State-space reduction and equivalence class sampling for a molecular self-assembly model

Daniel M. Packwood\textsuperscript{1,2,†}, Patrick Han\textsuperscript{1,3,4,5} and Taro Hitosugi\textsuperscript{1,6}

\textsuperscript{1}Advanced Institute for Materials Research (AIMR), Tohoku University, Sendai 980-8577, Japan
\textsuperscript{2}Japan Science and Technology Agency (PRESTO), Kawaguchi, Saitama 332-0012, Japan
\textsuperscript{3}California Nanosystems Institute, \textsuperscript{4}Department of Chemistry and Biochemistry, and \textsuperscript{5}Department of Materials Science and Engineering, University of California, Los Angeles, CA 90095, USA
\textsuperscript{6}Department of Applied Chemistry, Graduate School of Science and Engineering, Tokyo Institute of Technology, Tokyo 152-8552, Japan

Direct simulation of a model with a large state space will generate enormous volumes of data, much of which is not relevant to the questions under study. In this paper, we consider a molecular self-assembly model as a typical example of a large state-space model, and present a method for selectively retrieving ‘target information’ from this model. This method partitions the state space into equivalence classes, as identified by an appropriate equivalence relation. The set of equivalence classes \( H \), which serves as a reduced state space, contains none of the superfluous information of the original model. After construction and characterization of a Markov chain with state space \( H \), the target information is efficiently retrieved via Markov chain Monte Carlo sampling. This approach represents a new breed of simulation techniques which are highly optimized for studying molecular self-assembly and, moreover, serves as a valuable guideline for analysis of other large state-space models.

1. Introduction

The oft-cited quote ‘We are drowning in information but starved for knowledge’ implies that information has become so abundant that it is nearly impossible to draw useful conclusions from it.
The availability of affordable, high-performance computational resources allows for the simulation of sophisticated models with enormous state spaces. A researcher can, therefore, generate huge volumes of data with relatively little effort. On the other hand, the ‘target information’ desired by a researcher may be so deeply embedded in this dataset that it cannot be isolated without the use of elaborate statistical analysis. We will refer to these situations as large state-space problems. Techniques that efficiently recover target information from such large datasets are subject to intense research, and will remain of high priority so long as our hunger for facts and figures remains unsatisfied.

As a case in point, we consider the phenomenon of molecular self-assembly, in which molecules deposited on a surface spontaneously assemble into highly ordered structures [1–3]. Figure 1a illustrates the fabrication of graphene by self-assembly of molecules on a copper surface [4]. Upon heating, the molecules assemble into chain-shaped ‘islands’ (figure 1b,c), and on further heating a chemical reaction occurs between molecules, resulting in graphene (figure 1d). Island formation is a necessary ‘stepping stone’ for graphene formation, and one goal of this experiment is to identify conditions (temperature, type of molecule and type of surface) that support island formation. However, this experiment also yields considerable additional information, including the positions of the islands on the surface, the distances between the islands and so on. This information does not help us to identify island-forming conditions, although it may be interesting in other contexts. We are faced with the following problem: build a model for the molecular self-assembly process and then estimate the probability of forming particular islands (the ‘target information’) without estimating any other properties of the model. Such a model should possess a very large state space in order to incorporate every conceivable outcome of the experiment in figure 1. Molecular self-assembly, therefore, provides an example of a large state-space problem and is of sufficient scientific importance to warrant attention.

Numerous models and methods have been employed in the chemistry and physics literature to study the molecular self-assembly process. Some recent reports have described molecular dynamics simulations [5–10], density functional theory [11], kinetic Monte Carlo simulations [12], stochastic models [13], graph theory [14] and mean-field models [15], to name a few. However, the possibility of studying self-assembly by selective retrieval of target information as described above has not been considered in this literature. In fact, the need for such a study seems to be particularly acute for the case of molecular self-assembly. For example, while molecular dynamics simulations should eventually yield reliable statistics on island formation, the simulation times required for the entire state space to be sufficiently sampled are inaccessible with modern computational power. The enormous state space of typical molecular self-assembly models, therefore, provides a barrier to both data collection and data analysis. Specialized mathematical techniques that can astutely access the target information embedded in a molecular self-assembly model would, therefore, provide a new generation tools for studying molecular self-assembly.
molecular self-assembly. Moreover, such techniques would provide useful guidelines for developing similar methods for other types of large state-space problems.

In this paper, we present the equivalence class sampling (ECS) technique for acquiring target information from a simple molecular self-assembly model. In the ECS technique, the state space of the model is partitioned into equivalence classes via an appropriate equivalence relation. Then, Markov chain Monte Carlo (MCMC) sampling [16] is performed on the reduced state space $H$, where $H$ is the collection of equivalence classes. In this paper, two states belong to the same equivalence class if and only if they differ only in the position or orientation of their islands on the surface (precise details are provided in the main text). Recall that the target information for molecular self-assembly relates to the ‘kinds of islands’ that form under a given set of conditions. In ECS, all model configurations that contain the ‘same kinds of islands’ are placed into the same equivalence class. The reduced state space $H$ therefore lacks all information unwanted by the researcher, such as the distance between islands or positions of islands in the surface. From a physical point of view, it can be shown that ECS directly samples the model properties with respect to a free energy, which is most desirable for any method which calculates equilibrium (stationary) physical properties. The price of these benefits is that ECS (as developed in this paper) requires a ‘low-coverage approximation’, in which the density of molecules on the surface is very small. However, such conditions are typical in experiments which study island formation via molecular self-assembly. Molecular self-assembly processes occurring under high-coverage conditions tend to produce ‘two-dimensional crystals’ rather than molecular islands [8,9], and are, therefore, not concerned with island formation. The ECS concept has been explored for characterization of graphical models of statistical dependence (Bayesian networks) [17–23]. However, the results obtained in these studies do not seem to directly apply to the molecular self-assembly problem, and new developments of the ECS technique are presented here. In particular, we develop and characterize the extension–reduction chain, a Markov chain with state space $H$, which in turn is used to generate proposals in the MCMC part (specifically, for a Metropolis–Hastings (MH) algorithm) of the ECS technique.

This paper is organized as follows. In §2, we introduce the self-assembly model used in this study, and in §3, we introduce the ECS approach for this problem. In §4, we present some simulation results, and in §5, we draw conclusions.

2. Chiral block assembly model

The chiral block assembly (CBA) model is a simple model for the molecular self-assembly process described in figure 1. Consider a finite square lattice $L_d$ of dimension $d \times d$, where $d$ is an integer. $N$ blocks reside on the lattice (figure 2a), where $N$ is a fixed and given integer. A block is a square subset of the lattice that contains $3 \times 3$ points. The centre-most point of the block is called the coordinate of the block. The remaining eight points belong to the perimeter of the block. The perimeters of each block are decomposed as follows. Let $t$ be an arbitrary block and let $F$ denote the perimeter of $t$. Let $F_1$ (respectively, $F_2$, $F_3$, $F_4$) be the three points in $t$ that lie above (respectively, to the left of, below, to the right of) the coordinate of $t$. The sets $F_1$, $F_2$, $F_3$ and $F_4$ are called faces, and $F$ is the union of its faces. The block $t$ is said to have one of two types of orientations, $o_1$ and $o_2$, and two face types, $m_1$ and $m_2$. If $t$ has orientation $o_1$, then faces $F_1$ and $F_3$ are of type $m_1$, and faces $F_2$ and $F_4$ are of type $m_2$. If $t$ has orientation $o_2$, then faces $F_1$ and $F_3$ are of type $m_2$ and faces $F_2$ and $F_4$ are of type $m_1$. Two blocks are said to touch at their faces if the intersection of their faces contains more than one point. In this case, the two blocks are said to be neighbours. Note that this definition excludes cases where two blocks only share a ‘corner’. Two blocks are said to overlap if their coordinates are contained in their intersections. The blocks may be arranged on the lattice in any way and can touch at their faces, but no two blocks may overlap.

To define an interaction energy between neighbouring blocks (pairs of blocks that touch at their faces), we first classify the ways in which neighbours can be aligned with respect to each other. These alignments are referred as $P$, $S$ and $T$ and the rules for assigning the alignments are described in figure 2a. Note that the rules or assigning alignments are independent of the orientation of the block. For each pair of neighbouring blocks $i$ and $j$, we assign an interaction energy $\varepsilon_{r,m_1,m_2}$, where $r$ is the alignment of blocks $i$ and $j$ ($S$, $P$ or $T$), and $m_1$ ($m_2$) is the face of block $i$ (block $j$) which is touching the face of block $j$ (block $i$). The condition $\varepsilon_{r,m_1,m_2} = \varepsilon_{r,m_2,m_1}$ is enforced. The collection of these interaction energies is called the force field. The adjective ‘chiral’ refers to the fact that the alignment between any two neighbours does not possess mirror symmetry. This feature occurs in many real cases of molecular self-assembly on metal surfaces [4,24]. The CBA model is a special case of Winfree’s tile assembly model [25–27].
Without further mention, we will assume periodic boundary conditions on the CBA model. This is achieved by replacing the lattice $L_d$ with the lattice $L_{d+1} = X_{d+1} \times Y_{d+1}$, where $X_{d+1} = Y_{d+1} = [1, 2, \ldots, d, d+1]$, and then, imposing that the lattice points 1 and $d+1$ in $X_{d+1}$ are identical and that lattice points 1 and $d+1$ in $Y_{d+1}$ are identical. This results in the torus $T_d$. The nomenclature lattice and dimension will be taken to mean the set $X_d \times Y_d$ and the integer $d$, respectively.

A configuration is any choice of block positions and orientations that satisfies the conditions for the CBA model (no overlap of blocks). Let $\Omega$ be the collection of all configurations. For each $\sigma \in \Omega$, we assign the probability

$$\pi(\sigma) \propto e^{-\epsilon(\sigma)/k_B T_0},$$

where $k_B$ and $T_0$ are positive constants (the Boltzmann constant and temperature, respectively) and $\epsilon(\sigma)$ is the sum of the interaction energies for each pair of neighbouring blocks in $\sigma$. The normalizing constant in (2.1) is assumed unknown. A state is any choice of coordinates for the blocks that satisfy the conditions of the CBA model. We can write

$$\Omega = T \times O,$$

where $T$ is the state space and $O$ is all choices of orientations for the blocks in $T$. For each $b \in T$, we can assign the probability

$$\mu(b) \propto \sum_{o \in O} e^{-\epsilon(b, o)/k_B T_0},$$

where $\epsilon(b, o)$ is the sum of interaction energies for each pair of neighbouring blocks in the configuration $(b, o) \in \Omega$. Accordingly, we can write

$$\pi(b, o) = \mu(b) \lambda(o; b),$$

where $\lambda(o; b) \propto \exp(-\epsilon(b, o)/k_B T_0)$ is the probability of the blocks having orientations $o \in O$ given that the blocks are arranged according to $b$.

### 3. Equivalence class sampling

We wish to estimate the stationary state properties of the CBA model by sampling states from $T$ and orientations from $O$ according to the target distributions $\mu(\cdot)$ and $\lambda(\cdot; \cdot)$, respectively. We will focus exclusively on the problem of sampling from $T$, because the problem of sampling orientations (which is equivalent to sampling the Ising model) has been studied thoroughly in other contexts [28].

#### 3.1. State-space reduction and Markov chain Monte Carlo

The ECS technique involves two stages. In the first stage, the state space $T$ is reduced in such a way that some ‘target information’ is preserved and the remaining superfluous information is eliminated. For the self-assembly problem, the target information pertains to the kinds of islands that are likely to appear according to (2.3). We, therefore, begin by clarifying the wording ‘kinds of islands’ in terms of the CBA
model. Let $b$ be an arbitrary state of the CBA model and denote the blocks in $b$ by $t_1, t_2, \ldots, t_N$. An island in $b$ is a subset of blocks $I$ from $b$ such that

I.1. $I$ cannot be split into two disjoint subsets $I_1$ and $I_2$ in which there is no pair $t_a \in I_1$ and $t_b \in I_2$ which are neighbours, and

I.2. There does not exist a block $t_c \in I$ such that $t_c \in b$ and $I' = I \cup t_c$ satisfies I.1.

I.1. essentially says that an island cannot be split into two disjoint parts, and I.2. asserts that a subset of an island is not an island. Consider the two states $b_1$ and $b_2$ shown in figure 2b. Imprecisely, we can say that both $b_1$ and $b_2$ contain ‘same kinds’ of islands because if we rotate the islands in $b_1$ and translate them about the lattice, we can achieve $b_2$. More precisely, two states $b_1$ and $b_2$ are said to be rotational isomorphs if

RI.1. Each island in $b_1$ can be mapped onto an island in $b_2$ by rotations and translations alone, and

RI.2. this mapping is one-to-one and onto between the islands of $b_1$ and the islands of $b_2$.

Similarly, we say that two islands $I_1$ and $I_2$ are rotational isomorphs if $I_1$ can be mapped onto $I_2$ with rotations and translations alone. We can, therefore, partition $T$ into equivalence classes $E_1, E_2, \ldots, E_H$, where any two states $b_1$ and $b_2$ belong to the same equivalence class if and only if $b_1$ and $b_2$ are rotational isomorphs. We refer to the collection

$$ H = \{E_1, E_2, \ldots, E_H\}, \tag{3.1.1} $$

as the reduced state space. We also have the following.

**Theorem 3.1.** Let $b_1$ and $b_2$ be two states from the CBA model. If $b_1$ and $b_2$ belong to the same equivalence class, then $\mu(b_1) = \mu(b_2)$, where $\mu$ is the probability measure in equation (2.3).

Proof of theorem 3.1 is provided in appendix A.3. According to theorem 3.1, we can assign the probability

$$ \nu(E_k) \propto n(E_k)\mu(E_k) \tag{3.2} $$

to equivalence class $E_k \in H$, where $n(E_k)$ is the number of states that belong to equivalence class $E_k$ (the degeneracy of $E_k$), and $\mu(E_k)$ is the probability from equation (2.3) calculated for any member of $E_k$. The problem of ‘what kinds of islands appear at equilibrium’ can be stated as ‘which equivalence cases occur with relatively high probability $\nu’$. In order to obtain this target information, we only need to consider the reduced state space $H$, rather than the (considerably larger) space $T$. Note that, in contrast with the state space $T$, the reduced state space $H$ does not grow as the dimension $d$ of the lattice increases (providing that the number blocks $N$ is fixed). Thus, for very large $d$, the reduced state space $H$ will be considerably smaller than the full state space $T$.

In passing, we note an important physical aspect of the ECS approach. It can be shown that

$$ \nu(E_k) \propto e^{-A(E_k)/k_B T_0}, \tag{3.3} $$

where $A(E_k)$ can be regarded as the ‘free energy of equivalence class $E_k’$. This is discussed in detail in appendix A.1. ECS, therefore, directly incorporates thermodynamics into the MCMC sampling framework, which is a highly desirable feature when the equilibrium (stationary) state properties of a molecular self-assembly model are of interest.

In order to implement ECS in a computational environment, it is necessary to use a canonical representation for each equivalence class in $H$ (figure 2c). In this study, the canonical representation $C_k$ of equivalence class $E_k$ is generated by choosing an arbitrary state $b$ from $E_k$, and letting the elements of $C_k$ be the islands from $b$ as well as one empty element. The empty element is necessary for our implementation of MCMC as described in the following section. We do not distinguish between canonical representations that can be made equivalent by rotational isomorphism of their elements or re-ordering of their elements. Therefore, a canonical representation $C_k$ is a unique representation of the equivalence class $E_k$. The number of islands in equivalence class $E_k$ is the number $|C_k| - 1$.

### 3.2. Extension–reduction chain

To sample equivalence classes via the MCMC algorithm, we will employ the MH algorithm. We use $Z$ to denote the Markov chain generated by the MH algorithm (the ‘MH chain’), and $Y$ to denote the Markov chain used to generate proposals for the MH algorithm (the ‘proposal chain’). In this section,
we introduce a suitable proposal chain called extension–reduction chain. To define the extension–reduction chain, we first construct a transformation \( K \) that maps equivalence classes to other equivalence classes at random. Fix an equivalence class \( E_s \) with canonical representation \( C_s \) and consider the following actions.

T.1. Choose island \( k \) from \( C_s \) with probability

\[
P_k = \frac{i_k^{\alpha_1}}{\sum_{h=0}^{\left| C_s \right|} i_h^{\alpha_1}},
\]

where \( i_k \) is the number of blocks in island \( k \) and \( \alpha_1 \) is a positive, non-zero constant. Now, choose island \( j \) from \( C_s \) with probability

\[
Q_{jk} = \frac{1 + i_j^{\alpha_2}}{\sum_{h=0}^{\left| C_s \right|} (1 + i_h^{\alpha_2}) - (1 + i_k^{\alpha_2})},
\]

where \( \alpha_2 \) is a positive, non-zero constant.

T.2. For the island \( k \) chosen in step T.1, let \( I_k^- \) be the set of all islands that can be obtained by deleting a single block from island \( k \). Each of the islands in \( I_k^- \) must satisfy the conditions 1.1 and 1.2 outlined earlier. Now, if there are pairs of islands in \( I_k^- \) that are rotationally isomorphic, delete one member of the pair and repeat until there are no pairs of islands in \( I_k^- \) which are rotationally isomorphic. Denote the resulting set by \( T_k^- \). \( T_k^- \) is called the reduction set of island \( k \).

T.3. For the island \( j \) chosen in step T.1, let \( I_j^+ \) be the set of all islands that can be obtained by adding a single block to island \( j \). Then, delete islands from \( I_j^+ \) until there are no two pairs of islands in \( I_j^+ \) which are rotationally isomorphic. Denote the resulting set by \( T_j^+ \). \( T_j^+ \) is called the extension set of island \( j \).

T.4. Replace island \( k \) in \( C_s \) with a uniformly chosen element of \( T_k^- \), and replace island \( j \) in \( C_s \) with a uniformly chosen element of \( T_j^+ \). This produces a set \( C'_s \). Add one empty island to \( C'_s \) if there are no empty islands in \( C'_s \). If there are two empty islands in \( C'_s \), delete one empty island. Set \( C_r = C'_s \).

The resulting set \( C_r \) is a canonical representation for an equivalence class \( E_r \) which may or may not be identical to \( E_s \). This transformation is denoted with the symbol \( K \), and is referred to as the extension–reduction transformation. An example of an outcome of the extension–reduction transformation is shown in figure 3. Note that by virtue of equation (3.5), the empty island may be replaced with a single block with finite probability for any occurrence of the extension–reduction transformation. In such a case, we can intuitively think of the transformation as creating a new island by removing a block from an existing island.

The extension–reduction transformation \( K \) maps equivalence classes to equivalence classes at random. The probability

\[
d_{i_j} = P(K(C_i) = C_j)
\]

is evaluated in appendix A.5. With this characterization of the extension–reduction transformation, we can generate a Markov chain with state space \( H \) as follows.

**Theorem 3.2.** Choose an arbitrary equivalence class \( E_j \) from \( H \) and set \( Y_1 = C_j \), \( Y_2 = K(Y_1) \), \( Y_3 = K(Y_2) \), and so on. Then the sequence \( Y_1, Y_2, Y_3, \ldots \) is a \( \nu \)-irreducible and aperiodic Markov chain on \( H \) with transition probability \( P(Y_2 = E_j | Y_1 = E_k) = d_{i_j} \).
Theorem 3.2 is proved in appendix A.3. The Markov chain $Y$ in theorem 3.2 is the extension-reduction chain. The parameters $\alpha_1$ and $\alpha_2$ in equations (3.4) and (3.5) can be used to control the evolution of the $Y$. $Y$ does not have stationary distribution $\nu$, and to perform ECS we therefore use the transition probability in equation (3.6) to simulate the standard MH chain $Z$.

### 3.3. Approximation for the degeneracy factor

It does not appear possible to provide closed-form expressions for the degeneracy factor $n(E_k)$ in equation (3.2) in the general case. However, we can form a good approximation to it with the following two-step method. (a) Compute the degeneracy factor exactly for a special case in which each island only covers a single lattice point. (b) Show that the degeneracy factor for the actual CBA model converges to the degeneracy factor obtained from this special case in a 'low-coverage limit', in which the lattice is extremely large compared with the number of blocks.

The concept of a semicanonical representation is needed to carry out step (a) described above. Choose an equivalence class $E_k$ and let $b_1$ and $b_2$ be any two states that belong to $E_k$. Let $U_k$ and $V_k$ be the set of islands belonging to $b_1$ and $b_2$, respectively. Suppose that $U_k$ and $V_k$ can be made equivalent by translation of their islands without rotation. $U_k$ and $V_k$ are called semicanonical representations of equivalence class $E_k$. We do not distinguish between semicanonical representations that are equivalent under translation of islands, and therefore $U_k$ and $V_k$ are considered identical semicanonical representations. A canonical representation is also a semicanonical representation; however, two sets of islands corresponding to the same canonical representation may correspond to different semicanonical representations. Now, fix a semicanonical representation $V_k$. We can partition the set of islands in $V_k$ into subsets $V_k^0, V_k^1, \ldots, V_k^r$, where two islands belong to the same subset if and only if they can be made equivalent by translation about the lattice. We will refer to the quantity

$$n(V_k) = \frac{d^2!}{(d^2 - m)!a_1!a_2! \cdots a_r!},$$

as the degeneracy of the semicanonical representation $V_k$, where $m$ is the number of islands contained in $V_k$, and $a_1, \ldots, a_r$ are the number of islands belonging to the subsets $V_k^1, V_k^2, \ldots, V_k^r$ described above.

We now consider the meaning of $n(V_k)$ in equation (3.7). $n(V_k)$ is the number of ways to divide the $d^2$ points of the torus $T_d$ into $r+1$ subsets $V_k^0, V_k^1, V_k^2, \ldots, V_k^r$, where permutation of any points belonging to the same subset is does affect $n(V_k)$. The sets $V_k^1, V_k^2, \ldots, V_k^r$ are exactly those defined in the previous paragraph, whereas the set $V_k^0$ can be thought of as the set of $d^2 - m$ 'unoccupied' points of the lattice. Equation (3.7) therefore counts the number of ways to assign islands from a semicanonical representation to the lattice, but in such a way that the islands only cover a single point of the lattice while retaining information on their shape. The number

$$n_0(E_k) = \sum_{V \in \Gamma(E_k)} n(V),$$

where $\Gamma(E_k)$ is the set of all distinct semicanonical representations available to equivalence class $E_k$, can be interpreted as a degeneracy factor of the case where the islands only cover a single point of the lattice. Considering the actual degeneracy factor $n(E_k)$ in equation (3.2), we in general have that $n(E_k) < n_0(E_k)$. However, in appendix A.4, we show that for any equivalence class $E_k$,

$$\frac{n(E_k)}{n_0(E_k)} \to 1,$$

as $d \to \infty$ with $N$ (the number of blocks) fixed. Following step (b) mentioned above, we therefore employ the low-coverage approximation

$$n(E_k) \approx n_0(E_k),$$

in order to perform ECS in the following sections. The low-coverage approximation is mathematically permissible for the following reason. Let $Z$ and $Z^0$ be the MH Markov chains when sampling from the actual distribution $\nu$ in equation (3.2) and when using the low-coverage approximation, respectively. Without loss, we can take $Z$ and $Z^0$ to be $\nu$- and $\nu_0(E_k)$-distributed random variables, respectively. Equation (3.9) then implies that $\max_{E \in \mathcal{E}} |\nu_0(E) - \nu(E)| \to 0$ and hence that $Z$ converges to $Z^0$ in distribution as $d \to \infty$. From a physical point of view, the low-coverage approximation in equations (3.8) and (3.10) has connections with the statistical mechanical concept of indistinguishable and distinguishable particles. This suggests that under the low-coverage approximation, the CBA model
might be regarded as a kind of ‘two-dimensional lattice gas’ model. This very natural physical picture further supports the application of the low-coverage approximation in ECS. Note that study of MCMC Markov chains with approximate transition kernels appears to related to research on the ‘pseudo marginal’ MCMC algorithm [29–32].

3.4. Computational implementation

We have written a code called IslandPrediction which performs ECS on the CBA model as described above. To enhance convergence rates, the IslandPrediction code incorporates parallel tempering into the MH algorithm, exactly as described in reference [33]. The replica MH chains differ in the temperature of their stationary distribution, and each MH Markov chain has proposals generated according to its own extension–reduction chain, where these extension–reduction chains are simulated independently of one another. IslandPrediction is available as an R script [34,35] and as a program written in Go [36] (both language codes available upon request). The Go version of IslandPrediction is intended for multicore environments. A striking feature of IslandPrediction is that the MH Markov chain converges relatively quickly with the length of the extension–reduction chain. Using the conditions reported in table 1, we observe burn-in periods of around 20 000–30 000 steps of the MH algorithm and an asymptotic acceptance rate of between 0.1 and 0.5 (figure 4). Reasonable estimates of quantities such as island size distribution can be obtained after 100 000 steps of the MH algorithm, assuming that the burn-in period has been neglected. The simulations presented in the following section were actually run for 10⁶ steps, with the first 5 × 10⁵ steps removed before analysis. Results acquired with the shorter simulation criterion of only 100 000 steps described above showed no significant difference from the longer simulations.

Table 1. Parameters used in the simulations described in §4. \( T_0^1 \), \( T_0^2 \) and \( \tau_0^{step} \), respectively, refer to the lowest temperature, highest temperature and temperature step used for parallel tempering.

\[
\begin{array}{cccccc}
\epsilon_{P,11} & -0.1 \text{ eV} & \epsilon_{P,12} & 0.03 \text{ eV} \\
\epsilon_{S,11} & -0.08 \text{ eV} & \epsilon_{S,12} & 0.02 \text{ eV} \\
\epsilon_{T,11} & -0.05 \text{ eV} & \epsilon_{T,12} & -0.01 \text{ eV} \\
\epsilon_{P,22} & 0.05 \text{ eV} & d & 50 \\
\epsilon_{S,22} & 0.025 \text{ eV} & N & 10 \\
\epsilon_{T,22} & 0.10 \text{ eV} & \alpha_1, \alpha_2 & 1, 1 \\
T_0^1 & 100 \text{ K} & T_0^2 & 200 \text{ K} \\
\tau_0^{step} & 10 \text{ K} \\
\end{array}
\]

Figure 4. Plot of the quantity \( \mu \) in equation (3.2) as a function of \( Z_n \), the state of the MH chain at step \( n \), for a typical simulation of ECS at 100 K (see text for details). The red line corresponds to \( \alpha_n \), the acceptance rates up to step \( n \). Plot (a) shows the first 10⁵ steps of the simulation, and plot (b) shows the entire 10⁶ steps of the simulation.
Figure 5. (a) An equivalences class randomly chosen beyond the burn-in period for the simulation performed at 100 K (top row), and for the simulation performed at 150 K (bottom row). The insert explains how the graphics should be interpreted. (b) Island size distribution estimated between 100 and 200 K (see text). (c) Alignment distribution estimated between 100 and 200 K (see text for details). The sample of equivalence classes used to generate this figure is available as the electronic supplementary material.

presented in the following section. Computation of the transition probabilities $q_i^{j-1}$ in equations (A 25)–(A 28) is the major performance bottleneck in these calculations. These calculations can take between 0.5 to about 5 s depending on the number of and size of the islands in the canonical representations, and running the entire simulation for 100 000 steps of the MH algorithm typically takes around 16 h for the case of 10 blocks when the R code is used on a typical desktop computer. By contrast, simulations of 100 000 and $10^6$ steps of the MH algorithm only take around 30 min and 3 h, respectively, when the Go version of IslandPrediction is run on a 16 core cluster.

4. Simulation results

We now present results obtained using the ECS procedure with the conditions shown in table 1. The block orientations of any interesting islands were optimized individually using a standard technique used to study Ising models (see appendix A.1). Figure 5a shows randomly chosen equivalence classes that appeared beyond the burn-in regime in our calculations at 100 and 150 K. At low temperatures, we find that equivalence classes almost always contain a single island, whereas at higher temperatures, these equivalence classes instead consist of multiple, smaller islands. This observation is quantified in figure 5b, which shows the island size distributions computed from the above simulations. The island size distribution $f_k$ is the probability of an island containing $k$ blocks appearing in an equivalence class chosen randomly with respect to equation (3.2). At high temperatures, the island size distribution $f_k$ decays rapidly with the number of blocks $k$ in the island. A similar decay was reported for the graphene size distribution obtained in the experiment described in fig. 1 [4], which suggests that in these experiments, graphene was produced under conditions in which many small islands are present.

The behaviour of the model is most interesting at low temperatures. As temperature decreases, the distribution spreads towards larger island sizes. At about 120 K, a second sharp mode starts to appear at $k = 10$ blocks. The occurrence of this mode shows that, at low temperatures, once every block gathers into a single island, the blocks tend to remain in this state. In order to move away from this state, one of the blocks in this island must break away, resulting in a state containing two islands. One of
these islands contains nine blocks, and one contains one block. From a physical point of view, this second island contributes some entropy to the model; however, it also increases the overall energy of the model. This energy cost is particularly important at low temperature, and so this state either quickly reverts back to being a single large island, or another block departs from the island of size nine and joins the island containing one block. The appearance of two modes is therefore attributed to the fact that, once one block detaches from a large island, other blocks can be expected to detach in order to stabilize this block. As temperature decreases further, the blocks become increasingly stable when they are all part of a single island, and the probability density becomes concentrated on islands of size 10.

Figure 5c shows the alignment distribution estimated from the above simulations. This corresponds to the expected fraction of $P$, $S$- or $T$-type alignments observed in a randomly chosen equivalence class. At very low temperatures, almost all alignments are of the $P$-type. This quantifies the observation that large, chain-shaped islands tend to appear at low temperature, in agreement with the experimental observations described in figure 1. As temperature increases, the fraction of $S$-type alignments increases. This is a consequence of the entropic contribution to the model (described by the degeneracy term in equation (3.8)), which favours the appearance of islands of low symmetry.

5. Conclusion

Large state-space problems are rapidly becoming commonplace in most fields of science. For these problems, it is essential to have available methods that can efficiently extract ‘target information’ without burdening the researcher with volumes of superfluous information. This paper has presented a simple molecular self-assembly model that contains the basic elements of a large state-space problem, and developed the ECS technique for predicting a particular piece of target information (the ‘kinds of islands’ that appear with high probability in the model) with high precision. The key feature of ECS is that the model state space is reduced to a set of equivalence classes, where two states $b_1$ and $b_2$ belong to the same equivalence class if and only if $b_1$ can be transformed into $b_2$ under a suitably chosen map. By development of a special Markov chain on this reduced state space, we achieved efficient estimation of the target information in the model via MCMC sampling. While we have considered the special (but important) example of molecular self-assembly, we believe that the approach used in this study provides a ‘line-of-thinking’ also for other kinds of large state-space problems.

Two interesting mathematical problems are encountered in the development of ECS. In order to generate proposals for the MCMC sampling component of ECS (assuming that the MH algorithm is employed), one needs a Markov chain with state space $H$ and calculable transition probabilities, where $H$ is the collection of equivalence classes (reduced state space). Moreover, good estimates of the quantities $n(E_k)$, the number of states contained in equivalence class $E_k$, are needed to ensure that the MCMC algorithm samples from a distribution that is sufficiently close to the actual target distribution of the model. In this paper, we presented the extension–reduction chain as one example of such a Markov chain. However, there is motivation to develop other Markov chains and to study the convergence properties of the algorithms in detail. In particular, the types of Markov chains available for ECS are expected to depend upon the specific problem under consideration. We also presented an approximation for $n(E_k)$ in the case of low-coverage of molecules on the surface, and showed that this approximation converges as desired in the limit of zero surface coverage. While many experimental studies of molecular self-assembly use low-coverage conditions, it may also be interesting to develop other approximations to the degeneracies $n(E_k)$ that hold in the high-coverage limit. While not so relevant for the generation of small molecular islands, experiments in the high-coverage limit are widely used to generate so-called two-dimensional crystals via molecular self-assembly [24]. The high-coverage regime is also of academic interest as it often involves relatively unintuitive self-assembly behaviour [9]. Another open problem for implementing ECS algorithms concerns the development of efficient algorithms for detecting rotational isomorphism between islands and configurations of self-assembly models. This computation is a bottleneck in our current implementation of ECS. Perhaps, the greatest merit of studying molecular self-assembly via ECS is that the routes to improvement are very clear. We will explore them in detail in future research.

Data accessibility. The equivalence class sample used to generate figure 5 is available as an .Rdata workspace file for R. Upon loading this data into the R environment, one should type ‘Message’ to receive an explanation of the data structure. This data can be freely downloaded at http://dx.doi.org/10.5061/dryad.vd156.
Appendix A

A.1. Equivalence class sampling and free energy

Observe that equation (2.3) in the main text defines an ‘orientation partition function’ for the state \( b \). From statistical thermodynamics, we therefore obtain the identity

\[
A_0(b) = -k_B T \ln Z_0(b),
\]

where \( Z_0(b) = \mu(b) \), and \( A_0(b) \) can be regarded as a ‘partial’ free energy. The energetic component of \( A_0(b) \) arises from the interactions between neighbouring blocks in \( b \), and the entropic component arises from the multiple ways in which orientations can be assigned to the blocks in \( b \). Note that \( A_0(b) \) will have the same value for all \( b \) in equivalence class \( E_k \) (see appendix A.3), and so we can write \( A_0(E_k) = A_0(b) \) without any ambiguity. Now consider the ‘degeneracy factor’ \( n(E_k) \) in equation (3.2). We can define the entropy

\[
S(E_k) = k_B \log n(E_k),
\]

which arises from the multiple ways to place the blocks on the lattice such that the resulting state is in equivalence class \( E_k \). Solving equations (A 1) and (A 2) for \( n(E_k) \) and \( Z_0(b) \), respectively, and substituting the results into equation (3.2) of the main text gives

\[
u(E_k) \propto e^{-A_0(E_k)/k_BT},
\]

where the ‘full’ free energy \( A(E_k) \) is defined as

\[
A(E_k) = A_0(E_k) - TS(E_k).
\]

In this free energy, the energetic component arises from the interaction between neighbouring blocks for each state \( b \) in equivalence class \( E_k \), and the entropic component is a sum of two sources of entropy: an entropy arising from the multiple ways of choosing the orientations for the blocks, and an entropy arising from the multiple ways of placing the blocks on the lattice to achieve a state from equivalence class \( E_k \).

In the view of equation (A 3), ECS chooses equivalence classes with respect to a free energy. This is in contrast with sampling the configurations directly from (2.1), in which configurations are chosen with respect to their energy.

A.2. MCMC for sampling of block orientation distribution

For an equivalence class \( E \) with canonical representation \( C \), consider an island \( I \) from \( C \). Let \( O \) be the orientation space for the blocks in \( I \). For convenience, denote the possible two orientations of the blocks as \( \alpha \) and \( \beta \), respectively. A Markov chain \( Y \) on \( O \) can be generated as follows. We generate a sequence of random variables \( B_1, B_2, \ldots, B_m \) from a binomial distribution with \( n_I \) trials and a probability \( p \) of success per trial, where \( n_I \) is the number of blocks in island \( I \). Then, the \( m \)th step of the Markov chain, \( Y_m \), is generated by choosing \( B_m \) blocks at random and assigning them the orientation \( \beta \), and assigning the orientation \( \alpha \) to the other blocks. An MH algorithm using \( Y \) to generate proposals can then performed to analyse the distribution of the block orientations in the island \( I \).

For all cases in this study, the MH algorithm appeared to converge within tens of seconds for each island analysed. Markov chains were typically run for 3000 steps with \( p = 0.5 \). The block orientations from \( O \) that appeared most often in the MH output were then identified as the optimum choice of block orientations for the island.
A.3. Theorems and proofs

Proof of theorem 3.1. Let us temporarily denote the two possible orientations for each block as $\alpha$ and $\beta$, respectively. We first consider the case where the states $b_1$ and $b_2$ each contain one island $I_1$ and $I_2$, respectively. If $b_1$ and $b_2$ are in the same equivalence class, then $I_1$ and $I_2$ are rotational isomorphs, and according to condition RI.1 of the main text there is a bijection $v$ that maps the labels of the blocks in $I_1$ to the labels of the blocks in $I_2$. Let $m$ be the number of times that the island $I_1$ is rotated under the map $v$. Define the maps $r : O \to O$ and $f^m : O \to O$ such that for any $(o_1, o_2, \ldots, o_N) \in O$,

$$r(o_1, o_2, \ldots, o_N) = (o_{v(1)}, o_{v(2)}, \ldots, o_{v(N)})$$

and

$$f^m(o_1, o_2, \ldots, o_N) = (f^m(o_1), f^m(o_2), \ldots, f^m(o_N)),$$

where

$$f^m(o_k) = \begin{cases} \alpha & m \text{ is odd, } o_k = \beta, \\ \beta & m \text{ is odd, } o_k = \alpha, \\ o_k & m \text{ is even}. \end{cases}$$

Note that both $r : O \to O$ and $f^m : O \to O$ are bijective. For any $o = (o_1, o_2, \ldots, o_N) \in O$, we have

$$\epsilon(b_1, o) = \epsilon(b_2, r \circ f^m(o)),$$

and hence

$$\sum_{o \in O} (e^{-\epsilon(b_1, o)} - e^{-\epsilon(b_2, r \circ f^m(o))}) = 0,$$

where $\epsilon = (k_BT)^{-1}$. However, we also have that $r \circ f^m : O \to O$ is bijective. This gives

$$\sum_{o \in O} e^{-\epsilon(b_2, r \circ f^m(o))} = \sum_{o \in f^m(O)} e^{-\epsilon(b_2, o)} = \sum_{o \in O} e^{-\epsilon(b_2, o)},$$

where $r \circ f^m(O) = (r \circ f^m(o))_{o \in O}$. Substituting (A 10) into (A 9) and using the definition of $\mu$ in equation (2.3) of the main text gives $\mu(b_1) = \mu(b_2)$.

We now consider the case where the states $b_1$ and $b_2$ are in the same equivalence class and contain islands $(I_1, I_2, \ldots, I_n)$ and $(J_1, J_2, \ldots, J_n)$, respectively. As can be seen from the definition in equation (2.3), we have

$$\mu(b_1) = \mu(I_1) \mu(I_2) \cdots \mu(I_n)$$

and

$$\mu(b_2) = \mu(J_1) \mu(J_2) \cdots \mu(J_n).$$

According to condition RI.2 of the main text, there is a one-to-one correspondence between the islands in $(I_1, I_2, \ldots, I_n)$ and $(J_1, J_2, \ldots, J_n)$. Without loss, suppose that this correspondence is of the form $(I_1, J_1)$, $(I_2, J_2)$, $\ldots$, and $(I_n, J_n)$. Following the steps from the first part of the proof shows that $\mu(I_k) = \mu(J_k)$ for all $k$, which completes the proof.

Proof of theorem 3.2. $Y$ is a Markov chain, because the event that $Y_m = E_k$ depends upon $Y_{n-1}$ but not on $Y_{n-m}$, for $n \geq m > 1$. Moreover, by the definition of $q_k^{i,j,r,s}$ we have that $P(Y_2 = E_j | Y_1 = E_k) = q_k^{i,j,r,s}$. We proceed as follows to prove that $Y$ is $\nu$-irreducible. Let $E_u$ denote the equivalence class with a canonical representation

$$C_u = \{A_1, A_1, \ldots, A_1, \emptyset\},$$

where the island $A_1$, which is isomorphic to a single block, appears $N$ times in (A 13). Now consider any arbitrary equivalence class $E_k$ with a canonical representation

$$C_k = \{I_1, I_2, \ldots, I_m, \emptyset\},$$

where the islands in $C_k$ are ordered such that $|I_1| \geq |I_2| \geq \cdots \geq |I_m|$, where $|I_j|$ is the number of blocks in island $I_j$. Consider a transformation $E_k \to E_j$ where

$$C_k = \{I_1, I_2, \ldots, I_m, \emptyset\}$$

and

$$C_j = \{I_1, I_2, \ldots, I_m, A_1, \emptyset\}$$
are the canonical representations of $E_k$ and $E_j$, and $|I_1| = |I_1| - 1$. To describe this transformation, we say that island $I_1$ is reduced to $J_1$ and that the empty island $\emptyset$ is extended to $A_1$. This transformation occurs with probability,

$$q^{k \rightarrow j} \geq \frac{1}{|I_1|} P_{I_1}^k Q_{I_1,J_1}^k > 0,$$  \hspace{1cm} (A 17)

where $P_{I_1}^k$ is the probability that island $I_1$ of equivalence class $k$ is reduced during the extension–reduction transformation, and $Q_{I_1,J_1}^k$ is the probability that island $\emptyset$ is extended given that $I_1$ is reduced during the extension–reduction transformation. These quantities correspond to equations (3.4) and (3.5) of the main text, respectively. We also have that the reverse transformation $E_j \rightarrow E_k$ occurs with probability

$$q^{j \rightarrow k} \geq \frac{1}{|I_1|} P_{A_1}^j Q_{A_1,I_1}^j > 0.$$  \hspace{1cm} (A 18)

Applying this argument repeatedly shows that there is a finite sequence of extension–reduction transformations from an arbitrary equivalence class $E_k$ to $E_m$ with finite probability. We also have that there is a sequence of extension–reduction transformations from an arbitrary equivalence class $E_h$ to $E_m$ with finite probability. Putting these arguments together shows that an arbitrary equivalence class $E_h$ can be transformed into another arbitrary equivalence class $E_k$ via a sequence of extension–reduction transformation with finite probability. This confirms the $\varphi$-irreducibility of the Markov chain $Y$.

To prove the aperiodicity of $Y$, consider an equivalence class $E_k$ with canonical representation $C_k = \{I_1, I_2, \ldots, A_1, \emptyset\}$. By extending the empty island $\emptyset$ and reducing the island $A_1$, we can achieve the transformation $E_k \rightarrow E_k$. The event occurs with probability

$$q^{k \rightarrow k} \geq \frac{1}{|I_1|} P_{A_1}^k Q_{A_1,I_1}^k > 0,$$  \hspace{1cm} (A 19)

which shows that $\gcd(n-1: P(Y_m = E_k | Y_1 = E_k) > 0) = 1$. Aperiodicity for any arbitrary equivalence class then follows from the irreducibility of the Markov chain.

**Remark.** A formula for the probability $q^{k \rightarrow j}$ is provided in appendix A.5.

### A.4. Proof of equation (3.9)

Let $T_d$ be a torus as constructed in §3.3 and fix an equivalence class $E_k$. Suppose that $E_k$ contains $m$ islands. Let $x$ and $y$ be any distinct choice of $m$ points from $X_{d+1}$ and $Y_{d+1}$, respectively. We call the set $x \times y$ a grid. We consider two families of grids, $F_0$ and $F_1$. Grids from $F_0$ are generated without any restrictions on what points may be chosen from $X_{d+1}$ and $Y_{d+1}$. The number of grids contained in $F_0$ is equal to

$$s_0 = \binom{d}{m}^2.$$  \hspace{1cm} (A 20)

Grids from $F_1$ are generated as follows. Fix an integer $H$ and choose $m$ mutually exclusive sets of $H$ consecutive integers $j_1, j_2, \ldots, j_m$ from the set $[1, 2, \ldots, d]$. Choose another $m$ mutually exclusive sets of $H$ consecutive integers $j'_1, j'_2, \ldots, j'_m$ from the set $[1, 2, \ldots, d]$ independently of $j_1, j_2, \ldots, j_m$, and let

$$x = \bigcup_{k=1}^{m} \min(i \in j_k)$$

and

$$y = \bigcup_{k=1}^{m} \min(i \in j'_k).$$  \hspace{1cm} (A 21)

The set $x \times y$ is therefore a grid. $F_1$ is defined as the union of all grids which can be constructed in this manner, where the number $H$ is identical for all such constructions. Letting $P_m$ be the number of ways to choose $m$ sets of $H$ consecutive integers from the set $[1, 2, \ldots, d]$, the number of grids contained in $F_1$ is therefore

$$s_1 = \frac{P_m^2}{m}. $$  \hspace{1cm} (A 22)

This proof involves three steps. Step 1 involves computing the number $P_m$ in equation (A 22). In step 2, we show that $s_1/s_0 \rightarrow 1$ as $d \rightarrow \infty$ with $N$ (number of blocks) fixed. In step 3, we show that for any equivalence class $E_k$ containing $m$ islands, $s_1/s_0 \rightarrow 1$ implies that $n(E_k)/n_0(E_k) \rightarrow 1$, where $n(E_k)$ is the degeneracy factor for equivalence class $E_k$, and $n_0(E_k)$ is defined in equation (3.8).
Step 1. Let \( J_1, J_2, \ldots, J_m \) be \( m \) mutually exclusive sets of \( H \) consecutive integers from the set \( \{1, 2, \ldots, d\} \). To compute \( P_m \), we adopt the convention that \( J_1 < J_2 < \cdots < J_m \), where \( J_a < J_b \) means that there is no integer in \( J_a \) which is greater than or equal to any of the integers in \( J_b \). Let

\[
Q = d - \max_{i \in J_a} \,
\]  

(A 23)

We claim that the largest value of \( Q \) occurs when \( \max_{i \in J_a} i = mH \). Indeed, in this case we have \( J_1 = \{1, 2, \ldots, H\}, J_2 = \{H + 1, H + 2, \ldots, 2H\}, \ldots, \) and

\[
J_m = \{(m - 1)H + 1, (m - 1)H + 2, \ldots, mH\}. 
\]  

(A 24)

Let us now calculate the number of ways of choosing the sets \( J_1, J_2, \ldots, J_m \) such that

\[
\max_{i \in J_a} i = mH + q. 
\]  

(A 25)

In this case, there must be a total of \( q \) integers in the set

\[
S = \left\{1, \ldots, \max_{i \in J_a} i\right\} = \left\{\bigcup_{k=1}^{m} J_k\right\}
\]  

(A 26)

where

\[
K_j = \left\{1, \ldots, \min_{i \in J_a} i - 1\right\}
\]  

(A 27)

and

\[
K_j = \left\{\max_{i \in J_a} i + 1, \ldots, \min_{i \in J_a} i - 1\right\}.
\]  

(A 28)

If \( \min_{i \in J_a} i = 1 \) we set \( K_1 = \emptyset \). If for any \( j \leq m \) we have \( \max_{i \in J_a} i = \min_{i \in J_a} i \), we set \( K_j = \emptyset \). The number of ways of choosing \( J_1, J_2, \ldots, J_m \) such that (A 25) is satisfied is therefore the number of ways of assigning \( q \) indistinguishable objects to the \( m \) ‘cells’ \( K_1, K_2, \ldots, K_m \). This number is given by (p. 85 of [37])

\[
P_m^q = \left(\frac{q + m - 1}{q}\right).
\]  

(A 29)

The total number of ways of choosing sets of \( H \) consecutive integers \( J_1, J_2, \ldots, J_m \) from the set \( \{1, 2, \ldots, d\} \) is then

\[
P_m = \sum_{q=0}^{d-mH} P_m^q.
\]  

(A 30)

Substituting equation (A 30) into equation (A 22) yields a formula for \( s_1 \).

Step 2. It is sufficient to show that

\[
\sum_{q=0}^{d-B} \left(\frac{q + A}{q}\right) \left(\begin{array}{c} d \\ m \end{array}\right) \rightarrow 1,
\]  

(A 31)

as \( d \to \infty \), where \( B = mH \) and \( A = m - 1 \). Using the identities

\[
\left(\begin{array}{c} q + A \\ q \end{array}\right) = \left(\frac{(q + A)!}{q!A!}\right) = \left(\frac{(q + A)!}{A!((q + A) - A)!}\right) = \left(\begin{array}{c} q + A \\ A \end{array}\right)
\]  

(A 32)

and

\[
B = m(H - 1) + 1 + A,
\]  

(A 33)

we can write

\[
\sum_{q=0}^{d-B} \left(\frac{q + A}{q}\right) = \sum_{q=0}^{C-A} \left(\frac{q + A}{A}\right) = \left(\begin{array}{c} C + 1 \\ A + 1 \end{array}\right) = \left(\begin{array}{c} C + 1 \\ m \end{array}\right),
\]  

(A 34)

where \( C = d - m(H - 1) - 1 \). Equation (A 34) was obtained with the diagonal summation identity (p. 104 of [37]). Now, for integer \( R > m \), we have

\[
\left(\begin{array}{c} R \\ m \end{array}\right) = \frac{1}{m!} R(R - 1) \cdots (R - m + 1).
\]  

(A 35)
Equations (A 34) and (A 35) then show that the limit in (A 31) holds if \((C + 1)/d \to 1\). This property is evident from the definition of \(C\).

**Step 3.** Let \(l_1, l_2, \ldots, l_m\) denote the islands contained in canonical representation \(C_k\) for equivalence class \(E_k\). For each \(1 \leq k \leq m\), choose one block \(l_k\) from \(l_k\). The blocks \(l_1, l_2, \ldots, l_m\) are called lagged blocks. A state belonging to equivalence class \(E_k\) can be generated in the following way. Choose a grid \(g\) from \(F_0\), and then choose \(m\) points from \(g\). Then, place the islands \(l_1, l_2, \ldots, l_m\) on the torus \(T_q\) such that (a) the coordinates of the \(m\) tagged blocks have a one-to-one and onto correspondence with the \(m\) points chosen from \(g\) and (b) no two islands overlap. If condition (b) cannot be fulfilled, then we choose another grid from \(F_0\) and start again. We call this procedure \(T\). We can also define another procedure \(T_0\), which differs from \(T\) only in that condition (b) is not enforced. We can also define yet another procedure \(T_1\), which is identical to \(T_0\) except that the family \(F_0\) is replaced with the family \(F_1\).

Now, any state in \(E_k\) can be generated via \(T\). The number of unique states (up to rotational isomorphism) that procedure \(T\) can yield is therefore equal to the degeneracy factor \(n(E_k)\). Similarly, the number of unique states that procedure \(T_0\) can yield is equal to the quantity \(n_0(E_k)\) in equation (3.8) of the main text. Let \(n_1(E_k)\) be the number of unique states that procedure \(T_1\) can yield. For sufficiently large \(H\) and sufficiently large \(d > mH\), we have

\[
n_1(E_k) < n(E_k) < n_0(E_k). \tag{A 36}
\]

It is sufficient to show that \(s_1/s_0 \to 1\) implies \(n_1(E_k)/n_0(E_k) \to 1\). However, from the fact that \(F_1 \subset F_0\), we can write the identity

\[
F_1 = F_0 - F_0 \setminus F_1. \tag{A 37}
\]

\(s_1/s_0 \to 1\) then implies that \(F_0 \setminus F_1 = \emptyset\) and hence \(F_0 \to F_1\) as \(d \to \infty\). However, according to procedures \(T_0\) and \(T_1\), the numbers \(n_1(E_k)\) only differ \(n_0(E_k)\) when the families \(F_1\) and \(F_0\) contain different grids. It therefore follows that \(n_1(E_k)/n_0(E_k) \to 1\) and hence that \(n(E_k)/n_0(E_k) \to 1\) as \(d \to \infty\).

\[\Box\]

**A.5. Formulae for the probability \(q^{i \to k}\)**

In this section, we present a method to compute the probability \(q^{i \to k}\) that the extension–reduction transformation will transform equivalence class \(E_k\) into equivalence class \(E_j\). Our calculation uses a reaction representation for the extension–reduction transformation \(E_k \to E_j\). Let \(R'_k = C_k \cup C_j\), where \(C_k\) is a canonical representation of equivalence class \(E_k\). Then, remove islands from \(R'_k\) until \(R'_k\) contains no pairs of rotational isomorphic islands, and denote the resulting set by \(R_{kj}\). Label the islands in \(R_{kj}\) as 1, 2, \ldots, and \(s\). For the \(i\)th element of \(R_{kj}\), we define the numbers

\[
n_i = \sum_{a \in C_k} 1(a = R_{kj}(i)) \tag{A 38}
\]

and

\[
m_i = \sum_{a \in C_j} 1(a = R_{kj}(i)), \tag{A 39}
\]

where \(a = R_{kj}(i)\) is the event that island \(a\) and the \(i\)th island of \(R_{kj}\) are rotational isomorphs, and \(1(A)\) is the indicator function for event \(A\). We define the reactant set for the transformation \(E_k \to E_j\) as the vector

\[
RCT_{k \to j} = (n_1, n_2, \ldots, n_s), \tag{A 40}
\]

where \(s\) is the number of elements in the set \(R_{kj}\). Similarly, the product set for transformation \(E_k \to E_j\) is defined as the vector

\[
PDT_{k \to j} = (m_1, m_2, \ldots, m_s). \tag{A 41}
\]

By convention, we set the \(s\)th element of \(R_{kj}\) to be the empty island. This implies that \(n_s = m_s = 1\) for all extension–reduction transformations. The labels 1, 2, \ldots, \(s\) correspond to unique elements of \(R_{kj}\). The reaction characteristic for the transformation \(E_k \to E_j\) is the difference between the product and reactant sets,

\[
\Delta_{k \to j} = \{\delta \in PDT_{k \to j} - RCT_{k \to j} : \delta \neq 0\}. \tag{A 42}
\]

A composition is any realization of the reaction characteristic, where two compositions are considered equal if and only if they differ only in the ordering of their elements. A composition of the form \([-1, -1, +1, +1]\) means that two islands \(A\) and \(B\) are lost and two new islands \(C\) and \(B\) are gained during the extension–reduction transformation \(E_k \to E_j\). These islands can be uniquely identified by comparing the
Table 2. The nine possible compositions for the extension–reduction transformation. The compositions are referred to as ‘classes’. ∅ denotes the empty set.

| Class | Composition         |
|-------|---------------------|
| 1     | \{-1, −1, +1, +1\}  |
| 2     | \{-1, −1, +2\}      |
| 3     | \{-1, −1, +1\}      |
| 4     | \{-2, +1, +1\}      |
| 5     | \{-2, +1\}          |
| 6     | \{-1, +1, +1\}      |
| 7     | \{-1, +2\}          |
| 8     | \{-1\}              |
| 9     | ∅                   |

Table 3. Analysis of the \(M = 0\) cases (see proof of the theorem in §A.5).

| Condition | Reaction Representation | Realizable |
|-----------|-------------------------|------------|
| \(A \neq B\) | \(A \rightarrow C\)     | yes        |
| \(C \neq D\) | \(B \rightarrow D\)     |            |
| \(A \neq B\) | \(A \rightarrow C\)     | yes        |
| \(C = D\)    | \(B \rightarrow C\)     |            |
| \(A \neq B\) | \(A \rightarrow C\)     | yes        |
| \(C \neq D = ∅\) | \(B \rightarrow ∅\) |            |
| \(A = B\)    | \(A \rightarrow C\)     | yes        |
| \(C \neq D\) | \(A \rightarrow D\)     |            |
| \(A = B\)    | \(A \rightarrow C\)     | no\(^a\)   |
| \(C = D\)    | \(A \rightarrow C\)     |            |
| \(A = B\)    | \(A \rightarrow C\)     | yes        |
| \(C \neq D = ∅\) | \(A \rightarrow ∅\) |            |
| ∅ \(\neq A \neq B\) | ∅ \(\rightarrow C\) | yes        |
| \(C \neq D\) | \(B \rightarrow D\)     |            |
| ∅ \(\neq A \neq B\) | ∅ \(\rightarrow C\) | yes        |
| \(C = D\)    | \(B \rightarrow C\)     |            |
| ∅ \(\neq A \neq B\) | ∅ \(\rightarrow C\) | yes        |
| \(C \neq D = ∅\) | \(B \rightarrow ∅\) |            |

\(^{a}\) \(A\) cannot be in both the extension set and reduction set of \(A\).

\(^{b}\) \(B \rightarrow ∅\) and \(C \rightarrow ∅\) imply that both \(B\) and \(C\) contain only one block, and hence that \(B\) and \(C\) are rotational isomorphs. But this violates the condition \(M = 0\).

Product set with the reactant set. The event that islands \(A\) and \(B\) are lost and islands \(C\) and \(D\) are gained can be expressed with the reaction representation

\[
\begin{align*}
A & \rightarrow C \\
B & \rightarrow D
\end{align*}
\]

(A 43)

By convention, the island that gains a block is represented by the upper transformation \(A \rightarrow C\), and the island that loses a block is represented by the lower transformation \(B \rightarrow D\). We say that \(A\) is extended to \(C\) and \(B\) is reduced to \(D\) during this transformation. In analysing extension–reduction transformations with the composition \{-1, −1, +1, +1\}, we will also have to consider the case where \(A\) is reduced and \(B\) is extended, as well as the case where \(C\) is swapped for \(D\).

It is therefore possible to characterize a particular extension–reduction transformation with the composition of its reaction characteristic, and we can speak of the composition of an extension–reduction transformation without ambiguity. The following theorem lists all possible compositions of the extension–reduction transformation.
Table 4. Analysis of the $M = 1$ cases (see proof of the theorem in §A.5). We consider the two subcases $C = B$ and $D = A$. For the case $C = B$, all cases with $C = D$ or $A = B$ are not realizable, as the former implies that $B$ is reduced to $B$, and the latter implies that $B$ is extended to $B$. For similar reasons, the cases $C = D$ and $A = B$ are not realizable when $D = A$.

| condition | reaction representation | realizable |
|-----------|-------------------------|------------|
| $C = B$   | $A \rightarrow B$       | yes        |
| $A \neq B$| $B \rightarrow D$       |            |
| $C \neq D$|                          |            |
| $C = B$   | $A \rightarrow B$       | no         |
| $A \neq B$| $B \rightarrow \emptyset$| no         |
| $C \neq D = \emptyset$|                |            |
| $C = B$   | $\emptyset \rightarrow B$| yes        |
| $\emptyset = A \neq B$| $B \rightarrow D$|            |
| $C \neq D$|                          |            |
| $D = A$   | $A \rightarrow C$       | yes        |
| $A \neq B$| $B \rightarrow A$       |            |
| $C \neq D$|                          |            |
| $D = A$   | $A \rightarrow C$       | no         |
| $A \neq B$| $B \rightarrow \emptyset$| no         |
| $C \neq D = \emptyset$|                |            |
| $D = A$   | $\emptyset \rightarrow C$| no         |
| $\emptyset = A \neq B$| $B \rightarrow \emptyset$| no         |
| $C \neq D$|                          |            |
| $D = A$   | $\emptyset \rightarrow C$| no         |
| $\emptyset = A \neq B$| $B \rightarrow \emptyset$| no         |
| $C \neq D = \emptyset$|                |            |

If $B$ is reduced to the empty island, the $B$ must contain one block only, which incorrectly implies that $A = \emptyset$.
We consider the two subcases $C = B$ and $D = A$. For the case $C = B$, all cases with $C = D$ or $A = B$ are not realizable, as the former implies that $B$ is reduced to $B$, and the latter implies that $B$ is extended to $B$. For similar reasons, the cases $C = D$ and $A = B$ are not realizable when $D = A$.

**Theorem.** There are exactly nine compositions for the extension–reduction transformation, as listed in table 2.

**Proof.** Without reference to any particular composition, consider a general reaction representation of the type

$$
\begin{align*}
A & \rightarrow C \\
B & \rightarrow D,
\end{align*}
$$

(A 44)

Unless explicitly mentioned, we will assume that none of $A$, $B$, $C$ or $D$ are empty islands. Let

$$
M = 2 \times 1(A = D, B = C) + 1(A = D, B \neq C \lor A \neq D, B = C),
$$

(A 45)

where $A = D$ is the event that $A$ and $D$ are rotational isomorphs, and $A \neq D$ is the event that $A$ and $B$ are not rotational isomorphs. We have that $M = 0$, 1 or 2. For each of these three values of $M$, we have the possibilities that $A \neq B$, $A = B$ and $\emptyset = A \neq B$, where $\emptyset$ denotes the empty island ($B = \emptyset$ is excluded, because the empty island cannot be reduced). For each of these possibilities, we again have the possibilities that $C \neq D$, $C = D$ and $C \neq D = \emptyset$ ($C = \emptyset$ is excluded, because the empty island cannot
result from extension of an island). We analyse each of the \(3 \times 3 \times 3 = 27\) possibilities in tables 3–5. A reaction is said to be realizable if it is a possible occurrence of the extension–reduction transformation.

Following the analysis in tables 3–5, we can deduce the possible compositions for the reaction characteristic by considering each realizable reaction representation shown in the above tables. For example, the three reactions
\[
A \rightarrow C \quad A \rightarrow C \quad A \rightarrow C \\
B \rightarrow D \quad B \rightarrow C \quad B \rightarrow \emptyset,
\]
contribute compositions \{-1, -1, +1, +1\}, \{-1, -1, +2\} and \{-1, -1, +1\}, respectively, to the reaction characteristic. Proceeding in this manner yields the compositions listed in table 2.

Formulae for \( q^{k\rightarrow j} \) can be easily deduced by analysing each possible composition of the reaction characteristic case by case. For example, suppose that the reaction characteristic for extension–reduction transformation \( E_k \rightarrow E_j \) has composition \{-1, -1, +1, +1\}. Tables 3–5 from the proof of theorem A.4 then show that this composition corresponds uniquely to the reaction representation

\[
A \rightarrow C \\
B \rightarrow D. 
\]  

(A 46)

The islands \( A, B, C \) and \( D \) can be identified by comparing the sets \( RCT_{k\rightarrow j} \) and \( PDT_{k\rightarrow j} \). The corresponding probability for the transformation \( E_k \rightarrow E_j \) is

\[
q^{k\rightarrow j} = n_A n_B (q^{k\rightarrow j}_{BA,DC} + q^{k\rightarrow j}_{AB,DC} + q^{k\rightarrow j}_{BA,CD} + q^{k\rightarrow j}_{AB,CD}), 
\]  

(A 47)

where \( n_A \) is the coefficient from \( RCT_{k\rightarrow j} \), which corresponds to island \( A \), and \( q^{k\rightarrow j}_{BA,DC} \) is the probability that island \( B \) is reduced, island \( A \) is extended, island \( D \) is chosen from the reduction set of \( B \), and island \( C \) is chosen from the extension set of \( A \). According to §2 of the main text (particularly steps T.1–T.4 of the extension–reduction transformation), we have

\[
k^{j\rightarrow i}_{BA,DC} = \begin{cases} 
\frac{p^k_B q^k_{AB}}{I^+_B} & D \in I^-_B, C \in I^+_A \\
0 & \text{otherwise}
\end{cases} 
\]  

(A 48)

where \( p^k_B \) is the probability of reducing island \( B \) (equation (3.4) of the main text), \( Q^k_{AB} \) is the probability of extending island \( A \) following reduction of island \( B \) (equation (3.5) of the main text), \( I^+_B \) is the reduction set of \( B \), and \( I^+_A \) is the extension set of \( A \). The notation \( D \in I^-_B \) means that there is an island contained in \( I^-_B \) that is rotationally isomorphic to \( D \). The transition probabilities for every case in table 2 are listed in table 6.

References

1. Whitesides GM, Gryphobski B. 2002 Self-assembly at all scales. Science 295, 2418–2421. (doi:10.1126/science.1070821)

2. Tour JM. 2014 Top-down versus bottom-up fabrication of graphene-based electronics. Chem. Mater. 26, 163–171. (doi:10.1021/cm402179h)

3. Han P, Akagi K, Canova FF, Mutoh H, Shiraki S, Palmino F, Lamare S, Grandidier B, Cleri F. 2014 Atomic-scale modeling of two-dimensional structures by dispersion forces. J. Phys. Chem. C 118, 12 817–12 825. (doi:10.1021/jp509198w)

4. Copie G, Cleri F, Makoudi Y, Krzeminski C, Berthe M, Cheriotou X, Palmino F, Grandidier B. 2015 Surface-induced optimal packing of two-dimensional molecular networks. Phys. Rev. Lett. 114, 066101–066106. (doi:10.1103/PhysRevLett.114.066101)

5. Page AJ, Ding F, Irie S, Monokuma K. 2015 Insights into carbon nanotube and graphene formation mechanisms from molecular simulations: a review. Rep. Prog. Phys. 78, 036501–036539. (doi:10.1088/0034-4885/78/3/036501)

6. Saathoff JD, Clancy P. 2015 Simulation of graphene nanobomb aggregation and its mediated by edge decoration. J. Phys. Chem. B 119, 4766–4776. (doi:10.1021/jp503020g)

7. Cleri F. 2009 Surface ordering of molecular structures by dispersion forces. Phys. Rev. B 80, 235406–235402. (doi:10.1103/PhysRevB.80.235406)

8. Copie G, Makoudi Y, Krzeminski C, Cheriotou X, Palmino F, Lamare S, Grandidier B, Cleri F. 2014 Atomic-scale modeling of two-dimensional molecular self-assembly on a passivated surface.

9. Copie G, Cleri F, Makoudi Y, Krzeminski C, Berthe M, Cheriotou X, Palmino F, Grandidier B. 2015 Surface-induced optimal packing of two-dimensional molecular networks. Phys. Rev. Lett. 114, 066101–066106. (doi:10.1103/PhysRevLett.114.066101)

10. Perkett MR, Hagan MF. 2014 Using Markov state models to study self-assembly. J. Chem. Phys. 140, 214 101–214 115. (doi:10.1063/1.4878494)

11. Boto R, Contreras-Garcia J, Calatayud M. 2015 The role of dispersion forces in metal-supported self-assembled monolayers. Chem. Soc. Rev. 45, 322–327. (doi:10.1039/c4cs00112b)

12. Liu DJ, Evans JW. 2014 Realistic multisite lattice-gas modeling and KMC simulation of catalytic surface reactions: kinetics and multiscale spatial behavior for CO oxidation on meal (100) surfaces. Prog. Surf. Sci. 89, 393–521. (doi:10.1016/j.progsurf.2013.10.001)

13. Aranson IS, Tsimring LS. 2006 Theory of self-assembly of microtubules and motors. Phys. Rev. E 74, 031915–031930. (doi:10.1103/PhysRevE.74.031915)

14. Jacobs WM, Reinhardt A, Frenkel D. 2015 Theoretical prediction of free-energy landscapes for complex self-assembly. J. Chem. Phys. 142, 021101–021105. (doi:10.1063/1.4905670)

15. Piskalski J, Cicah A, Almarza MG. 2014 Periodic ordering of clusters and stripes in a two-dimensional lattice model. I. Ground state, mean-field phase diagram and structure of the disordered phase. J. Chem. Phys. 140, 114 701–114 712. (doi:10.1063/1.4868001)

16. Robert CL, Casella G. 2004 Monte Carlo statistical methods. New York, NY: Springer.

17. Corander J, Gyllenberg M, Koski T. 2006 Bayesian model learning based on a parallel MCMC strategy. Stat. Comput. 16, 255–362. (doi:10.1007/s10260-006-9399-y)

18. Pena JM. 2007 Approximate counting of graphical models via MCMC. In Proc. of the 11th Int. Conf. on Artificial Intelligence, San Juan, Puerto Rico, 21–24 March, pp. 352–359.

19. Corander J, Bzdok D, Koski T. 2008 Parallel interacting MCMC for learning topologies of graphical models. Data Min. Knowl. Disc. 17, 431–456. (doi:10.1007/s10618-009-0099-9)

20. Pencr J, Nyman H, Koski T. 2015 Labeled directed acyclic graphs: a generalization of context-specific independence in directed graphical model. Data Min. Knowl. Disc. 29, 503–533. (doi:10.1007/s10618-014-0355-0)

21. Sonntag D, Pena JM, Gomaz-Olmedo M. 2015 Approximate counting of graphical models via MCMC revisited. Int. J. Intell. Syst. 30, 384–420. (doi:10.1002/int.21704)

22. He Y, Ji J, Yu B. 2013 Reversible MCMC on Markov equivalence classes of sparse directed acyclic graphs. Ann. Stat. 41, 1742–1779. (doi:10.1214/13-AOS1125)

23. Kuipers J, Molla G. 2015 Uniform random generation of large acyclic digraphs. Stat. Comput. 25, 227–242. (doi:10.1007/s11222-013-9428-y)
24. Cai J, Ruffieux P, Jaafar R, Bieri M, Braun T. 2010 Atomically precise bottom-up fabrication of graphene nanoribbons. Nature 466, 470–473. (doi:10.1038/nature09211)
25. Winfree E. 1998 Algorithmic self-assembly of DNA. PhD thesis, California Institute of Technology.
26. Rothemund PWK, Papadakis N, Winfree E. 2004 Algorithmic self-assembly of DNA Sierpinski triangles. PLoS Biol. 2, 2042–2053. (10.1371/journal.pbio.0020424)
27. Patitz MJ. 2012 An introduction to tile-based self-assembly and a survey of recent results. Nat. Compt. 13, 195–224. (doi:10.1007/s11047-013-9379-4)
28. Landau DP, Binder K. 2009 A guide to Monte-Carlo simulation in statistical physics. Cambridge, UK: Cambridge University Press.
29. Storvik G. 2010 On the flexibility of Metropolis-Hastings acceptance probabilities in auxiliary variable proposal generation. Scand. J. Stat. 38, 342–358. (doi:10.1111/j.1467-9868.2010.00709.x)
30. Andrieu C, Roberts GO. 2009 The pseudo-marginal approach for efficient Monte Carlo computations. Ann. Stat. 37, 697–725. (doi:10.1214/07-AOS574)
31. Alquier P, Friel N, Everitt R, Boland A. 2014 Noisy Monte Carlo: convergence of Markov chains with approximate transition kernels. Stat. Comput. 26, 29–47. (doi:10.1007/s11222-014-9521-x)
32. Sherlock C, Thiery AH, Roberts GO, Rosenthal JS. 2015 On the efficiency of pseudo-marginal random walk Metropolis algorithms. Ann. Stat. 43, 238–275. (doi:10.1214/14-AOS1278)
33. Geyer CJ. 2011 Importance sampling, simulated tempering, and umbrella sampling. In Handbook of Markov chain Monte Carlo (eds S Brooks, A Gelman, GL Jones, XL Meng). pp. 299–300. Boca Raton, FL: CRC Press.
34. R Development Core Team. 2008 R: a language and environment for statistical computing. Vienna, Austria: R Foundation for Statistical Computing.
35. Csardi G, Nepusz T. 2006 The igraph software package for complex network research. Int. J. Complex Syst. 1695, 1–9.
36. The Go Programming Language. See https://golang.org/.
37. Rosen K. 2000 Handbook of discrete and combinatorial mathematics. Boca Raton, FL: CRC Press.