinatorial number of configurations makes solutions for large systems difficult to achieve. Though we are using an implicit enumeration approach, we are still limited by the problem combinatorics. The OSEMP formulation is a means of reducing the computational complexity. As was seen with OESMPw2_1D, this formulation does not require that the desired configuration be the minimum energy state in the entire phase space. Hence, the solution to this problem creates a rugged energy landscape,

where the desired configuration is the minimum only within the set of configurations considered in the constraints, i.e., configurations that are one MC step away from the desired configuration. If we compare the MC simulation results for PSEMPw_1D and OSEMPw1_1D, we see that the former is more robust than the latter. The δ values in Table 3.1 do not suggest this result. However, we have to consider that the δ in PSEMP compares the desired configuration to all other phase space configurations, while the δ in OSEMP only compares the desired configuration to the small subset of configurations that are one step away. These objective function values are therefore not comparable, and it is not surprising that the OSEMP value is higher since this problem is less constrained.

The introduction of external controls leads to the formation of ergodic components within the system phase space. However, it is not possible to define the component that the desired configuration belongs to without knowing the point condition strengths. Hence, PSEMP and OSEMP represent the two extreme problem formulations, the former being the most restrictive and the latter being the least restrictive. However, in certain cases, we may know most, if not all, component configurations because we know where barrier-forming point conditions introduce large barriers to the energy landscape. This allows for the binning of particles within the corresponding subvolumes that define the regions where there are energetic wells. Assuming the system temperature is sufficiently low to lock the particles within the wells between the barriers, this binning of particles defines the set of configurations that approximate the component, which includes the desired configuration. These configurations therefore make up the constraints in CEMP.

Using the 1D desired configuration (Figure 3-6) and the 3 barrier-forming point conditions found through the minimum tiling algorithm, we may define the component as the set of configurations that have three, two and one particle in the wells separated by the three barriers and the system edges, see Figure 3-11. The number of component

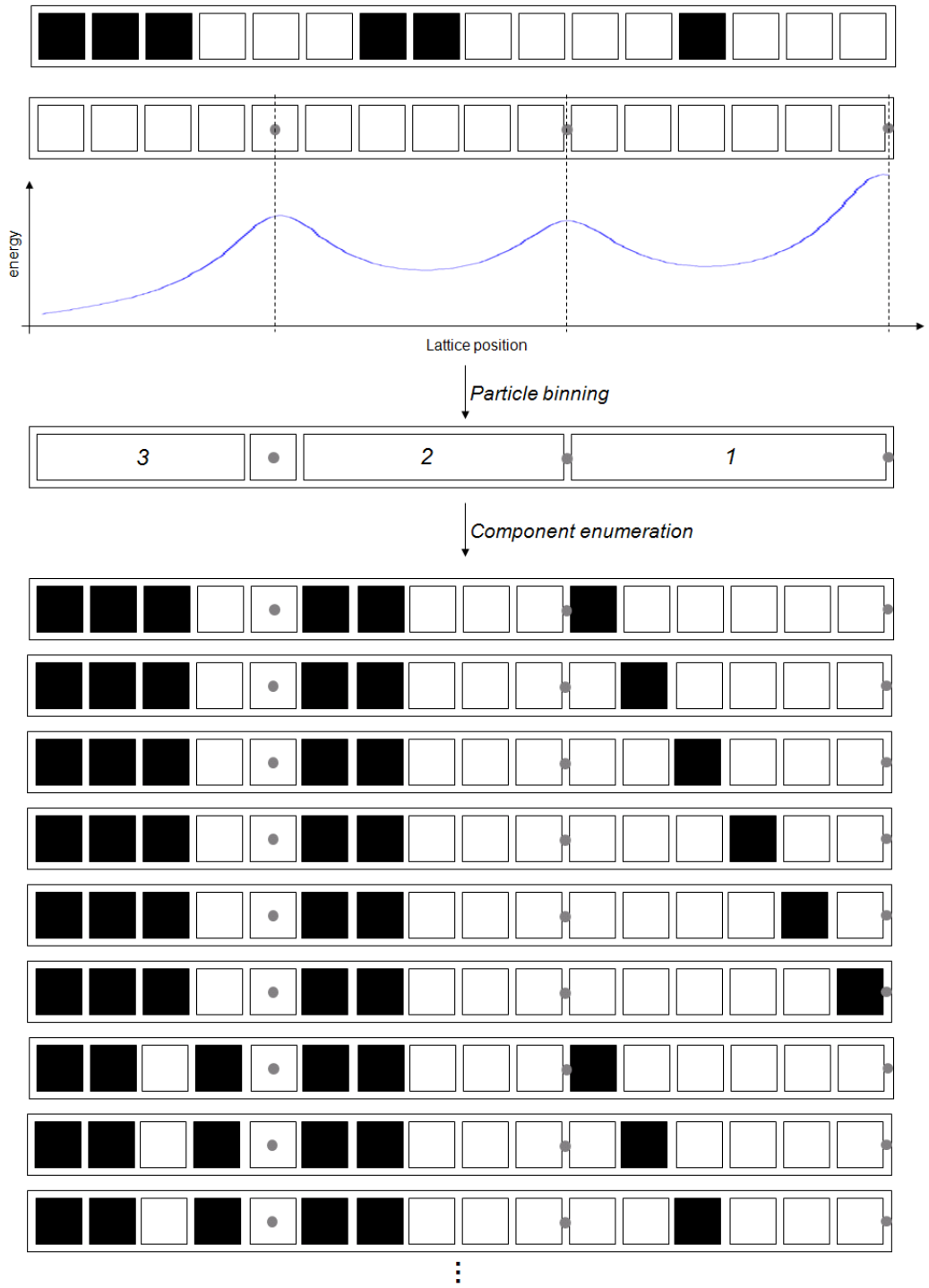

Figure 3-11: The enumeration of all configurations in the ergodic component of the 1D example system. These configurations were used in the CEMP formulation.

states is calculated as follows:

$$\begin{pmatrix} 4 \\ 3 \end{pmatrix} \begin{pmatrix} 5 \\ 2 \end{pmatrix} \begin{pmatrix} 6 \\ 1 \end{pmatrix} = 240. \tag{3.46}$$

OSEMP considers only 5 competing configurations. PSEMP considers all possible configurations in phase space:

$$\begin{pmatrix} 16 \\ 6 \end{pmatrix} = 8,008. \tag{3.47}$$

In the optimization problem formulation of CEMP, the constraints defined by Equation 3.21 are now used. For the example above, the following specific constraints were added:

$$\sum_{j=1}^{4} z_j = 3, \tag{3.48}$$

$$\sum_{j=6}^{10} z_j = 2, \tag{3.49}$$

$$\sum_{j=11}^{16} z_j = 1. \tag{3.50}$$

From Table 3.1, we see the strengths and δ value from solving CEMP for the 1D example system. Figures 3-12 and 3-13 show the results of dynamic MC simulations at various temperatures. As in the results for OSEMP, we see that at low temperatures the system exhibits non-ergodic behavior within its component. However, as the system temperature increases, there is a transition to ergodic behavior, and the system follows the Boltzmann distribution.

The solutions to PSEMP, OSEMP and CEMP show similar results for the Static Problem. Given the optimum strength values that maximize δ, the energy difference between the desired configuration and a set of competing configurations, we see that the system maintains the desired configuration at low temperatures, exhibiting nonergodic behavior. As the temperature increases, there is an eventual transition

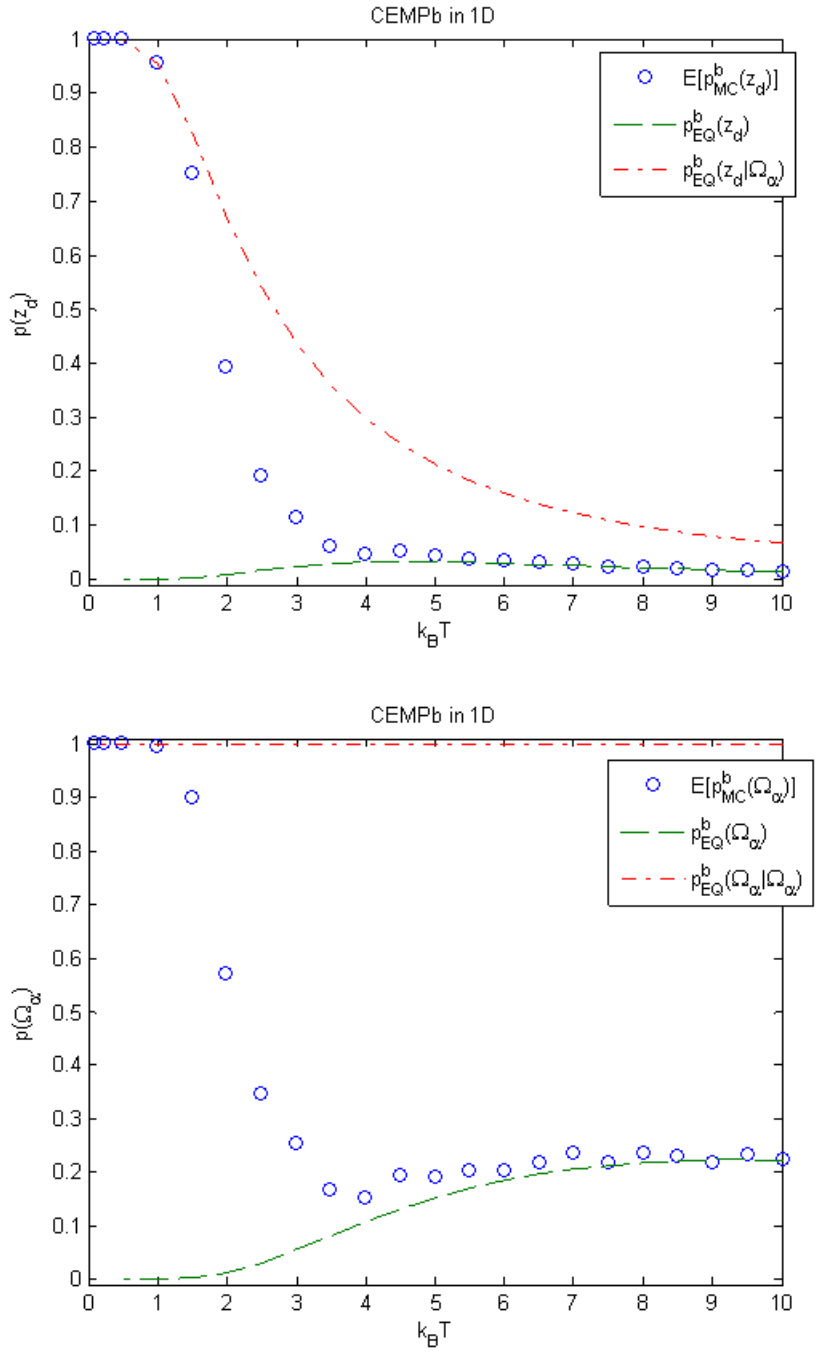

Figure 3-12: Dynamic MC results using the solution to CEMP for the 1D example system compared to the Boltzmann probability distribution function.

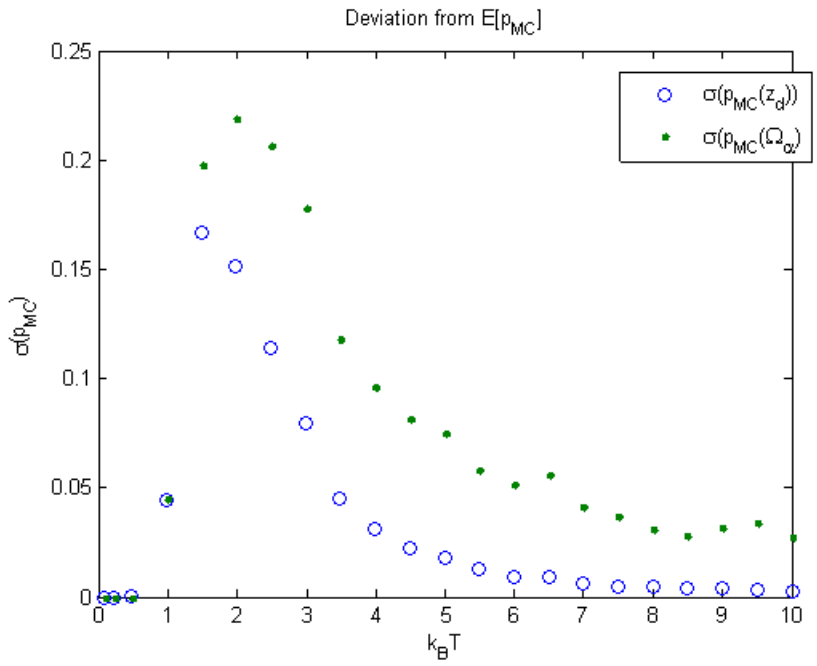

Figure 3-13: Dynamic MC results for the variability of the ensemble probabilities using the solution to CSEMP for the 1D example system.

to ergodic behavior. The larger the δ value, the later this transition, i.e., the glass transition temperature is higher. The transition region also exhibits large deviations in the probability of the desired configuration because the system is able to sample outside the desired component into a subset of neighboring metastable phase space components. From these results, we see that OSEMP is the best problem formulation for the Static Problem. PSEMP is hampered by the combinatorially large number of competing configurations (which represents the number of constraints needed). OSEMP solves with a small subset of the constraints needed in PSEMP with comparable results. Therefore, considering all possible system states in phase space is not a requirement for guaranteeing a robust desired configuration. CEMP requires knowledge of all competing states within the desired configuration's component. In certain situations, such as the 1D example above, this is possible, but without prior knowledge of the point condition strengths, it is difficult to perform the same analysis, especially in higher dimension systems.

3.4.2 2D example systems

We present two 2-D square-lattice example systems and their Static Problem results. The first example has a particle number $N = 7$ and system volume $V = 16$, i.e., a 4-by-4 square-lattice. This has the same number of sites as the 1-D example, and the number of phase space states (11440 total) is still small enough for full enumeration. The second example has a particle number $N = 19$ and system volume $V = 64$, i.e., an 8-by-8 square-lattice. The desired configurations for both systems are depicted in Figure 3-14. The minimum tiling algorithm was used to find the point condition locations, shown in Figure 3-15. Given these point condition locations, we solved both PSEMP and OSEMP; results are shown in Tables 3.3 and 3.4. As in the 1D example, we initialized the outer problem in PSEMP with one constraint that has all the particles in lattice sites $1, 2, \cdots, N$; lattice site numbering is left-to-right and top-to-bottom, as depicted by Figure 3-16. Defining specific phase space components, as was done in the 1D CEMP example, is difficult in 2D without knowledge of the point condition strengths, and therefore we do not present any CEMP solutions.

2D Example 1: $N = 7, V = 16$

Problems PSEMP1w_2D and PSEMP1b_2D are the outputs of PSEMP using the well- and barrier-forming point conditions, respectively. Both sets have similar δ outputs, and Figure 3-17 shows that they follow similar transitions from non-ergodic behavior at low temperatures to ergodic behavior at higher temperatures, i.e., the two systems have similar glass transition temperatures. In this situation, we see that though PSEMP1b_2D has a higher δ value, it is slightly less robust than PSEMP1w_2D. This is due to the fact that δ is only a measure of the energy gap between the desired configuration and the minimum energy state in the rest of phase space. This is an important factor in terms of the robustness of the desired configuration, but we are not considering the energies of all the other configurations in phase space. The distribution becomes especially important when the δ values from two optimization problems are similar. Figure 3-18 shows the energy distributions from both PSEMP1w_2D and

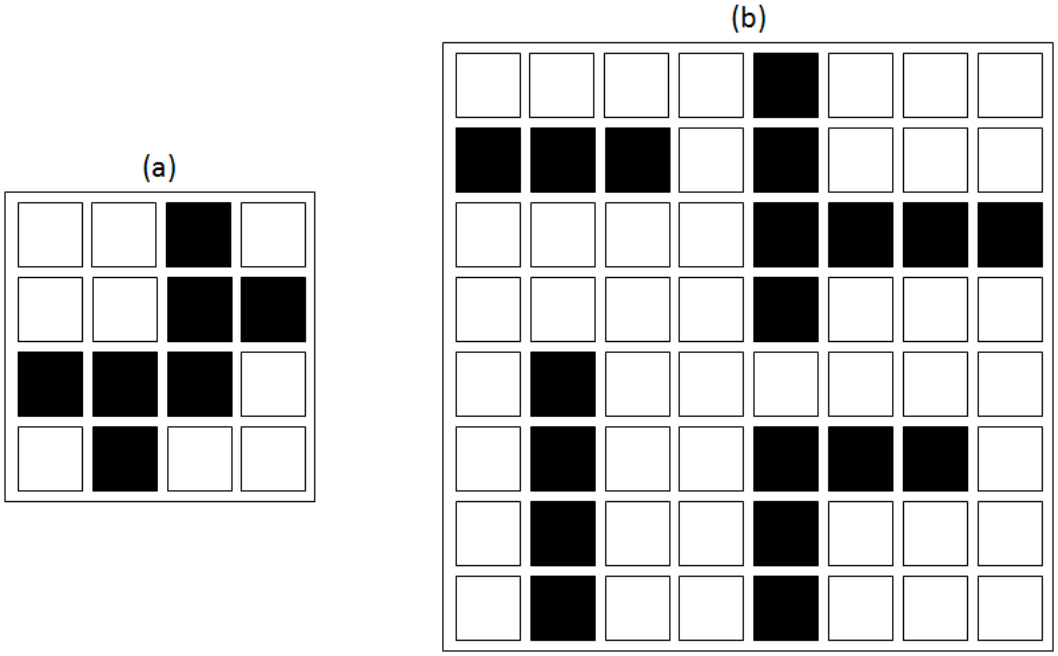

Figure 3-14: The desired configurations for the two 2D example systems: (a) $N = 7, V = 16$, (b) $N = 19, V = 64$.

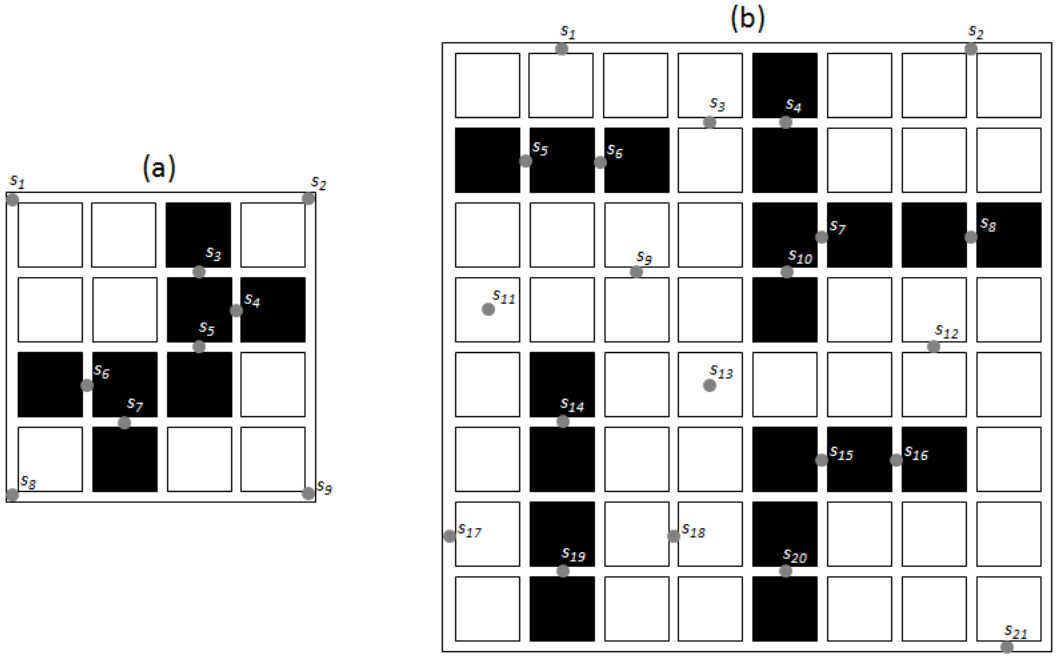

Figure 3-15: The solution to the minimum tiling algorithm for both 2D example systems.

Table 3.3: EMP results for the 2D example system ($N = 7, V = 16$).

Problem	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	δ	# Constraints
PSEMP1w_2D	0	0	13.55	100	-68.95	8.23	83.13	0	0	2.36	18
PSEMP1b_2D	-100	-43.79	0	0	0	0	0	-43.79	-82.30	2.67	17
OSEMP1w1_2D	0	0	100	100	-3.38	100	99.20	0	0	32.17	12
OSEMP1w2_2D	0	0	76.22	100	100	100	100	0	-43.14	33.01	12
OSEMP1b1_2D	-100	-47.62	0	0	0	0	0	-47.62	-93.07	4.22	12
OSEMP1b2_2D	-100	-58.925	0	0	55.40	0	0	-39.34	-100	5.29	12

Table 3.4: EMP results for the 2D example system ($N = 19, V = 64$).

Problem	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}	s_{11}	s_{12}
OSEMP2w1_2D	0	0	0	91.77	100	68.16	42.42	63.67	0	32.06	0	0
OSEMP2w2_2D	0	0	0	90.86	100	68.87	43.02	65.58	0	30.89	0	0
OSEMP2b1_2D	0.04	0.31	1.59	0	0	0	0	0	0.63	0	-0.28	0.98
OSEMP2b2_2D	-40.30	-78.14	-7.85	0	0	0	0	0	-0.96	0	-28.31	-29.23

Problem	s_{13}	s_{14}	s_{15}	s_{16}	s_{17}	s_{18}	s_{19}	s_{20}	s_{21}	δ	# Constraints
OSEMP2w1_2D	0	100	32.67	76.65	0	0	100	86.01	0	8.14	41
OSEMP2w2_2D	0	100	27.76	77.20	0	10.28	100	100	0	8.52	41
OSEMP2b1_2D	0.92	0	0	0	0.32	0.82	0	0	-6.31	0.13	41
OSEMP2b2_2D	-12.16	0	0	13.61	-54.82	-18.71	0	0	-100	0.63	41

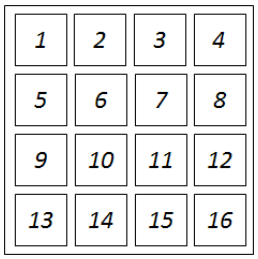

Figure 3-16: The numbering of lattices sites from left-to-right and top-to-bottom for 2D square lattice systems.

PSEMP1b_2D. Both sets of point conditions allow the desired configuration to be the minimum energy configuration. However, the optimum barrier-forming point condition strengths have more configurations with energies closer in value to the minimum. This allows the well-forming point conditions to not only sample out of the desired configuration less, but also allows the system to return with higher probability to the desired configuration.

As was done in the 1D example system, OSEMP was solved using all one-site one particle movements away from the desired configuration, i.e., there were 12 constraints representing all allowed one particle movements for each occupied lattice site. OSEMP1w1_2D and OSEMP1b1_2D use the well- and barrier-forming point conditions used in PSEMP. The marginal values for all 12 constraints in OSEMP1w1_2D show that the constraining feature of the desired configuration is the particle occupying lattice site 8 and its movement to lattice site 12. OSEMP1w2_2D therefore adds the point condition at the bottom right corner of the system volume, s_9. Performing a similar analysis for OSEMP1b1_2D, we find the constraining feature to be the particle in lattice site 7 and its movement to lattice site 6. We therefore add the point condition between sites 7 and 11, s_5, in OSEMP1b2_2D. In both cases, we see an improvement in the δ value. Figure 3-19 shows the dynamic MC simulation results for OSEMP1w1_2D, OSEMP1w2_2D and OSEMP1b1_2D. The well-forming point conditions provide a more robust system. Also, since the δ values are similar for OSEMP1w1_2D and OSEMP1w2_2D, we see similar trends for the desired configuration probability with respect to system temperature.

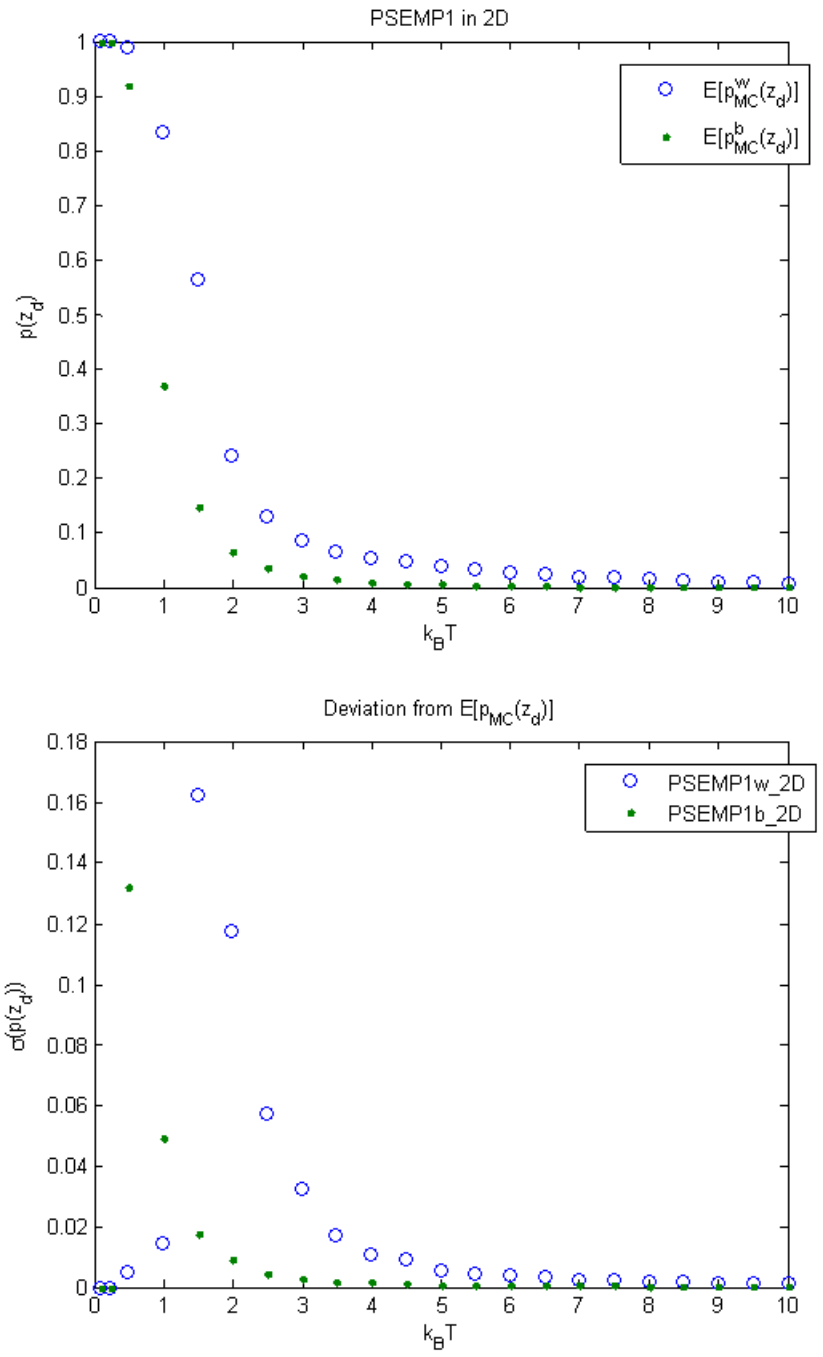

Figure 3-17: Dynamic MC results using the solution to PSEMP for the 2D example system ($N = 7, V = 16$).

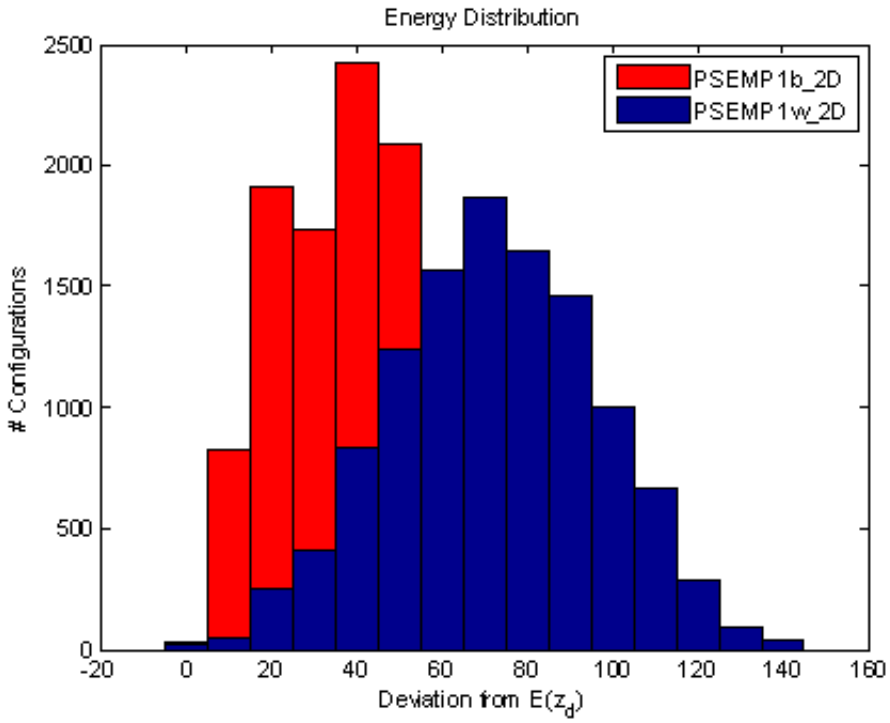

Figure 3-18: Phase space energy distributions for PSEMP1w_2D and PSEMP1b_2D.

2D Example 2: $N = 19, V = 64$

PSEMP was not performed for the larger 2D example system. Even with the Blankenship and Falk approach for implicit enumeration, the increased number of variables and constraints did not allow for a reasonable solution time for the quadratic integer optimization problem, even after using the Adams and Sherali linearization technique. As shown in the 1D and the smaller 2D examples, OSEMP provides a computationally less complex problem with results comparable to PSEMP. Using the well- and barrier-forming point conditions from the minimum tiling algorithm, we solved OSEMP2w1_2D and OSEMP2b1_2D for the optimum point condition strengths with 41 constraints representing all one step single-particle movements away from the desired configuration. The constraining feature in the desired configuration for OSEMP2w1_2D, i.e., the configurational change represented by the constraint with the largest marginal value, is the particle in lattice site 58 and its motion to site 59. We therefore added the barrier-forming point condition s_{18} in OSEMP2w2_2D. Similarly,

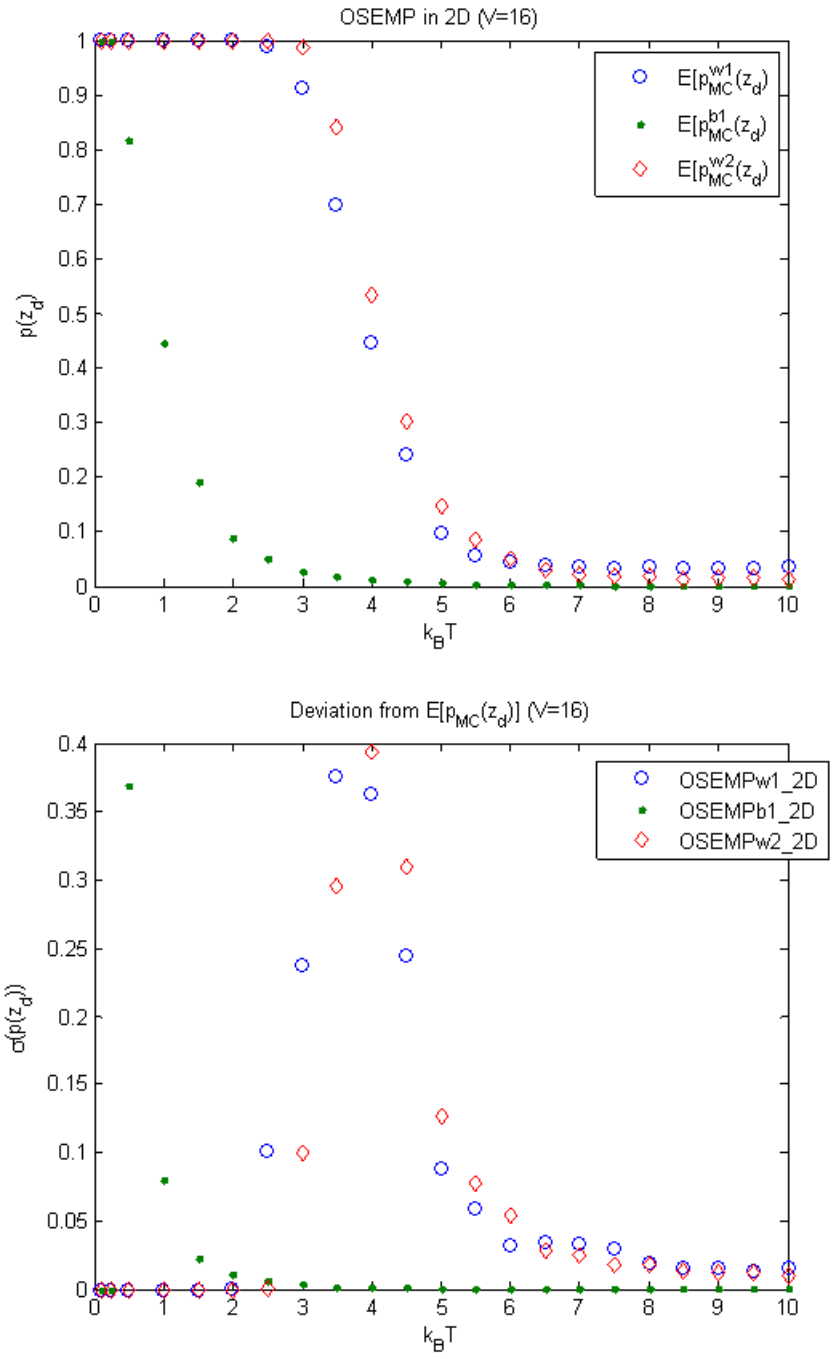

Figure 3-19: Dynamic MC results using the solution to OSEMP for the 2D example system ($N = 7, V = 16$).

the constraining feature for OSEMP2b1_2D is the particle in lattice site 47 moving to site 48. OSEMP2b2_2D therefore includes s_{16}, which has site 47 in its area of influence. Figure 3-20 shows dynamic MC results for OSEMP2w1_2D and OSEMP2b2_2D, which both have the same number of point conditions but very different robustness results.

3.5 Current technology and the imposed limitations on the desired structure

It is important to clarify the technological limitations constraining the creation of external controls on the physical domain in the examples above, which are phenomenological models for templated self-assembly systems. There are several practical approaches to creating these external controls:

Self-assembled monolayers (SAMs): SAMs are ordered assembly of organic molecules on the surface of metals, metal oxides and semiconductors. This technology is therefore limited by the minimum feature sizes of lithographic techniques. Current high-throughput semiconductor technology is at the 32nm node, which creates features with a characteristic lengthscale of ∼32nm[42, 43]. Also, techniques for the removal (and replacement) of SAMs allow for resolutions of ∼25nm[44].

Electrostatic templates: Surfaces with varying electronic charge densities have been studied for the templated self-assembly of charged particles and molecules. Electrical microcontact printing allows for feature sizes of ∼50nm[45, 46], while electron beam irradiation allows for feature sizes of ∼5-20nm that are ∼50nm apart[47]. Ion beams can also be used, but with larger feature sizes of ∼6μm[48]. Atomic force microscopy (AFM) with a conducting tip can be use to create charged features of ∼30nm resolution[49].

Electrodes created by electron beam lithography: Electron beam lithography is capable of creating nanopores of <5nm in size[50], which can be used to create

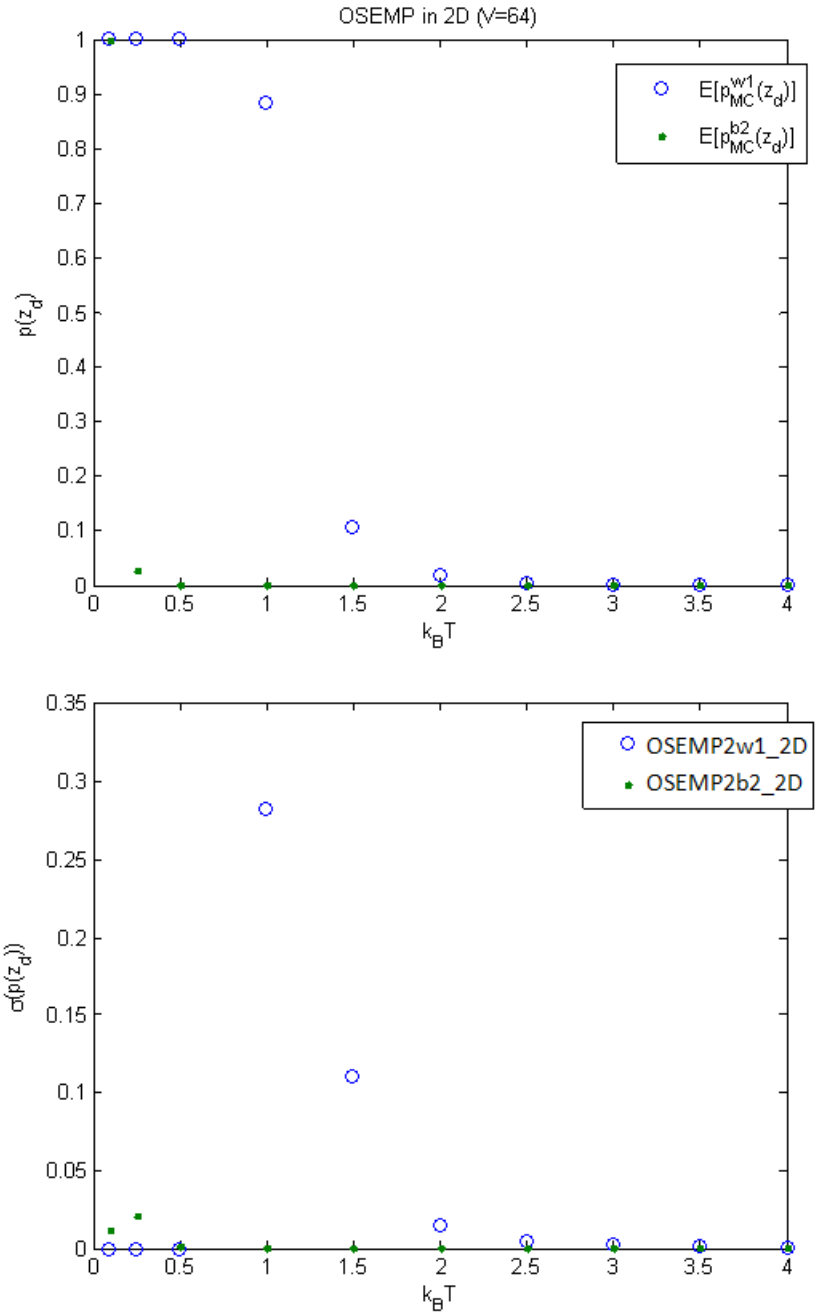

Figure 3-20: Dynamic MC results using the solution to OSEMP for the 2D example system ($N = 19, V = 64$).

electrodes that introduce electric fields to attract/repel particles[13]. However, due to scattering, a radius of 50-100nm around the nanopore shows "wedge" effects, i.e., the pores would need to be >100nm apart.

These techniques all allow for the control of the locations of energetic wells and barriers on the system potential energy surface. Because the energetic wells may be steep, they may allow for the focused placement of the constituents into precise regions that are smaller in length scale than the control features on the surface. For example, Koh[51] has shown the precise placement of nanoparticles (with ∼5nm precision) in a line between differentially charged SAMs with a characteristic lengthscale of 100nm.

Using this understanding of research literature, current technology allows for controls that are about 20-50nm in size with about 20-50nm of space between them. This helps us analyze the sizes of our example lattice model systems and the point conditions we impose on them to achieve our desired configurations. For instance, let us analyze the 1D example system in Figure 3-6 and the point conditions we impose on the system in Figure 3-7. If the δ value from OSEMPw1_1D is sufficient, the minimum distance between the 3 point condition locations is roughly 5.5 lattice sites. If this distance is taken as about 20nm, then we know that the system 1D volume is about 58nm, with about 3.6nm between each lattice site. This system can therefore model a 1D system on a line of length ≥58nm with nanoparticles of diameter ≥3.6nm. If we needed to introduce the additional point condition in OSEMPw2_1D to improve system robustness, the minimum distance between the 4 point conditions is now 2.5 lattice sites. Given current technology, this would be able to model a 1D system on a line of length ≥128nm with nanoparticles of diameter ≥8nm. Any systems that are smaller than these respective dimensions would not currently be manufacturable.

Given the 2D example system in Figure 3-14 and their respective point conditions in Figure 3-15, we can perform a similar analysis. For the example system in Figure 3-14(a), OSEMP1w1_2D outputs a large δ value. However, these point conditions have a small minimum distance of about 0.7 lattice sites. This system can only be created if it had the following dimensionalities: ≥12,800nm^2 system volume ($L \geq$ 113nm), and ≥28nm-diameter nanoparticles. Similarly, OSEMP2w1_2D also has a

minimum distance of about 0.7 lattice sites for the desired configuration in Figure 3-14(b). This requires the following system dimensionalities: $\geq$51,200nm^2 system volume ($L \geq$ 226nm), and $\geq$28nm-diameter nanoparticles. Any systems that are smaller in size would require new technology for the placement of system controls.

Chapter 4

The dynamic problem

In the Static Problem, we were able to reduce the phase space combinatorics to only the neighboring configurations that were one step away, and we showed that these considerations were sufficient to guarantee a robust desired structure. However, we assumed in Chapter 3 that we were able to arrive at the desired structure. Given that we have introduced nonergodicity to the system energy landscape, i.e., we have created a rough energy landscape that traps the system in subsets of phase space (or components), we know that arriving at the desired configuration from any initial configuration can only occur if we are in the desired configuration's component. Therefore, we must design a process that allows the system to reach the desired configuration with high probability: a robust dynamic path.

The proposed dynamic process is inspired by protein folding studies[32, 33, 52, 53]. More specifically, we systematically modify the degrees of freedom in a manner that shrinks the desired configuration's component in time, starting from the entire phase space to a component with just the desired configuration as its sole member. Here, ergodicity plays a large role in guaranteeing that we are trapping the system in the desired component.

4.1 Systematic shrinking of accessible phase space states

There are many ways of systematically breaking the phase space up into components. For instance, in a lattice system, one can use site occupancy as a means of systematically isolating the desired configuration. Let us analyze this approach for the 1D example system in Chapter 3 (Figure 3-6). The desired configuration has 6 particles in lattice sites $1, 2, 7, 8$, and 13. We can start by specifying that we would like to isolate all configurations with a particle in lattice site 1, i.e., our desired component is made up of all configurations with site 1 occupied. This component has the following number of configurations:

$$\begin{pmatrix} V-1 \\ N-1 \end{pmatrix}. \tag{4.1}$$

This represents all possible ways of having the $N-1$ particles occupy the remaining $V-1$ lattice sites. Given this, we then move to the next stage, where we would like to isolate all configurations with lattice sites 1 and 2 occupied. We continue to do this in the same fashion, in stages, until we have specified all 6 occupied lattice sites in the desired configuration, therefore isolating the desired configuration as the sole member of a system component.

If we now consider the process of modifying the degrees of freedom such that we isolate the group of configurations that belong to the desired component, we see an inherent flaw in this approach. The problem lies in the dissimilarities between the configurations that make up the desired component in the self-assembly process. Figure 4-1 shows four configurations that belong to the first isolated component (i.e., all configurations with lattice site 1 occupied) in the 1D example. To maximize the probability of being in this component, we may look at the Boltzmann distribution:

$$p(\Omega_\alpha^{(i)}) = \frac{\sum_{\mathbf{z}_j \in \Omega_\alpha^{(i)}} e^{-\beta E(\mathbf{z}_j)}}{\sum_{\mathbf{z}_k \in \Omega_\alpha^{(i-1)}} e^{-\beta E(\mathbf{z}_k)}}, \tag{4.2}$$

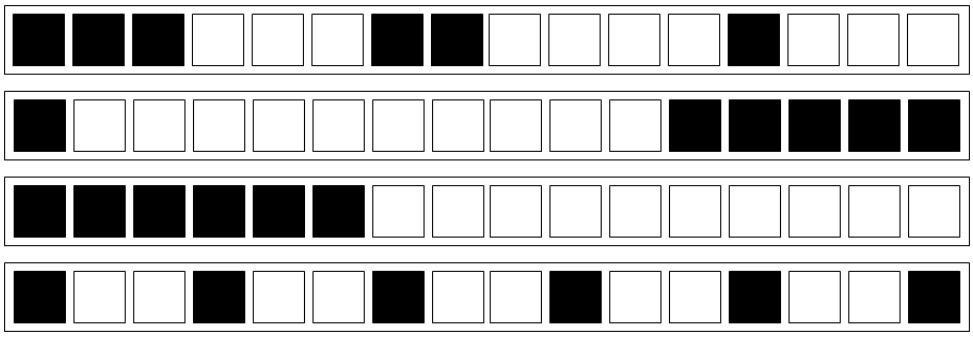

Figure 4-1: Four representative configurations of the set of configurations with an occupied lattice site 1.

where we are essentially comparing the collective energies of the current stage's set of desired configurations, $\Omega_\alpha^{(i)}$, to the previous stage's set of desired configurations, $\Omega_\alpha^{(i-1)}$. In Stage 1 of the 1D example, the numerator consists a sum over all configurations with lattice site 1 occupied ($\Omega_\alpha^{(1)}$), and the denominator represents all configurations in phase space ($\Omega_\alpha^{(0)}$). To maximize this probability, we have to minimize the energies of all the configurations that make up the component compared to the other configurations that belong to the previous stage's component. Because, as depicted in Figure 4-1, the configurations that belong to the desired component vary greatly from each other, they would require different degrees of freedom to minimize their energies. The necessary degrees of freedom for each individual configuration are therefore competing with each other to maximize the probabilities of each individual configuration within the component. This competition decreases the overall probability of the component and is therefore undesirable. One can imagine that we can assign a dominant configuration within the component and use the degrees of freedom needed to maximize the probability of that particular configuration, but this creates a "needle-in-the-haystack" situation, where the one optimized state is difficult to find in a reasonable time and the system may become trapped in an undesirable local minimum state.

Therefore, the method used to isolate sets of configurations is crucial; we must divide the phase space into sets of similar configurations that require similar degrees of freedom to optimize their probabilities. In other words, the energies (and there-

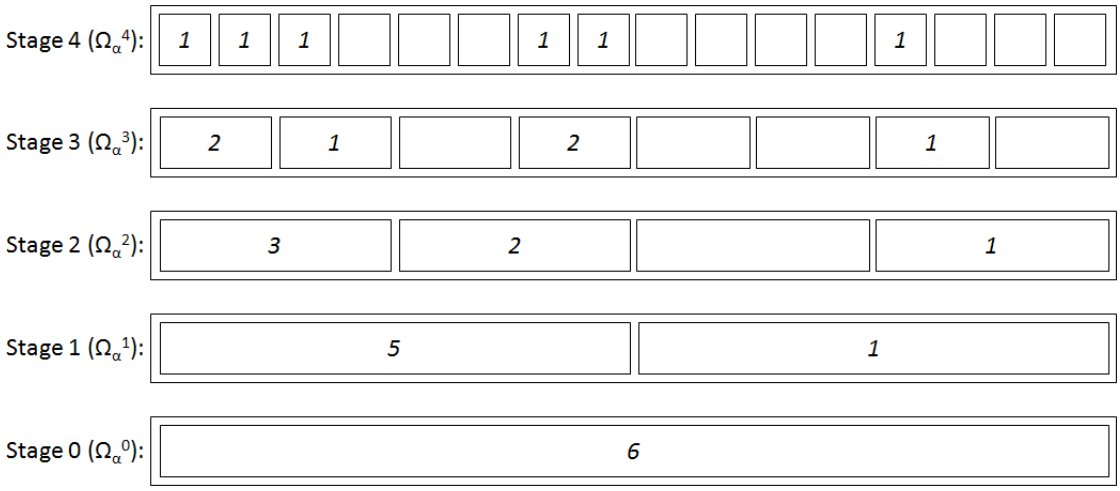

Figure 4-2: The multiresolution distribution of particles for the Dynamic Problem solution.

fore probabilities) of the configurations in a particular component must be highly correlated.

We propose a multiresolution view of the system particle number as a means of systematically breaking down the phase space. Looking once again at the 1D example lattice system, we can coarsen the particle number specification from one-site specifications to two-site specifications. Then, we can look at neighboring groups of two-site specifications and specify four-site specifications. If we continue to do this for the 1D example, we'll have 5 stages of coarse-graining, see Figure 4-2. Starting at Stage 0, we specify 6 particles in the 16-site 1D volume. Given that our system is in the Canonical prescription, we know this has a probability of unity, i.e., we will always have 6 particles within the system volume.

If we now increase the resolution of the system to Stage 1, we isolate the system configurations that have 5 particles in the first 8 lattice sites and 1 particle in the last 8 lattice sites. This represents the first restriction on the allowable system states. Starting from an ergodic system in Stage 0, we now want to transition to a Stage 1 system that guarantees a $(5, 1)$ particle number distribution. To do this, we maximize the probability of all configurations within $\Omega_\alpha^{(1)}$ with respect to the entire phase space

(since the system begins as an ergodic one at Stage 0):

$$\max p(\Omega_\alpha^{(1)}) = \max \frac{\sum_{\mathbf{z}_j \in \Omega_\alpha^{(1)}} e^{-\beta E(\mathbf{z}_j)}}{\sum_{\mathbf{z}_k \in \Omega_\alpha^{(0)}} e^{-\beta E(\mathbf{z}_k)}}. \tag{4.3}$$

The self-assembly process should proceed in a manner such that the system is now nonergodic in Stage 1, and the only accessible configurations belong to $\Omega_\alpha^{(1)}$. Given this, we now transition to a Stage 2 system, where we maximize the probability of having a $(3, 2, 0, 1)$ particle distribution:

$$\max p(\Omega_\alpha^{(2)}) = \max \frac{\sum_{\mathbf{z}_j \in \Omega_\alpha^{(2)}} e^{-\beta E(\mathbf{z}_j)}}{\sum_{\mathbf{z}_k \in \Omega_\alpha^{(1)}} e^{-\beta E(\mathbf{z}_k)}}, \tag{4.4}$$

where the normalizing factor in the denominator is over all Stage 1 states. Because we have already maximized the probability of being in Stage 1, we assume here that the starting system state is within the subset of configurations $\Omega_\alpha^{(1)}$. Maximizing the probability in Equation 4.4 will push the system towards a state in $\Omega_\alpha^{(2)}$. We continue with Stage 3 and 4, maximizing the following two probabilities:

$$\max p(\Omega_\alpha^{(3)}) = \max \frac{\sum_{\mathbf{z}_j \in \Omega_\alpha^{(3)}} e^{-\beta E(\mathbf{z}_j)}}{\sum_{\mathbf{z}_k \in \Omega_\alpha^{(2)}} e^{-\beta E(\mathbf{z}_k)}}, \tag{4.5}$$

$$\max p(\Omega_\alpha^{(4)}) = \max \frac{\sum_{\mathbf{z}_j \in \Omega_\alpha^{(4)}} e^{-\beta E(\mathbf{z}_j)}}{\sum_{\mathbf{z}_k \in \Omega_\alpha^{(3)}} e^{-\beta E(\mathbf{z}_k)}}. \tag{4.6}$$

In Stage 4, we have maximized the probability of our system being in the desired configuration, which is the sole member of the subset of configurations $\Omega_\alpha^{(4)}$. After we have reached this desired configuration, we may now impose the optimized degrees of freedom generated from the Static Problem's OSEMP. Figure 4-3 portrays the dynamic path through phase space described above. As can be seen, the desired configuration is always a member of each stage's set of desired states, and we are restricting the accessible states at each stage until we reach the final stage, where we have isolated the desired configuration.

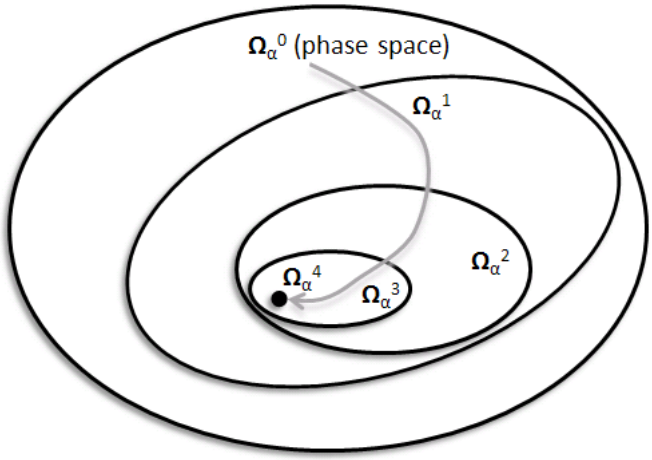

Figure 4-3: The dynamic path that restricts the system to progressively smaller subsets of the system's phase space.

4.2 Qualitatively shaping the energy landscape at each stage of the dynamic process

Now that we have defined the appropriate problems to solve in Equations 4.3-4.6, we must also define the degrees of freedom needed for each stage. In terms of the point condition locations, we may use two approaches: (1) Redefine the point condition locations at each stage to fit the desired particle number specification of each particular stage, (2) Utilize the same set of point conditions as the Static Problem. Figure 4-4 shows what the system designer may choose as locations for point conditions at each stage, given that the locations are redefined based on the coarse-grained particle number specifications at each stage. Although we do not rigorously define a methodology for redefining the point condition locations, the general principle for placement is similar to the Static Problem's minimum tiling algorithm; we place point conditions in the general desired locations of wells and barriers, given the desired configuration at each particular level of detail.

We may also choose to utilize the same point condition set as the one found using the minimum tiling algorithm for the desired configuration. Let us analyze the probabilities at each stage of the 1D example, comparing these probabilities to the

Figure 4-4: Possible locations of point conditions for each stage of the dynamic process.

ergodic Boltzmann probabilities. If we assume the system is ergodic, the Stage 4 desired configuration probability takes the following form:

$$p_{EQ}(\Omega_\alpha^{(4)}) = \frac{\sum_{\mathbf{z}_j \in \Omega_\alpha^{(4)}} e^{-\beta E(\mathbf{z}_j)}}{\sum_{\mathbf{z}_k \in \Omega_\alpha^{(0)}} e^{-\beta E(\mathbf{z}_k)}}. \tag{4.7}$$

We know that at any given temperature, the normalizing sum in the denominator of this equation is always greater than the sum in the denominator of Equation 4.6. Therefore,

$$p_{EQ}(\Omega_\alpha^{(4)}) \leq p(\Omega_\alpha^{(4)}). \tag{4.8}$$

Using the same reasoning, we can generalize this inequality for all stages:

$$p_{EQ}(\Omega_\alpha^{(i)}) \leq p(\Omega_\alpha^{(i)}). \tag{4.9}$$

Similarly, we can also compare the ergodic Boltzmann probabilities at each stage, and

see that they follow the following inequalities:

$$\frac{\sum_{\mathbf{z}_j \in \Omega_\alpha^{(4)}} e^{-\beta E(\mathbf{z}_j)}}{\sum_{\mathbf{z}_k \in \Omega_\alpha^{(0)}} e^{-\beta E(\mathbf{z}_k)}} \leq \frac{\sum_{\mathbf{z}_j \in \Omega_\alpha^{(3)}} e^{-\beta E(\mathbf{z}_j)}}{\sum_{\mathbf{z}_k \in \Omega_\alpha^{(0)}} e^{-\beta E(\mathbf{z}_k)}} \leq \frac{\sum_{\mathbf{z}_j \in \Omega_\alpha^{(2)}} e^{-\beta E(\mathbf{z}_j)}}{\sum_{\mathbf{z}_k \in \Omega_\alpha^{(0)}} e^{-\beta E(\mathbf{z}_k)}}$$
$$\leq \frac{\sum_{\mathbf{z}_j \in \Omega_\alpha^{(1)}} e^{-\beta E(\mathbf{z}_j)}}{\sum_{\mathbf{z}_k \in \Omega_\alpha^{(0)}} e^{-\beta E(\mathbf{z}_k)}} \leq \frac{\sum_{\mathbf{z}_j \in \Omega_\alpha^{(0)}} e^{-\beta E(\mathbf{z}_j)}}{\sum_{\mathbf{z}_k \in \Omega_\alpha^{(0)}} e^{-\beta E(\mathbf{z}_k)}} = 1; \tag{4.10}$$

$$p_{EQ}(\Omega_\alpha^{(4)}) \leq p_{EQ}(\Omega_\alpha^{(3)}) \leq p_{EQ}(\Omega_\alpha^{(2)}) \leq p_{EQ}(\Omega_\alpha^{(1)}) \leq p_{EQ}(\Omega_\alpha^{(0)}) = 1. \tag{4.11}$$

This tells us that if we can guarantee a large value for $p_{EQ}(\Omega_\alpha^{(4)})$, then we can guarantee that the degrees of freedom used can guarantee a high probability at each stage of the dynamic process. However, guaranteeing that we can create this "needle-in-the-haystack" is not feasible for large systems due to the combinatorially large number of configurations in phase space. Because of this reasoning, though, we may move forward with the same approach of determining the locations of the point conditions as the Static Problem, i.e., use the minimum tiling algorithm in Section 3.1, because this approach defines the minimum set of degrees of freedom needed to shape the system energy landscape such that the desired configuration is the energetic minimum. We can proceed in a similar fashion as in the Static Problem, in that if the subset of point conditions used do not seem to provide a robust self-assembly process from any initial configuration, we can analyze what additional degrees of freedom are needed in order to increase robustness by determining the "constraining features" of the desired structure (see Section 3.3). This can be done by examining what competing configurations, in each stage of the self-assembly process, trap the system in an undesirable system state.

4.3 Quantitatively shaping the energy landscape at each stage of the dynamic process

The potential energy equation we utilize involves knowing both the locations and intensities of the system point conditions. In the Static Problem, the minimum tiling algorithm provides the locations of the point conditions, and we solved variants of the Energy-gap Minimization Problem (EMP) to find the point condition intensities (or strengths) that, given the point condition locations, maximize the desired configuration's robustness. EMP essentially uses energy differences between the desired configuration and a set of competing configurations and attempts to maximize the energy difference between the desired configuration and the minimum energy state of the competing configurations. This approach allowed us to linearize the problem formulation while still guaranteeing a robust desired structure. One can imagine formulating a similar problem for the dynamic problem:

$$\begin{array}{ll} \max_{\mathbf{s}\in\mathbf{S}} & E^*(\mathbf{s}) - E'(\mathbf{s}) \\ s.t. & E^*(\mathbf{s}) = \min_{\mathbf{z}\in\Omega_\alpha^{(i-1)}\setminus\Omega_\alpha^{(i)}} E(\mathbf{z},\mathbf{s}), \\ & E'(\mathbf{s}) = \max_{\mathbf{z}_d\in\Omega_\alpha^{(i)}} E(\mathbf{z}_d,\mathbf{s}). \end{array} \tag{4.12}$$

In this optimization problem formulation, we are trying to maximize the energy gap between the configuration in the competing set with minimum energy and the configuration in the desired set with the maximum energy. As in the Static Problem, we can pose this as a combinatorially-constrained problem:

$$\begin{array}{ll} \max\limits_{\mathbf{s}\in\mathbf{S},\delta} & \delta \\ s.t. & E(\mathbf{s},\mathbf{z}) - E(\mathbf{s},\mathbf{z}_d) \geq \delta, \forall(\mathbf{z},\mathbf{z}_d) \in \Omega_\alpha^{(i-1)}\setminus\Omega_\alpha^{(i)} \times \Omega_\alpha^{(i)}. \end{array} \tag{4.13}$$

Many different types of energy distributions can arise when solving this problem; Figure 4-5 highlights four possible outcomes. The first (a) shows an ideal situation where δ is positive and all states in the set of desired configurations have low energies compared to the competing states. The second possible outcome (b) has a negative

δ value, where the maximum energy configuration in the desired component is in the range of the competing configurations' energies. However, there exists at least one configuration in the desired set with energy lower than the minimum competing energy. This is less ideal, but because the minimum energy configuration belongs to the desired component, then this can produce a high probability of achieving the desired state at that particular stage of the self-assembly process. Figure 4-5(c) shows a situation where the δ value is zero, which can be advantageous for a robust dynamic process because the minimum also belongs to the desired component. It is energy distributions such as the one depicted in Figure 4-5(d) that causes problems. In this case, the δ value is also negative, but the minimum energy state belongs to the competing set of configurations. This would not give a high probability of achieving a configuration in the desired set. We cannot protect against this type of distribution by only analyzing the minimum competing and maximum desired configurations. Therefore, the only way to use EMP is to guarantee that $\delta \geq 0$. Using the multiresolution approach defined in Section 4.1, however, does not allow for this guarantee, especially in the earlier stages of the dynamic process. Figure 4-6 shows an example that portrays this difficulty. In essence, we cannot always guarantee that all configurations in a desired component are energetically more favorable (i.e., lower in value) than the most energetically favorable competing configuration.

Stepping back from the EMP formulation, we now have to work with the original non-linear problem formulation (Equations 4.3-4.6). However, this non-linear form makes it difficult to numerically guarantee a global optimum solution. Also, because of the combinatorially large number of configurations in the normalizing factor, numerical optimization techniques no longer pose a possible solution. Therefore, we propose a genetic algorithm approach that searches the parameter space for the point condition strengths and uses dynamic MC simulations to calculate the "fitness" of a potential solution.

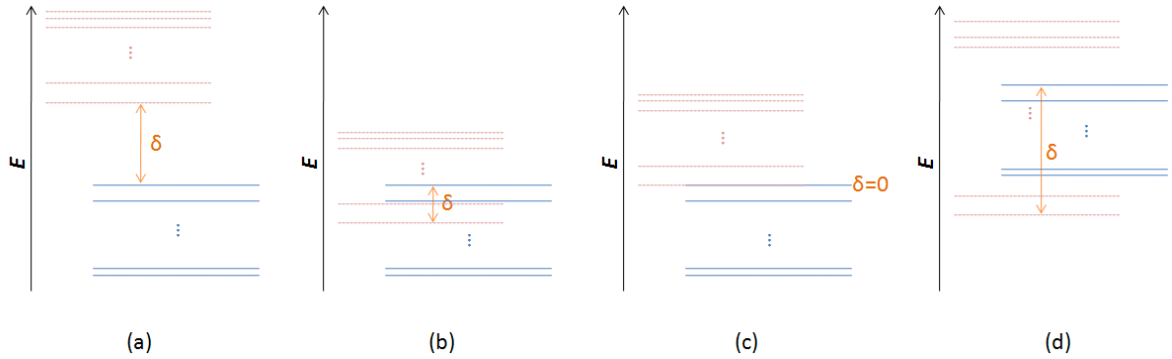

Figure 4-5: Types of energy distributions that can result from solving EMP at each stage of the dynamic process: (a) positive δ separating the desired from the competing configurations; (b) negative δ with the two sets of configurations overlapping; (c) zero δ; (d) negative δ with the desired set of configurations subsumed in the energy range of the competing set of configurations.

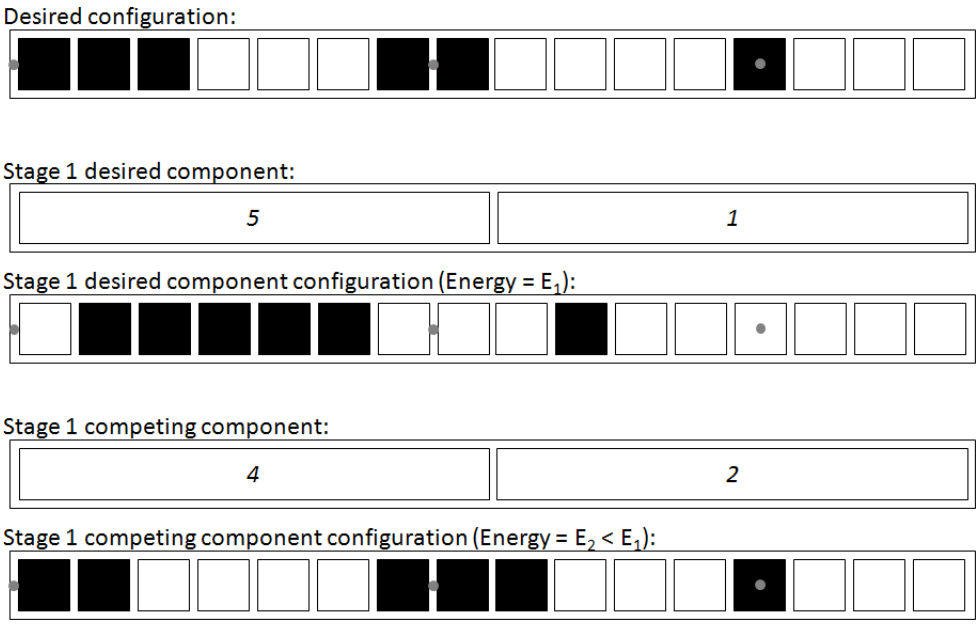

Figure 4-6: Two configurations belonging to 2 different components that portray the difficulty in guaranteeing that $\delta \geq 0$ in the multiresolution EMP formulation. Using the well-forming point conditions, the level 1 competing configuration shown will have a lower energy than the level 1 desired configuration shown.

4.3.1 Genetic algorithm approach

Genetic algorithms[54, 55] are a family of computational methods inspired by the evolution process in biology. Given a population of solutions to a specific problem, one calculates each member's fitness with a specific fitness function. Then, based on these fitness values, "parents" are chosen for the next generation. Recombination and mutation are applied to the parent population in order to form the "children" of the next generation. This process is repeated until a solution of satisfactory fitness is found.

Specific to the optimization problems we are trying to solve, we perform a search at each stage of the dynamic process. If we change the point condition locations in each stage of the dynamic process (e.g., Figure 4-4), we can provide either a randomly generated set of initial possible point condition strengths. However, given that we have a general idea of where we would like energetic wells and barriers to form within the system volume, we may also provide an intelligent initial guess, that defines whether a point condition will be well- or barrier- forming, based on its location, and a general order of magnitude, based on the particle number specifications at each stage of the dynamic process. If we utilize the Static Problem point conditions in the final stage of the dynamic process, as we do in Stage 4 for the 1D example system, we may use OSEMP results for the point condition strengths as the initial guess.

If we are utilizing the same set of the point condition locations throughout the dynamic process, i.e., those generated in the minimum tiling algorithm of the Static Problem, we may also provide a random or intelligent initial guess for the point condition strengths. However, we may perform the parameter search sequentially between the stages in order to use a solution in one stage to initialize the next stage's initial solution population. More specifically, we may look at each stage as either a restriction of the previous stage (i.e., a constriction of the subsets of configurations that represent the accessible and desired states) or a relaxation of the next stage (i.e., a controlled growth of the subsets of configuration that represent the accessible and desired states). The solution at one stage is therefore a good initial guess for either

the next or previous stage. Again, the Static Problem OSEMP solution can be used as an initial guess for the search in the final stage of the dynamic process. After a good solution is found for the final stage, we may use the solution as an initial guess to the previous stage's search. We can continue to do this until we reach Stage 1, the first stage we impose a restriction on the accessible and desired states of phase space. Figure 4-7 depicts this approach.

The genetic algorithm proceeds as follows:

Step 0: Solve the minimum tiling algorithm and EMP for the Static Problem degrees of freedom, $\mathbf{s}^{(SP)}$.

Step 1: If you are using the same set of point condition locations in Stage N of the dynamic process, initialize your first generation with $\mathbf{s}^{(SP)}$ being the first member, i.e., $\mathbf{s}^{(N+1)} = \mathbf{s}^{(SP)}$. If not, provide a random or intelligent guess, $\mathbf{s}^{(N+1)}$. Create a generation of m_G total members by mutating each point condition strength in $\mathbf{s}^{(N+1)}$, i.e., $s_k^{(N+1)} = s_k^{(N+1)} + \Delta$, where Δ is a random number normally distributed with $E[\Delta] = 0, \sigma_\Delta = d$. Set $i = N$.

Step 2: With each possible solution, i.e., member of the generation, simulate a Stage i system using dynamic MC. Initialize the simulation with a configuration within the Stage $i-1$ desired component. Operating at a particular k_BT value, repeat the MC simulation n_{sim} times. Output the average probability and standard deviation of achieving the desired configuration(s) in Stage i.

Step 3: Repeat Step 2 for all m_G possible solutions.

Step 4: Calculate the fitness, f_j, of the each possible solution, where $j = 1, 2, \cdots, m_G$. Fitness is calculated by the following equation:

$$f_j = E[p_j(\Omega_\alpha^{(i)})] + E[p_j(\Omega_\alpha^{(i)})] \cdot (1 - \sigma_{p_j(\Omega_\alpha^{(i)})}) = E[p_j(\Omega_\alpha^{(i)})] \cdot (2 - \sigma_{p_j(\Omega_\alpha^{(i)})}) \quad (4.14)$$

This fitness equation tells us that the fitness is not only a function of the probability of the desired state but also its variability; the higher the probability and lower the variability, the higher the fitness of the potential solution.

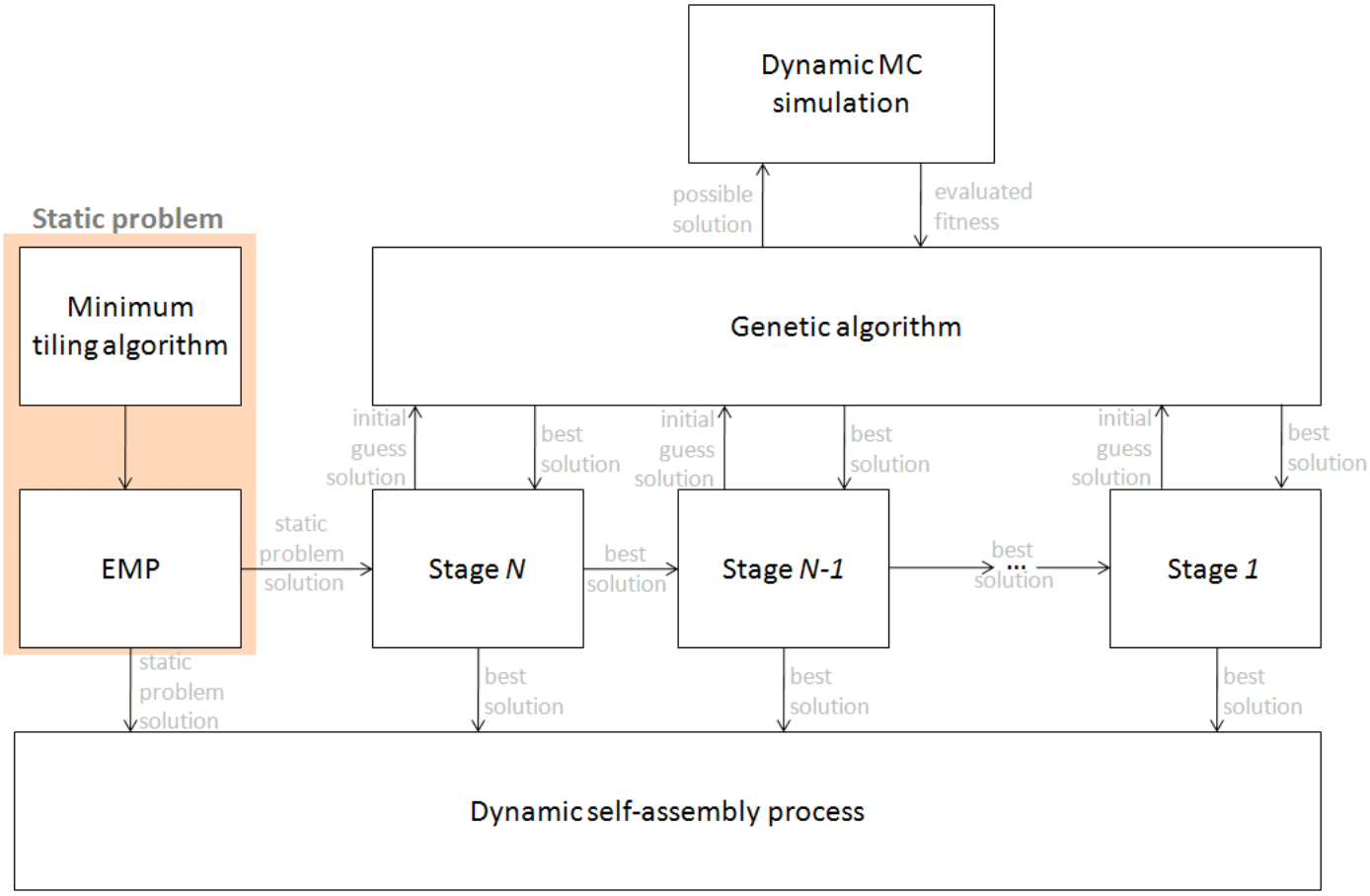

Figure 4-7: The flowchart of the dynamic self-assembly process approach when the same point condition locations are used in each process stage.

Step 5: Normalize the fitness by dividing each fitness by the sum of all fitness values within the generation, i.e.,

$$F_j = \frac{f_j}{\sum_k f_k}. \tag{4.15}$$

Step 6: Given all fitness values, choose m_G parents for the next generation. The probability of choosing a particular possible solution as a parent should be proportional to its fitness value. One can imagine mapping all the fitness values on a line between 0 and 1, where the length of the j^{th} segment is equal to F_j. Then, one can perform a stochastic sampling using m_G random numbers generated from a uniform distribution between 0 and 1.

Step 7: Perform a recombination on the new parent set. Select pairs of parents, and with probability p_c, recombine these solutions to form two new possible solutions in the next generation. If recombination is accepted, a random number generator can be used to choose the location to perform the crossover in the two possible solutions, i.e., choose a random integer value, r_c, between 1 and $N_{pc} - 1$. For example, if we perform a crossover with $r_c = 3$ on the strings, 11000 and 10101, the next generation strings are 11001 and 10100.

Step 8: Given the set of possible solutions in the next generation, mutate each point condition strength by a low probability, p_m. The mutation is performed similar to the initial mutation of the first generation in Step 1 above.

Step 9: Perform Steps 2-8, keeping track of the solution with the highest fitness value. Terminate the genetic algorithm for Stage i when this maximum fitness value is acceptable to the system designer. Though the fitness has an upper bound of 2, solutions with lower fitness values, representing local optima may be sufficient to guarantee a robust dynamic path at Stage i.

Step 10: If $i = 1$, we have found point condition strengths for each stage of the dynamic process. If $i > 1$, perform Steps 2-9 for $i = i - 1$.

4.4 The dynamic self-assembly process

Given the point condition locations and strengths defined in Sections 4.2 and 4.3, we must now define the dynamic process. At each stage of the process, we assume that the system is first able to sample all the states defined by the previous stage's set of desired configurations, i.e., the previous stage's desired configurations represent the accessible states at the current stage. However, at the end of each stage, we want to restrict the system's accessible configurations to only those that belong to the stage's set of desired configurations. This can only be done by imposing a transition in the behavior of the system toward nonergodicity. Because we guarantee that our desired set of configurations has a sufficiently high probability at each stage, then the likelihood of trapping the system within the component of desired configurations is high.

To cause this transition to nonergodic behavior, we must be able to adjust the relative magnitudes of energy differences between two adjacent configurations in the allowable region of phase space, ultimately having the ability to change what is considered a surmountable energy gap between configurations (defined by the system temperature). There are two ways to do this for the systems and degrees of freedom we are currently working with: (1) modify the heights/depths of peaks/wells by modifying the magnitudes of the point condition strengths, (2) change the system temperature. The latter is self-explanatory. The former can be done by slowly introducing the point conditions, i.e., providing strengths that are of the same relative magnitudes, but of lower overall strength. As time continues, these strengths can be increased to their final values.

At Stage 0, we have an ergodic system. To restrict the accessible region of phase space, we introduce the Stage 1 degrees of freedom with strengths and a system temperature that still allows the system to be ergodic. Then, we modify the appropriate parameters (i.e., the point condition strength magnitudes and/or system temperature) to trap the system in the appropriate component. In Stage 2, we want to start with the desired states of Stage 1 being accessible. Then, we slowly modify parame-

ters to restrict the system to the Stage 2 desired component. This is different from the transition from Stage 0 to Stage 1 because we no longer want to start with the system exhibiting ergodic behavior. Hence, a system in Stage 2 of the self-assembly process must start in the transition region between an ergodic and nonergodic system, therefore allowing the system to access states near the desired specification. Further sampling outside of this neighborhood may trap the system in an undesirable component. If we are working with the system temperature, we may start the system at a temperature within the glass transition region (see Section 3.3.1) and lower the temperature in time to restrict the accessible phase space region. We then use this same method for all the later stages until we have finished the final stage, where we have guided the system to the desired configuration. At this point, we utilize the Static Problem solution to guarantee a robust desired state.

Figure 4-8 shows the ergodic, nonergodic and transition regions for a probability distribution with respect to the system parameter being used to modify the system's behavior. For Stage 1, we want to start with an ergodic system and transition to a nonergodic system. For subsequent stages, we start with a system in the transition region and modify the system parameter(s) to create a nonergodic system. Figure 4-9 depicts the overall dynamic self-assembly process.

4.5 Dynamic problem examples

We performed this dynamic problem approach on the 1D and 2D example desired configurations depicted in Figures 3-6 and 3-14(a), respectively. In the former system, $N = 6, V = 16$, and in the latter system, $N = 7, V = 16$.

4.5.1 1D example system

The dynamic process for the 1D example system has 4 stages, depicted in Figure 4-2. For this example, we modify the point condition locations at each stage, as shown in Figure 4-4. Table 4.1 shows the initial guesses and the point condition strength values (with their fitness values) generated from the genetic algorithm. Within the genetic

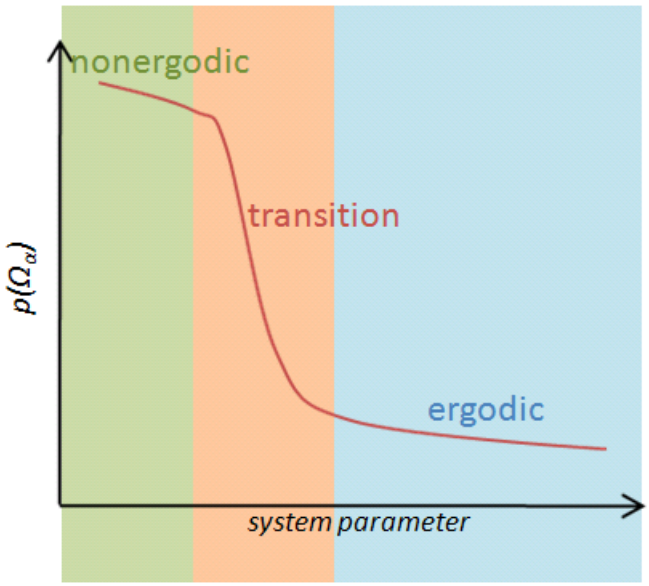

Figure 4-8: A pictorial representation of the different regions of the probability of achieving the desired state at a particular stage. Given changes in a particular system parameter, the system can transition from an ergodic to a nonergodic system, and vice versa.

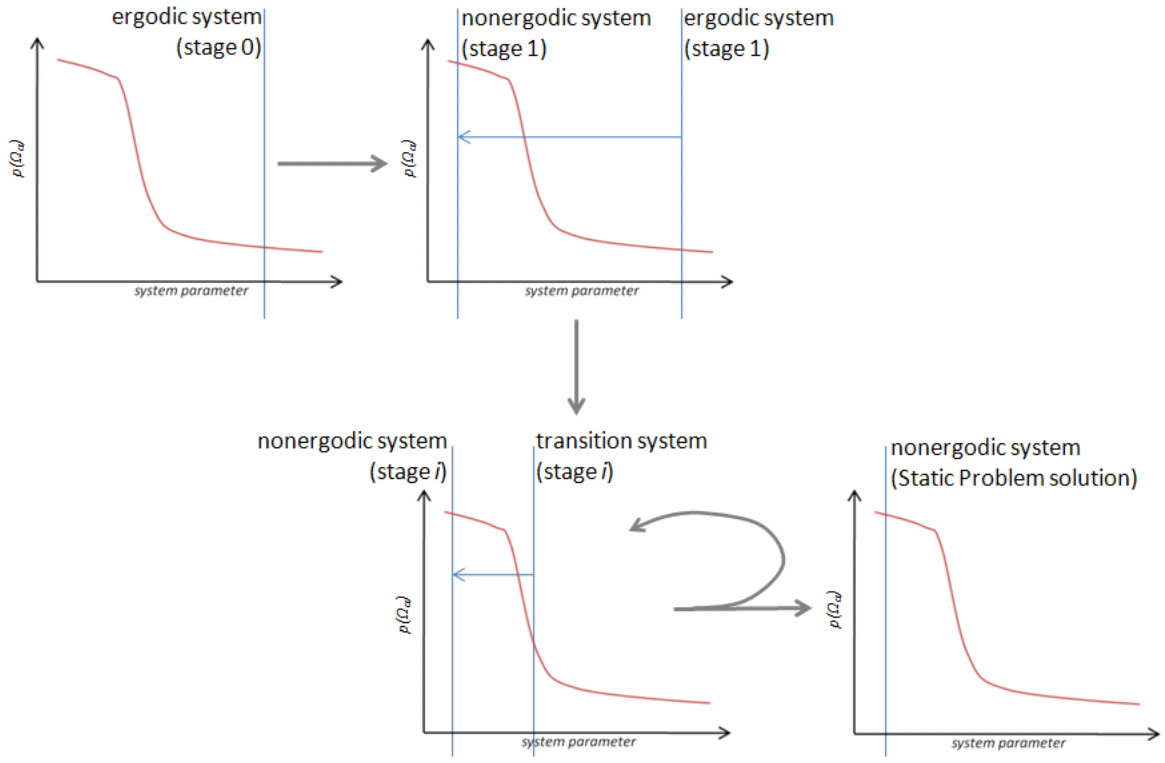

Figure 4-9: A view of the dynamic process at each stage, defined by the multiresolution view of the system.

Table 4.1: Initial guesses and generated point condition strengths for the 1D example system.

i (stage)	$s_{i,1}$	$s_{i,2}$	$s_{i,3}$	$s_{i,4}$	$s_{i,5}$	$s_{i,6}$	$s_{i,7}$	f
4, initial guess	100	27.83	33.47					
4, final	367.52	126.96	134.53					1.97
3, initial guess	100	50	-50	100	-100	50	-50	
3, final	106.82	-22.68	-103.34	97.04	-95.22	91.62	-64.72	2.00
2, initial guess	50	-50	50	-50	50			
2, final	100.53	-86.29	79.42	-146.10	20.44			1.84
1, initial guess	100	-50	50					
1, final	223.39	72.44	42.15					1.28

algorithm, we use the following parameter values: $m_G = 20$, $\sigma_\Delta = 50$, $k_BT = 10$, $n_{sim} = 10$, $p_c = 0.4$, $p_m = 0.02$.

Using these point condition locations and strengths at each stage, we simulated the dynamic self-assembly process through dynamic MC, using temperature as a means of controlling the system ergodicity. To find the appropriate simulation temperatures, we performed dynamic MC simulations of all the stages, initializing each simulation with a configuration in the desired component. Figure 4-10 shows the results of these simulations. The purpose of these simulations is to show us at what temperatures the system is ergodic, nonergodic and transitioning.

For Stage 1, we need temperatures which allow the system to exhibit ergodic and nonergodic behavior. Using the dynamic MC results, we transition from $k_BT = 6$ (ergodic) to $k_BT = 0.25$ (nonergodic), using increments of $\Delta(k_BT) = 0.25$, in the Stage 1 dynamic process. For all subsequent stages, we need two temperature values, one which allows the system to be nonergodic and one which allows the system to be in the transition region between ergodicity and nonergodicity. For Stage 2, we transition from $k_BT = 12$ (transition) to $k_BT = 6$ (nonergodic), using increments of $\Delta(k_BT) = 0.5$. For Stage 3, we transition from $k_BT = 6.5$ (transition) to $k_BT = 3$ (nonergodic), using increments of $\Delta(k_BT) = 0.5$. For Stage 4, we transition from $k_BT = 8$ (transition) to $k_BT = 3$ (nonergodic), using increments of $\Delta(k_BT) = 0.5$.

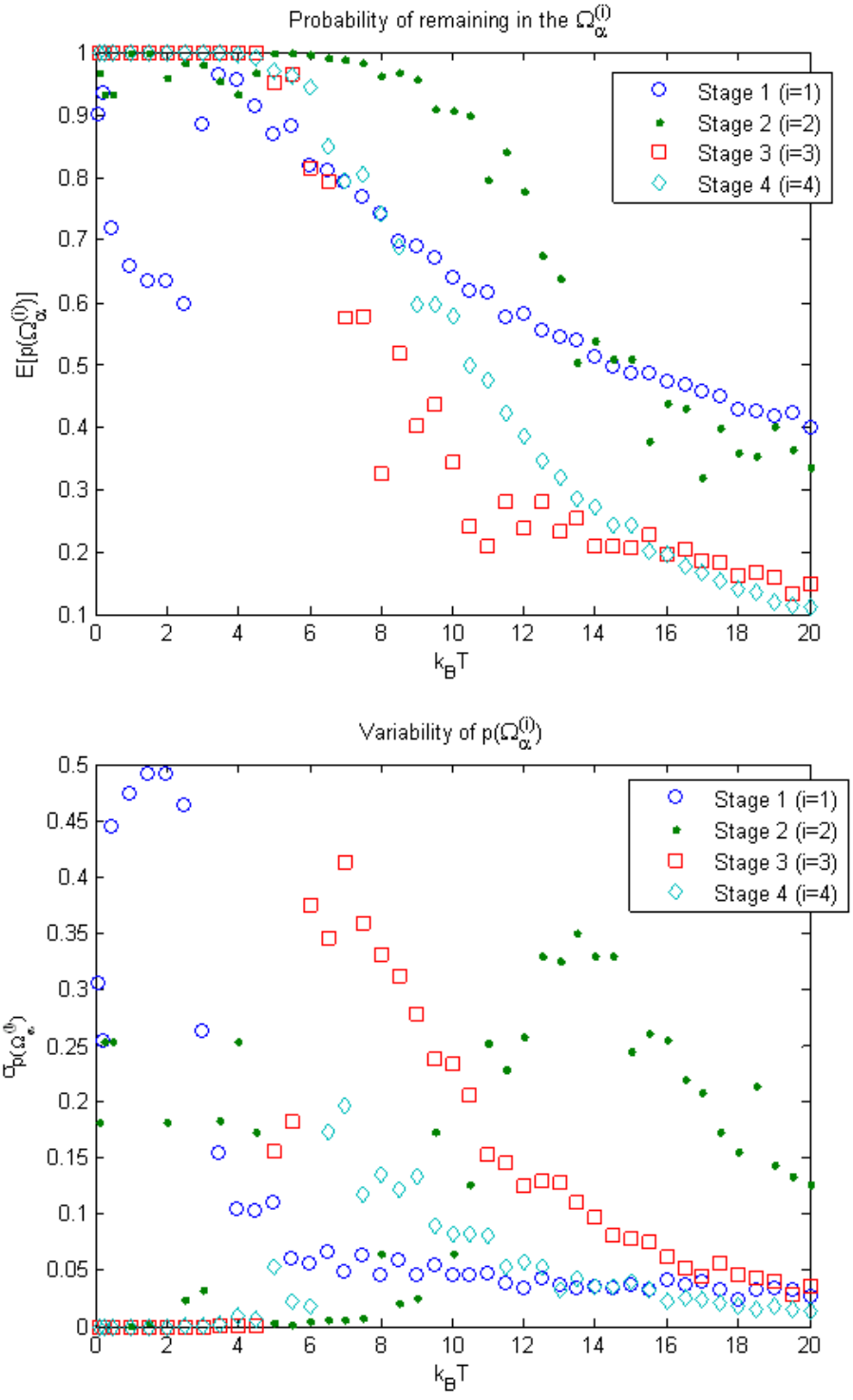

Figure 4-10: Dynamic MC results for staying in the desired set of configurations for each stage of the dynamic process for the 1D example system ($N = 6, V = 16$).

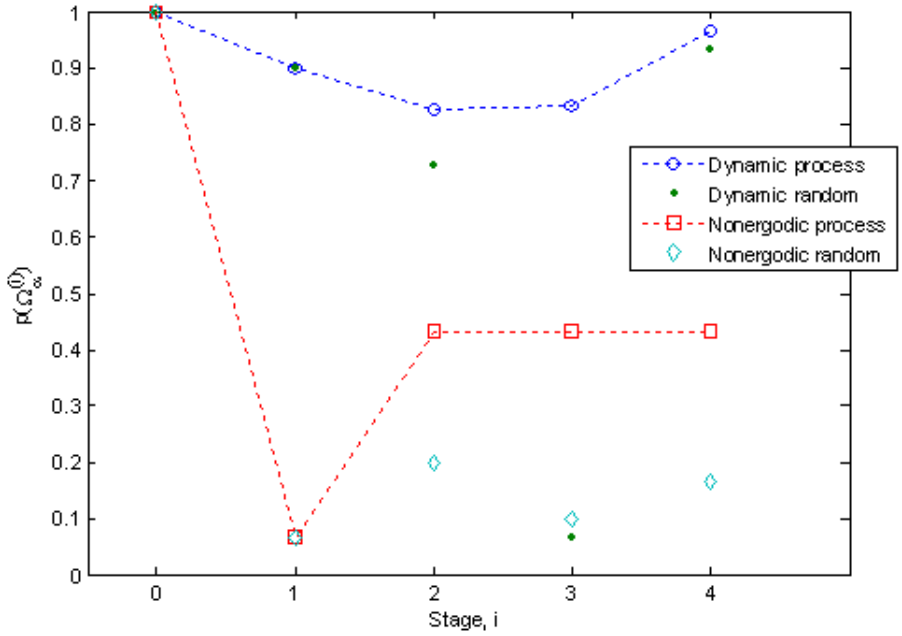

Figure 4-11: Results from dynamic MC simulations of the dynamic self-assembly process for the 1D example system.

Figure 4-11 shows the average of 30 simulation results of the entire self-assembly process from Stage 0 to Stage 4. We start at Stage 0 with a probability of unity for satisfying the desired configuration specification. After the Stage 1 transition from ergodic to nonergodic behavior, we see that the system has a 0.97 probability of becoming trapped in the set of desired configurations. We then move to Stage 2, where after we transition from a system in the transition region to a nonergodic system, we see that the system has a 0.87 probability of becoming trapped in the desired component. Hence, there is a higher probability of sampling outside the desired component in Stage 2. After Stage 3, the system has a 0.83 probability of achieving the desired component, and after Stage 4, the system has a 0.97 probability of arriving at the desired configuration. This tells us that the Stage 4 point conditions allow us to recover a majority of the systems that became trapped in a competing component in earlier stages of the dynamic process.

Figure 4-11 also includes three other data sets. The points labeled "nonergodic process" represent the dynamic process results if we use only the final temperatures

of each stage in the dynamic process, i.e., the simulation was performed at constant nonergodic system temperatures within each stage. We see that the probability of achieving the desired component in each stage is consistently lower at the nonergodic temperatures. Hence, the manipulation of system ergodicity throughout the dynamic process is important to ensuring that we find the desired configuration with high probability at the end of the process. The points labeled "nonergodic random" in Figure 4-11 represent simulations at nonergodic temperatures and a random initial configuration at each stage. These probabilities are consistently lower than both the dynamic and nonergodic process probabilities, which means that the use of stages that constrict the accessible region of the system phase space is advantageous to our dynamic process. Finally, the points labeled "dynamic random" represent simulations that use the dynamic process at each stage but initialize with a random configuration. These probabilities are consistently lower than the dynamic process probabilities. However, one should note that the Stage 4 system has a high probability of arriving at the desired configuration from a random initial configuration by just modifying the system ergodicity, which tells us that the desired configuration is not difficult to find using a simple annealing process. This will not be the case in the 2D example system. Overall, these results show us that the dynamic self-assembly process defined through a multiresolution view of the system provides a robust path from any initial configuration to the desired configuration.

4.5.2 2D example system

For the 2D example system ($N = 7, V = 16$), we define 3 Stages in the dynamic process, see Figure 4-12. We demonstrate both the dynamic process with variable point condition locations and also the point condition locations generated from the minimum tiling algorithm (Section 3.1), see Figure 4-13. Table 4.2 shows the initial guesses and the point condition strength values (with their fitness values) generated from the genetic algorithm. We used the same genetic algorithm parameter values as in the 1D example. For Stage 2, we used the OSEMP optimum point condition strengths as an initial guess into a genetic algorithm that solves the problem in

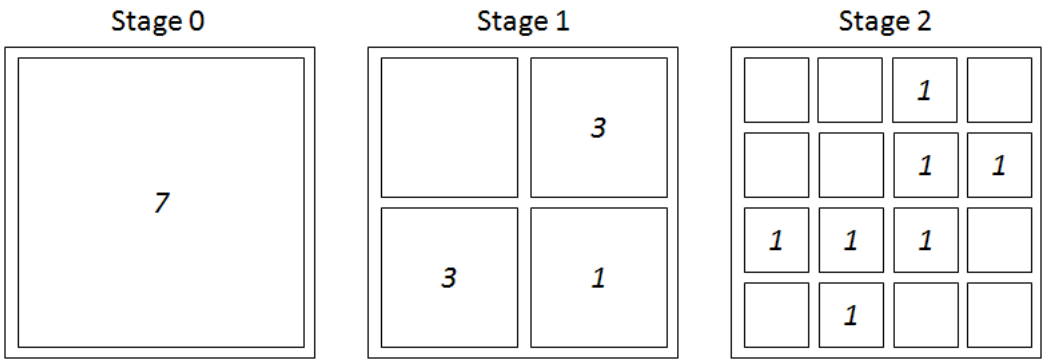

Figure 4-12: The desired particle number distributions in the three stages of the dynamic process for the 2D example system ($N = 7, V = 16$).

Table 4.2: Initial guesses and generated point condition strengths for the 2D example system.

i (stage)	$s_{i,1}$	$s_{i,2}$	$s_{i,3}$	$s_{i,4}$	$s_{i,5}$	f
2, initial guess	100	100	-3.38	100	99.20	
2, final	132.18	80.98	174.09	114.79	172.95	1.93
1, initial guess (static)	132.18	80.98	174.09	114.79	172.95	
1, final (static)	32.76	111.65	-111.62	-37.19	158.67	0.35
1, initial guess (variable)	-50	50	-50	50	50	
1, final (variable)	-114.28	134.58	-188.39	141.68	-70.55	1.47

Equation 4.4. For Stage 1 (static), we use the Stage 2 point condition strengths generated from the first genetic algorithm as an initial guess into another genetic algorithm that solves the problem in Equation 4.3. For Stage 1 (variable), we use an intelligent guess for the point condition strengths, based on the features we would like to generate in the system energy landscape.

Using the point condition strengths generated by the genetic algorithm at each stage, we simulated the dynamic self-assembly process. Within the simulation, we again used temperature as means of controlling the ergodicity of the system. To find the appropriate simulation temperatures, we performed dynamic MC simulations of both stages, initializing each simulation with a configuration in the desired component. Figure 4-14 shows the results of these simulations. For Stage 1 (static), we transition from $k_BT = 10$ (ergodic) to $k_BT = 1$ (nonergodic), using increments of

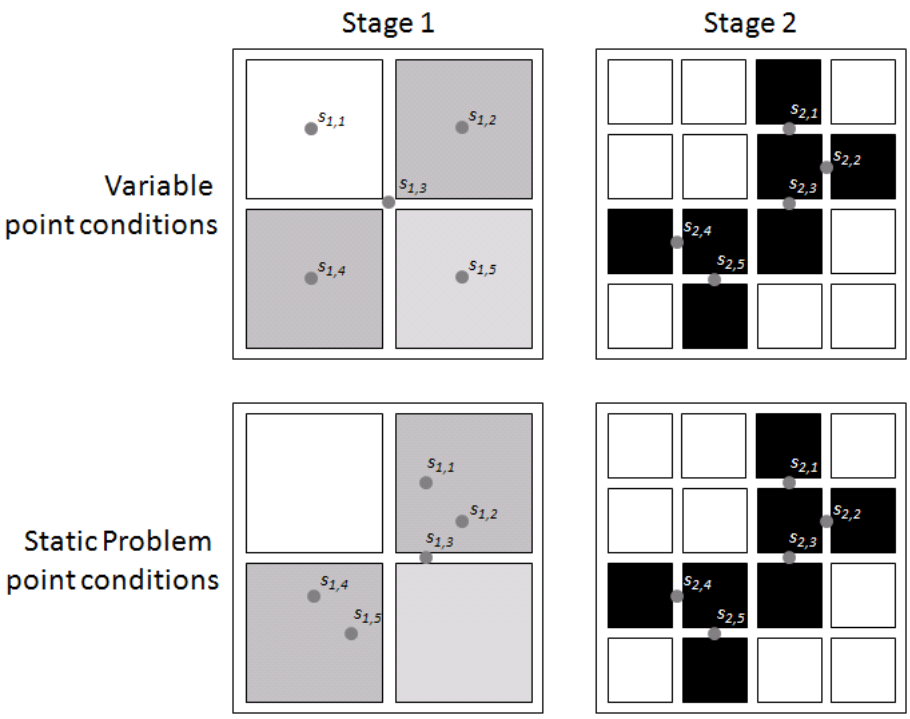

Figure 4-13: The point condition locations analyzed for the 2D example system.

$\Delta(k_BT) = 0.5$. For Stage 1 (variable), we transition from $k_BT = 18$ (ergodic) to $k_BT = 4$ (nonergodic), using increments of $\Delta(k_BT) = 1$. For Stage 2, we need to transition from a system in the transition temperature range to a nonergodic system. Using the dynamic MC simulation results, we transition from $k_BT = 5$ (transition) to $k_BT = 3$ (nonergodic), using increments of $\Delta(k_BT) = 0.25$.

Figure 4-15 shows the results of simulations of the entire self-assembly process from Stage 0 to Stage 2 using both variable and static point condition locations. For all stages, we see that the dynamic process results show equal or greater probabilities than their nonergodic counterparts. Also, the self-assembly process probabilities for both the dynamic and ergodic processes performed better than the randomly initialized results. The plots show that the variable point conditions process seems to be more robust. However, we note that the static point conditions had a low fitness value in the genetic algorithm. If we allowed the genetic algorithm to run longer and/or with a larger generation size, we may have arrived at a "fitter" solution, which in turn may have allowed for a better dynamic process with the static point conditions. In

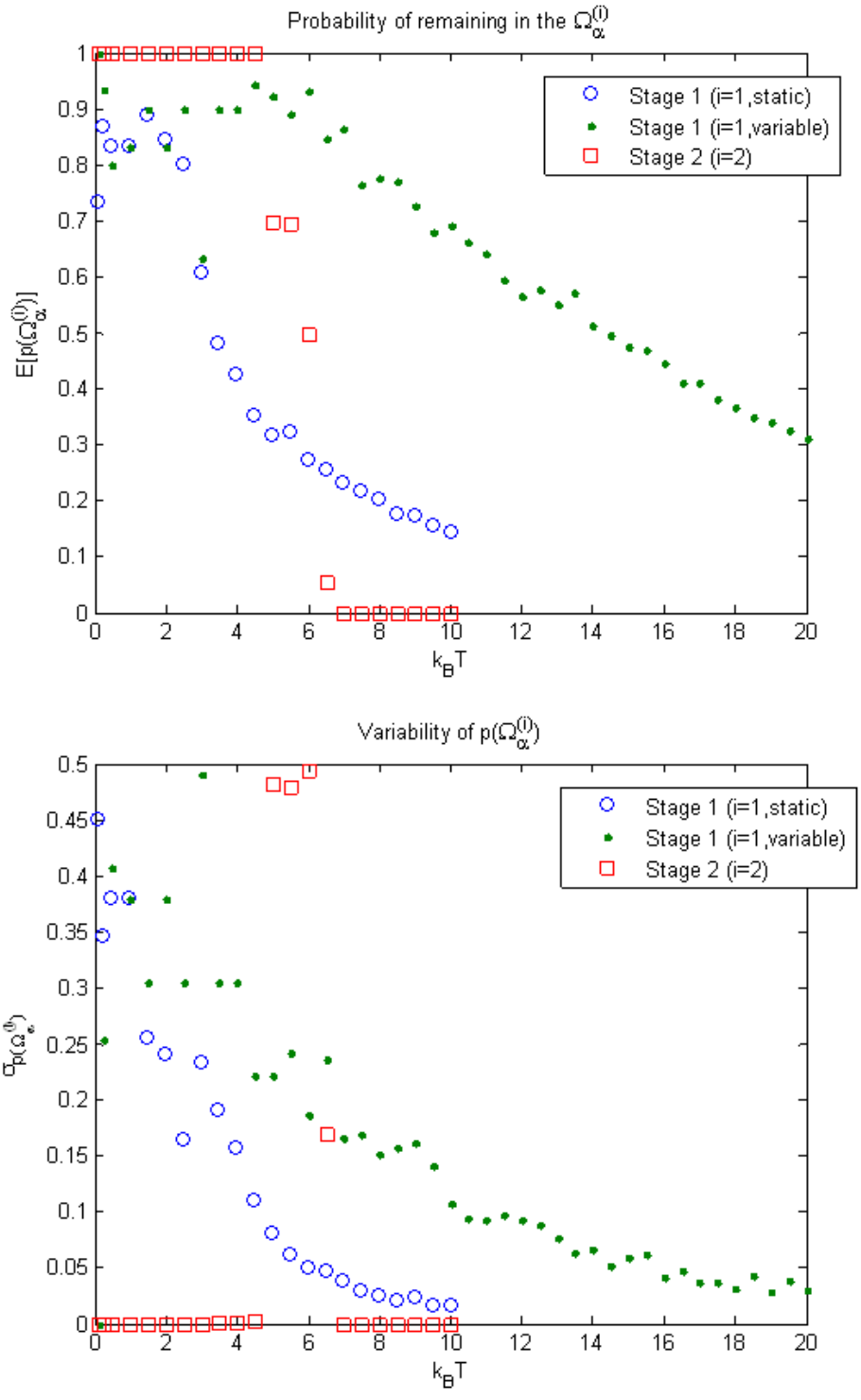

Figure 4-14: Dynamic MC results for staying in the desired set of configurations for each stage of the dynamic process for the 2D example system ($N = 7, V = 16$).

comparison to the 1D results, we see that the dynamic self-assembly process performs much better than a system that is annealed at the final stage of the process. Overall, the 2D example system results, similar to the 1D results, show us that the dynamic self-assembly process creates a robust dynamic path from any initial configuration to the desired configuration.

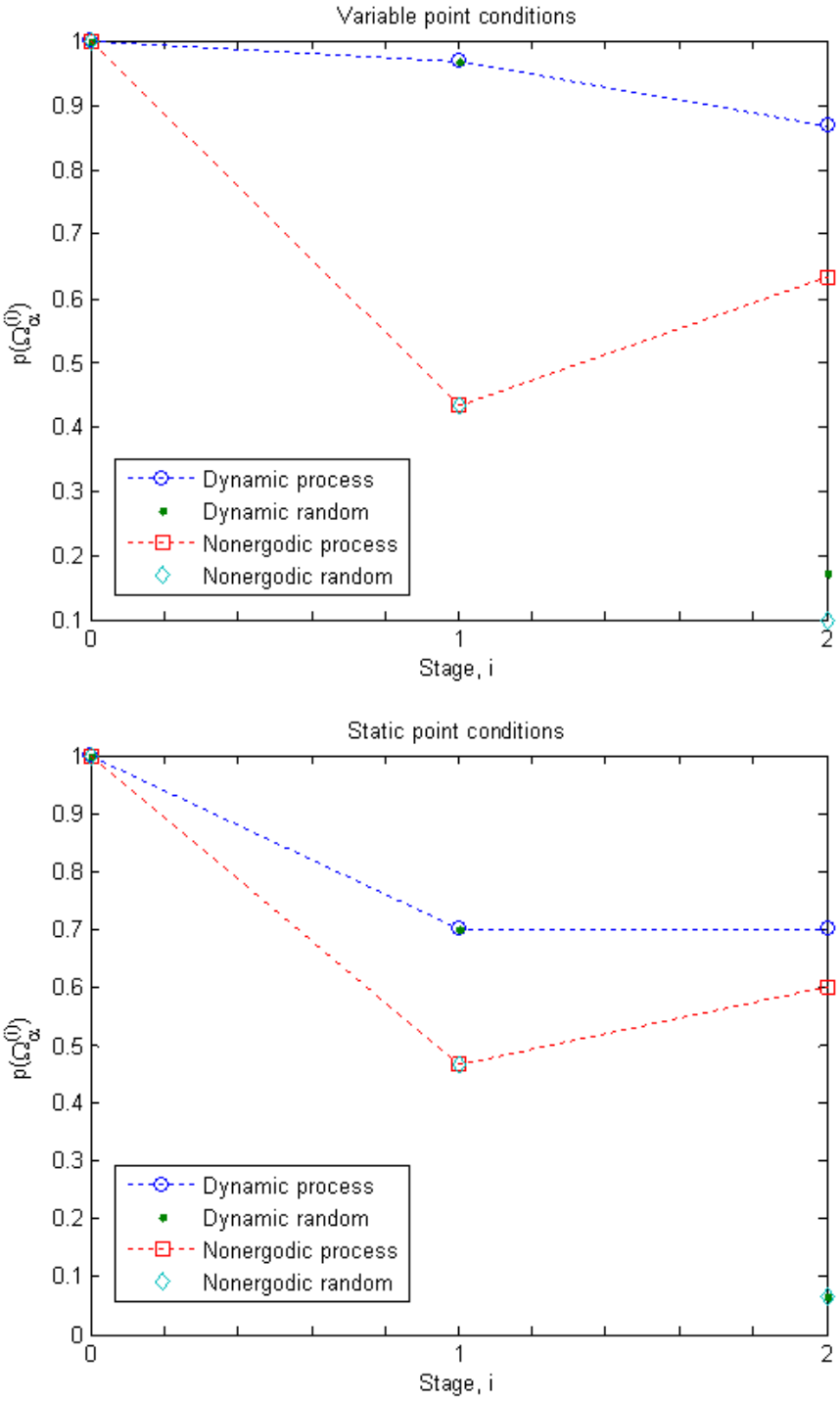

Figure 4-15: Results from dynamic MC simulations of the dynamic self-assembly process for the 2D example system.

Chapter 5

Conclusions and future directions

The construction of nanoscale structures with complex desired designs can be achieved through controlled self-assembly of nanoparticles. In this thesis, we have proposed a two-phase approach for the design and construction of such structures. In the first phase we solve the Static Problem that ensures the local stability and robustness of the desired structure through the judicious specification of external controls, while in the second phase we develop a time-varying specification of external controls in order to ensure that the final desired structure is reached from any initial particle distribution.

Some future investigations that can branch out from this body of work are outlined below:

Further investigation of the dynamic self-assembly process. In this thesis, we present positive results for a dynamic self-assembly process that uses a multiresolution view of the system particle number to constrict the accessible regions of phase space. We presented two approaches of defining the point condition locations but have not proposed a systematic approach to placing point conditions at the different stages of the self-assembly process. Within the self-assembly process, we discussed a transition to nonergodic behavior at each stage. However, further investigations may determine how process time, system temperature and level of system ergodicity work together to create a robust dynamic process.

Application to more complex models for real systems. The results described in this thesis used a simple phenomenological model for interparticle interactions and interactions with external controls. In moving towards the direction of applying the developed methods and tools, the use of more complex energy functions that have successfully been used to model real nanoparticle systems would bring this work closer to being useful to experimental researchers. Ultimately, the main concepts behind the problem formulations hold independent of the energy models used, so the applicability to a wide variety of nanoparticle systems is foreseeable.

Exploration of the degrees of freedom. The only system controls specified in all the examples were point conditions. However, there are many different types of system controls or degrees of freedom for nanoparticle systems. For example, there is a large body of work[15, 56] on how the geometric and surface properties of nanoparticles may affect their interactions amongst themselves and with their environment.

Experimental use of the methodologies and tools. Many researchers are working on experimental templated self-assembly systems[7]. The direct application of these design principles, for example, on charged nanoparticle self-assembly on a surface is apparent in the 2D example systems used throughout the thesis.

Developing a better understanding of designable systems. We discussed in this body of work the limitations current technology poses on the locations of the degrees of freedom. However, we did not pose such limitations on the point condition strengths. A study that discusses what is designable and what technologies need to be developed to allow for more complex designable structures would help provide a direction to alleviating the current limitations in achieving a desired self-assembled structure.

Appendix A

Proof of maximum-term method

In Section 1.1.1, we discussed how fluctuations disappeared in the thermodynamic limit, where we referenced the maximum-term method[19]. For completeness, we provide this proof.

In a constant N, V, T system, the energy is a fluctuating random variable. It can be shown that for a system in the thermodynamic limit ($N \to \infty, V \to \infty$, finite $\nu = N/V$), the partition function,

$$Z = \sum_{\{m\}} e^{-\beta E(m)} = \sum_{\{\epsilon\}} e^{-\beta F(\epsilon)} \approx e^{\beta F(\epsilon^*)}. \tag{A.1}$$

To prove this, consider the sum:

$$S = \sum_{i=1}^{M} \epsilon_i, \tag{A.2}$$

where each term is positive with an exponential dependence on N:

$$0 \leq \epsilon_i \sim O\left(e^{N\phi_i}\right). \tag{A.3}$$

Note that the pdf for a constant N, V, T system contains exponentials that are a function of energy, which is an extensive property, i.e., $E(\mu) \sim O(N^1)$. For each term

in the sum,

$$0 \leq \epsilon_i \leq \epsilon_{max}. \tag{A.4}$$

Therefore, the sum is bounded:

$$\epsilon_{max} \leq S \leq M \cdot \epsilon_{max}. \tag{A.5}$$

If we form the intensive property, $(\log S)/N$, it will be bounded as follows:

$$\frac{\log \epsilon_{max}}{N} \leq \frac{\log S}{N} \leq \frac{\log \epsilon_{max}}{N} + \frac{\log M}{N}. \tag{A.6}$$

Because M has a combinatorial relationship with N, i.e., $M \propto N^p$, the ratio $(\log M)/N$ vanishes in the thermodynamic limit:

$$\lim_{N \to \infty} \frac{\log S}{N} = \frac{\log \epsilon_{max}}{N} = \phi_{max}. \tag{A.7}$$

Therefore, in the thermodynamic limit of a system in the canonical prescription, the partition function can be simplified as shown in Equation A.1, guaranteeing that the most probable energy state has a probability that is essentially unity. This tells us that equilibrium energy fluctuations are insignificant for thermodynamic systems.

Appendix B

More Static Problem example systems

A smaller 1D example system ($N = 5, V = 8$) and a larger 2D example system ($N = 119, V = 1024$) were used for Conference Proceeding papers[57, 58].

The desired configuration and point condition locations, determined through the minimum tiling algorithm of Section 3.1, for the 1D example system are shown in Figure B-1. If we utilize the well-forming point conditions, s_1 and s_3, and solve PSEMP, we find that $\delta = 12.7$, $s_1 = 71.4$, and $s_3 = 100$.

The desired configuration and point condition locations for the larger 2D example system are shown in Figure B-2. For this example system, $N_{PC} = 4,225$. Before solving the set cover problem, we were able to reduce this number to 905 barrier- and 105 well- forming point conditions. After the set cover problem was solved, there remains 32 barrier- and 54 well- forming point conditions, shown in Figure B-2. We solved OSEMP with the 32 barrier-forming point conditions to find $\delta = -0.1$. To

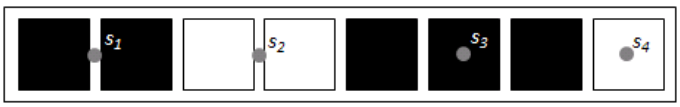

Figure B-1: The desired configuration and point condition locations for another 1D example system ($N = 5, V = 8$).

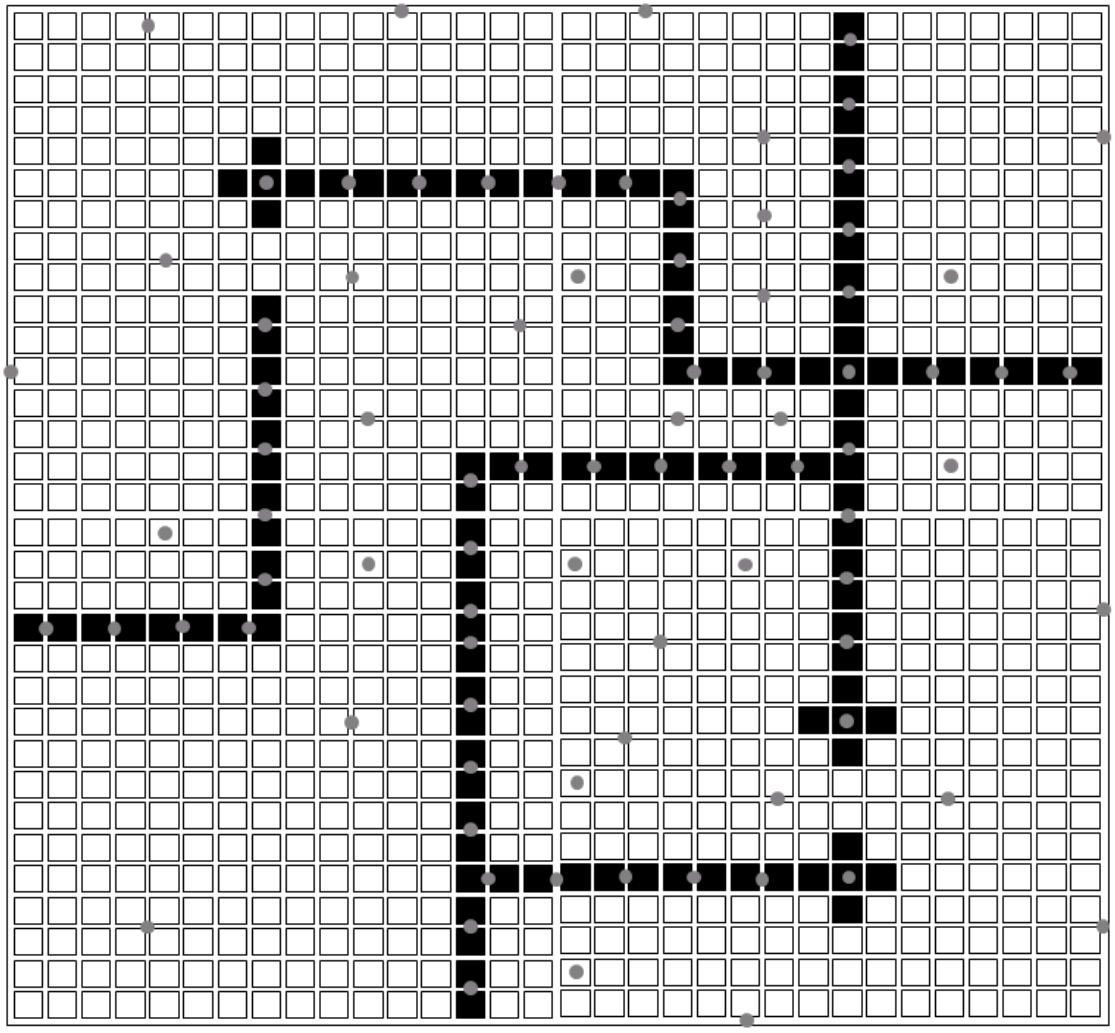

Figure B-2: The desired configuration and point condition locations for another 2D example system ($N = 119, V = 1024$).

make this positive, we analyzed the marginal values and added 23 of the well-forming point conditions, shown in Figure B-3. We solved OSEMP again to find $\delta = 1.2$.

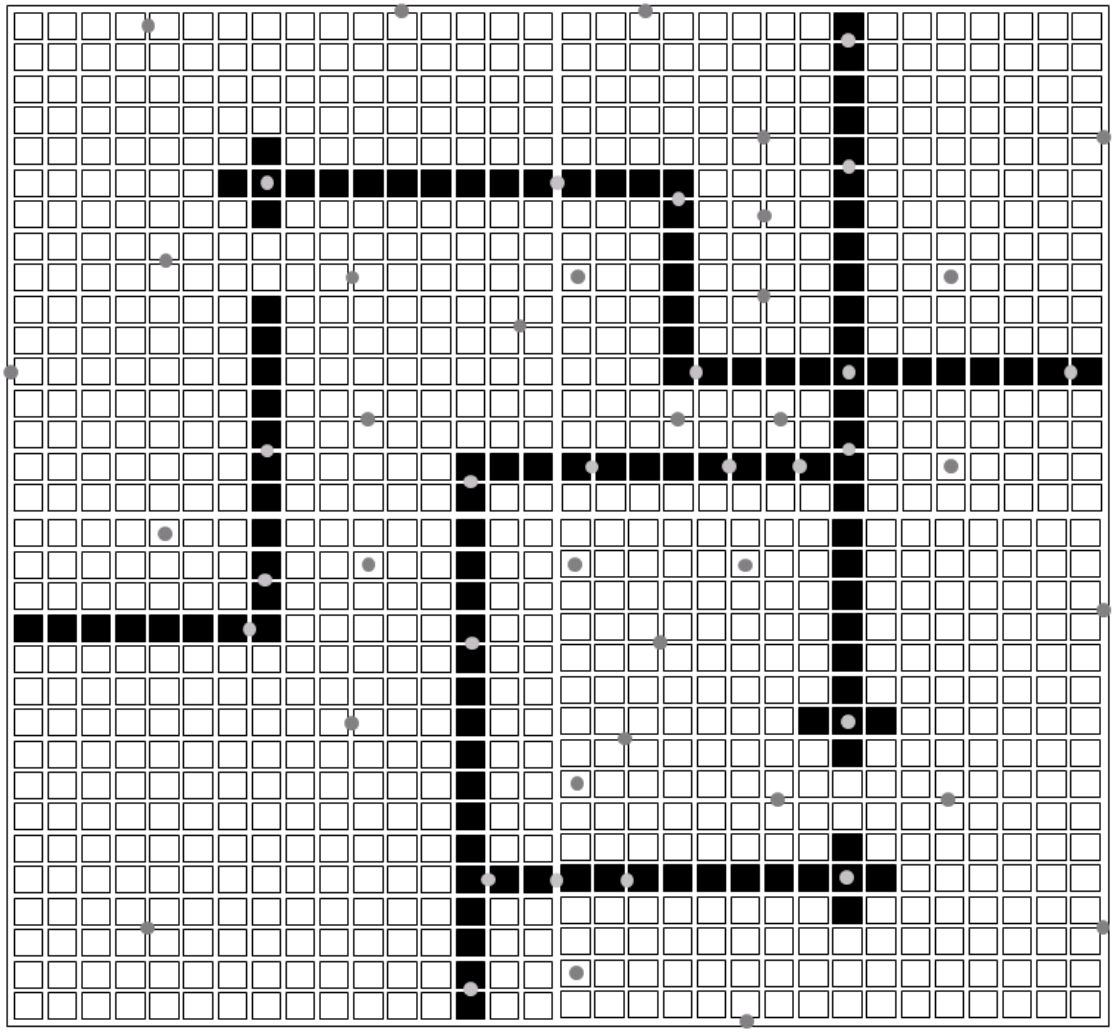

Figure B-3: The barrier-forming point conditions and 23 well-forming point conditions that provide an OSEMP objective function value, $\delta = 1.2$.

Bibliography

[1] Stephanopoulos N, Solis EOP, Stephanopoulos, G. *AIChE Journal*. 51 **2005**, p. 1858.

[2] Love JC, Estroff LA, Kriebel JK, Nuzzo RG, Whitesides G. *Chemical Reviews*. 105 **2005**, p. 1103.

[3] Bates FS, Fredrickson GH. *Thermoplastic Elastomers (2nd Ed)*. Holden G, Legge NR, Quirk R, Schroeder HE (Editors). **1996**.

[4] Fasolka MJ, Mayes AM. *Annual Reviews Materials Research*. 31 **2001**, p. 323.

[5] Aizenberg J, Black AJ, Whitesides GM. *Nature*. 398 **1999**, p. 495.

[6] Burns MM, Fournier JM, Golovchenko JA. *Science*. 249 **1990**, p. 749.

[7] Koh SJ. *Nanoscale Research Letters*. 2 **2007**, p. 519.

[8] Senturia SD. *Microsystem Design*, Kluwer Academic Publishers, **2000**.

[9] Plummer JD, Deal MD, Griffin PB. *Silicon VLSI Technology: Fundamentals, Practice and Modeling*, Prentice Hall, **2000**.

[10] Gates BD, Xu Q, Love JC, Wolfe DB, Whitesides GM. *Annual Reviews Materials Research*. 34 **2004**, p. 339.

[11] Stewart MD, Wilson CG. *MRS Bulletin*. 30, **2005**, p. 947.

[12] Hsu KH, Schultz PL, Ferreira PM, Fang NX. *Nano Letters*. 7 **2007**, p. 446.

[13] Krapf D, Wu M, Smeets RMM, Zandbergen HW, Dekker C, Lemay SG. *Nano Letters*. 6 **2006**, p. 105.

[14] Tuukkanen S, Toppari JJ, Kuzyk A, Hirviniemi L, Hytonen VP, Ihalainen T, Torma P. *Nano Letters*. 6 **2006**, p. 1339.

[15] Zhang ZL, Glotzer SC. *Nano Letters*. 4 **2004**, p. 1407.

[16] Rothemund PWK. *Nature*. 440 **2006**, p. 297.

[17] Park SH, Yin P, Liu Y, Reif JH, LaBean TH, Yan H. *Nano Letters*. 5 **2005**, p. 729.

[18] Zheng J, Constantinou PE, Micheel C, Alivisatos AP, Kiehl RA, Seeman NC. *Nano Letters*. 6 **2006**, p. 1502.

[19] McQuarrie DA. *Statistical Mechanics*, University Science Books, **2000**.

[20] Huang K. *Statistical Mechanics*, John Wiley & Sons, **1987**.

[21] Chandler D. *Introduction to Modern Statistical Mechanics*, Oxford University Press, **1987**.

[22] Ma SK. *Statistical Mechanics*, World Scientific Publishing Co., **1985**.

[23] Hill TL. *Thermodynamics of Small Systems, Part 1*, W.A. Benjamin, Inc., **1963**.

[24] Hill TL. *Thermodynamics of Small Systems, Part 2*, W.A. Benjamin, Inc.F, **1964**.

[25] Tsallis C. *Journal of Statistical Physics*. 52 **1988**, p. 479.

[26] Wang GM, Sevick EM, Mittag E, Searles DJ, Evans DJ. *Physical Review Letters*. 89 **2002**, p. 050601-1.

[27] Mohazzabi P, Mansoori GA. *Journal of Computational Theoretical Nanoscience*. 2 **2005**, p. 138.

[28] Hill TL. *An Introduction to Statistical Thermodynamics*, Dover Publications, **1986**.

[29] Palmer RG. *Advances in Physics*. 31 **1982**, p. 669.

[30] Fischer KH, Hertz JA. *Spin Glasses*, Cambridge University Press, **1991**.

[31] Binder K, Kob W. *Glassy Materials and Disordered Solids*, World Scientific, **2005**.

[32] Shakhnovich EI. *Chemical Reviews*. 106 **2006**, p. 1559.

[33] Shakhnovich EI, Gutin AM. *Nature*. 346 **1990**, p. 773.

[34] Landau DP, Binder K. *A Guide to Monte Carlo Simulations in Statistical Physics*, Cambridge University Press, **2000**.

[35] Frenkel D, Smit B. *Understanding Molecular Simulation: From Algorithms to Applications*, Academic Press, **2002**.

[36] Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AN, Teller E. *Journal Chemical Physics*. 21 **1953**, p. 1087.

[37] Whitelam S, Geissler PL. *Journal Chemical Physics*. 127 **2007**, p. 154101.

[38] Cormen TH, Leiserson CE, Rivest RL, Stein C. *Introduction to Algorithms*, MIT Press and McGraw-Hill, **2001**.

[39] Blankenship JW, Falk JE. *Journal Optimization Theory and Applications*. 19 **1976** p. 261.

[40] Adams WP, Sherali HD. *Management Science*. 32 **1986**, p. 1274.

[41] Mazurin OV. *Glass Physics and Chemistry*. 33 **2007**, p. 22.

[42] Lammers D. IBM Alliance Partners 'Open for Business' for 32 nm High-k/Metal Gate Designs. *Semiconductor International*. **14 April 2008**.

[43] LaPedus M. Nikon tops ASML for Intel's 32-nm node. *EE Times*. **07 July 2008**.

[44] Ross CB, Sun L, Crooks RM. *Langmuir*. 9 **1993**, p. 632.

[45] Barry CR, Gu J, Jacobs HO. *Nano Letters*. 5 **2005**, p. 2078.

[46] Cao TB, Xu QB, Winkleman A, Whitesides GM. *Small*. 1 **2005**, p. 1191.

[47] Chi PY, Lin HY, Liu CH, Chen CD. *Nanotechnology*. 17 **2006**, p. 4854.

[48] Fudouzi H, Kobayashi M, Shinya N. *Advanced Materials*. 14 **2002**, p. 1649.

[49] Tzeng, SD, Lin KJ, Hu JC, Chen LJ, Gwo S. *Advanced Materials*. 18 **2006**, p. 1147.

[50] Wu MY, Krapf D, Zandbergen M, Zandbergen H. *Applied Physics Letters*. 87 **2005**, p. 113106.

[51] Ma LC, Subramanian R, Huang HW, Ray V, Kim CU, Koh SJ. *Nano Letters*. 7 **2007**, p. 439.

[52] Levinthal C. *Journal Chimie Physique*. 65 **1968**, p. 44.

[53] Dill KA, Chan HS. *Nature Structural Biology*. 4 **1997**, p. 10.

[54] Whitley D. *Statistics and Computing*. 4 **1994**, p. 65.

[55] Michalewicz Z, Schoenauer M. *Evolutionary Computation*. 4 **1996**, p. 1.

[56] Jackson AM, Myerson JW, Stellacci F. *Nature Materials*. 3 **2004**, p. 330.

[57] Solis EOP, Barton PI, Stephanopoulos G. *FOCAPD Proceedings*, **2009**.

[58] Solis EOP, Barton PI, Stephanopoulos G. *PSE Proceedings*, **2009**.