Nanotechnology **10** (1999) 300–307. Printed in the UK PII: S0957-4484(99)01160-5

Robust self-assembly using highly designable structures

Tad Hogg

Xerox Palo Alto Research Center, Palo Alto, CA 94304, USA

E-mail: hogg@parc.xerox.com

Received 21 January 1999

Abstract. Self-assembly creates structures through a statistical exploration of many possibilities. In some cases, these explorations give rise to a few *highly designable* structures that can be formed in exceptionally many ways. Using such structures for self-assembly tasks is a general approach to improving their reliability. This design principle can be applied to a variety of situations, including molecular devices and coordinated behaviours from collections of autonomous robots.

1. Introduction

Manufacturing often builds objects from their components by directly placing them in the necessary arrangements. Common examples include buildings, cars and electronic circuits. This technique requires knowledge of the precise structure needed to serve a desired function, the ability to create the components with the necessary tolerances and the ability to place each component in its proper location in the final structure.

When these requirements are not met, *self-assembly* offers another approach to building structures from components. This method involves a statistical exploration of many possible structures before settling into the final one. The particular structure produced from given components is determined by biases in the exploration, given by component interactions. These biases arise when the strength of interaction between components depends on their relative locations in the structure. These interactions can reflect constraints on the desirability of a component being near its neighbours in the final structure. For each possible structure, the interactions combine to give a measure of the extent to which the constraints are violated, which can be viewed as a cost or *energy* for that structure. Through the biased statistical exploration of structures, each set of components tends to assemble into that structure with the minimum energy for that set. In these terms, self-assembly can be viewed as a process using a *local specification*, in terms of components and their interactions, to produce a resulting *global structure*. The local specification is, in effect, a set of instructions that implicitly describes the resulting structure.

Self-assembly can be very precise in spite of the inherently statistical nature of the process. Examples include chemical reactions driven by diffusive mixing of the reactants, such as the creation of polymers [1,2], proteins [3] and molecular assemblies [4], patterned mesoscale objects [5, 6] and structures consisting of tiny robots [7–10]. This technique can also automatically reconfigure structures when their environments or task requirements change, or when a few components break.

While self-assembly can create a wide range of structures, it has some basic difficulties. First, the precise set of components and interactions that will construct a given global structure can be difficult to determine, especially if the components themselves have manufacturing defects or the environment surrounding the components is noisy. Second, the statistical exploration of different possibilities provides the power of self-assembly, but can also make it difficult to settle on a single final structure, or to resist continual environmental noise once assembled. Third, the assembly process can become stuck in local minima, thereby requiring a long time to identify the final structure.

To help overcome these difficulties, this paper describes some characteristics of the statistical distributions of self-assembled structures. These characteristics in turn suggest a general principle for designing self-assembly processes that minimizes these problems. This principle relies on the fact that many choices for the components and their interactions can produce the same end result. While such choices are often determined by ease of component construction, they also provide an opportunity to improve the performance of the self-assembly process.

2. Examples of self-assembly

Self-assembly is useful for a wide range of applications. Some of these are described in this section, giving a concrete basis for the discussion of aggregate statistical properties in section 3. These examples are also used to suggest a number of extensions to self-assembly beyond the construction of specific structures.

For protein assembly, the components are amino acids arranged in a specific sequence. The global structures are the folded three-dimensional shapes of the proteins. The interactions between amino acids depend on their relative

0957-4484/99/030300+08$30.00

locations in the folded structure. In simple terms, the interactions can be expressed as constraints on the relative affinities of polar and nonpolar amino acids [3, 11]. These interactions are determined by the chemical properties of the components, so the relevant degrees of freedom for the local specification are the choices of the amino acids in the sequence. Engineering applications need not be restricted to naturally occuring amino acids, giving the possibility of using additional components, with different interactions. Various folding polymers offer an even larger range of possibilities for engineered structures [1, 2].

This example suggests a general manufacturing strategy: lay out the components connected in a simple structure, such as a chain or, more generally, a planar graph, and then allow it to fold into the final shape. Because the initial connected structure is much easier to create directly than a complex three-dimensional structure, this technique is a useful combination of directed manufacturing of a relatively simple structure followed by self-assembly of the final, more complex, structure. Alternatively, a nominal initial configuration could be directly constructed while subsequent adjustments to the actual environment or failed components takes place through self-assembly [12].

A wider range of interactions is possible with manufactured particles of various sizes. At a molecular scale, suitable interactions can produce self-assembled structures with new properties [13]. At somewhat larger scales, attaching DNA to small particles allows the use of the selective binding interactions of DNA to create specific structures in response to environmental changes, which can act as sensors [14]. For larger particles, individual shapes and surface properties determine the interactions and hence the resulting structures [5].

The widest range of component interactions arises when the components are complex enough to have computational capabilities, e.g., programmable robots [9, 10, 15–19]. The interactions between robots are determined both by their physical properties (e.g., their weight) and the choices made in their programs (e.g., whether to hold on to a particular neighbour). The component robots assemble into various overall shapes, or global structures. By disconnecting from their neighbours, collections of robots can explore a wider range of topological possibilities than the fixed sequence of amino acids in the protein example.

Collections of robots offer an interesting contrast between self-assembly and direct construction. For instance, each robot could, in principle, be instructed precisely where to go through a predetermined programme. While suitable for relatively small groups of robots and a well-understood environment, this programming task becomes increasingly difficult with larger numbers of robots acting in poorly defined or unpredictably changing environments [20]. An alternate and more robust approach is for the programs to specify only simple local interactions that together produce the final desired structure through an exploration of possibilities, i.e., through self-assembly.

As these examples show, the suitability of a global structure need not just depend on its particular physical shape. Instead, its functional properties may be more important for some applications. For example, a task may require exerting particular forces, e.g., to support or move other objects, rather than a specific shape. Thus, self-assembly can be viewed as a technique for finding combinations of components that satisfy some global constraints on behaviour. That is, self-assembly can solve a combinatorial optimization or a constraint satisfaction problem [21] where the resulting structure should satisfy as many of the constraints as possible. In the case of modular robots, techniques such as simulated annealing [22] and genetic algorithms [23] have been applied to identify suitable structures [15, 18]. Finally, the statistical exploration of self-assembly is also useful in organizing purely software systems [24].

As a final observation on the application of self-assembly, complex artifacts often consist of a series of levels, where the components used at one level of the structure are in turn formed from a smaller set of components [25]. In such cases, some levels could be assembled through direct construction while self-assembly is used to create others. Such situations give further opportunities to design the components and levels to exploit statistical properties of self-assembly described in section 3.

3. Statistical properties of self-assembly

Self-assembly can form precise structures beyond the current capability of direct manufacturing. Choosing appropriate components and interactions to take advantage of this capability can be difficult. The most straightforward technique for designing self-assembly is to examine, with a computer simulation, the neighbours of each component in the desired global structure, and then choose the interactions between components to encourage those neighbours to be close together.

Unfortunately, such a design may inadvertently make some other structure even more stable or introduce numerous local minima in the energies leading to a slow assembly process. Furthermore, the limited range of interactions and component types in some cases, such as for proteins, may not be able to produce the desired interactions among all the components. Even worse, the limited knowledge of the precise global structure desired, environmental noise and component defects may make it impossible to determine the best set of interactions in the first place.

Instead, the statistical nature of self-assembly gives rise to a variety of statistical regularities when the number of components is large. In many cases, these regularities can be exploited to design a robust self-assembly process. Examples of such properties are described in the remainder of this section.

3.1. Designability: resisting component defects

One difficulty with designing self-assembly processes is the indirect, or *emergent*, connection between the interactions and the properties of the resulting global structures. While this difficulty connecting local behaviours with global results arises in many different contexts, a particular consequence for self-assembly is the possibility of errors due to defective components or environmental noise. To help address this problem, it would be useful to arrange the self-assembly so

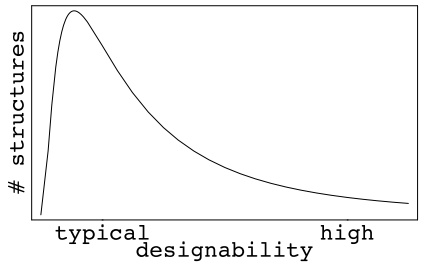

Figure 1. Schematic distribution of designability, i.e., the number of different component configurations producing a given global structure. Each point on the curve indicates the number of global structures with a given designability. The long tail of the distribution indicates a few global structures are highly designable, i.e., far more designable than typical cases.

the desired structures can be formed in many ways, increasing the likelihood that they will be correctly constructed even with some unexpected changes in the components or their interactions. That is, the resulting global structure should not be too sensitive to errors that may occur in the local specification.

This notion is formalized by the *designability* of global structures. Specifically, if G is a global structure and L is a collection of components, or the local specification, then the interactions among the components based on their placement in the global structure can be viewed as defining an energy $E(G, L)$. Small values for the energy correspond to global structures that satisfy many of the constraints. The best global structure for a given set of components L is the one that minimizes this energy. With a sufficient statistical exploration of the possible global structures the minimum energy structure will be the one eventually assembled from L. Notationally, let $L \rightarrow G$ denote the global structure G is the assembled result from L. That is, among all possible global structures, G has the minimum energy for L.

In this context, we can ask how many different component sets assemble to the same global structure G. We refer to this count as the designability of the structure, i.e.,

$$d(G) = |\{L|L \rightarrow G\}| \tag{1}$$

where $|\ldots|$ denotes the size of a set and $\{L|L \rightarrow G\}$ is the set of all component collections that assemble to G.

A given assembly process can then be characterized by a distribution of designability, i.e., the number of global structures with various designability values. This distribution is given by values of $|\{G|d(G) = x\}|$ over a range of choices for x.

A schematic example of such a distribution is shown in figure 1. Significantly, the distribution of designability illustrated here is extremely skewed: a few structures are much more designable than most others. These highly designable structures can be formed in relatively many ways in response to interactions among the component parts. Thus such structures are relatively more tolerant of errors in the choice of components than is typically the case, producing one form of robustness for self-assembly [26].

Designability can be contrasted with measures of the entropy of a global structure G. For instance, among all component collections L that produce G, we can determine the fractions that have various properties, and their correlations. These properties could include choices relevant for the design, e.g., the choice of amino acid at a particular position for a protein or the choice of programmed command action for a particular robot. The extent to which most of the components in a collection L producing G have the same value for a particular property indicates the importance of that property for the overall design. Identifying the important properties for a given global structure is useful for improving the assembly process. Moreover, if the important choices tend to be independent, identifying an appropriate self-assembly process is simplified since each such choice can be considered separately from the others. Because highly designable structures can be formed in a relatively large number of ways, they should be more likely than typical structures to allow independent changes in their components without changing the final assembled structure. With respect to a set of property values S characterizing the component choices, this degree of independence can be characterized by a entropy measure based on assuming the choices for these properties for the components are independent, i.e.,

$$H(G) = -\sum_i \sum_{s \in S} f_s(i) \log f_s(i) \tag{2}$$

where the outer sum ranges over the components in the set L and $f_s(i)$ is the fraction of times, among the component collections L that produce G, that the ith component has property value s. Independence of the property choices for the different components is indicated when $H(G)$ is close to $\log d(G)$, i.e., the entropy characterizes the number of ways G can be constructed. With strong dependences in the choices, $H(G)$ will be much larger than this value. The relation between high designability and relative independence of component choices has been observed for simple protein models [11], but the generality of this relation remains to be investigated.

A simple generalization of this definition of designability is when a variety of global structures have the same functional properties. For instance, if the global structures are physical shapes, rotated versions of these shapes may be functionally identical. Or in the case of tiny robots, matching a shape to within a specific tolerance or delivering a required force, rather than *exactly* matching a specific shape, may be sufficient. In such cases, rather than requiring a single specific resulting structure, a self-assembly process that produces *any* structure with the desired behaviour can be viewed as successful. This leads to considering functionally identical global structures as equivalent when defining designability. That is, an equivalence relation, $G \equiv G'$, can be defined on the global structures that give rise to identical behaviours for the particular application under consideration. Designability then becomes

$$d(G) = |\{L| \text{ for some } G',\ L \rightarrow G' \text{ and } G \equiv G'\}|. \tag{3}$$

Thus an additional ingredient to consider for designing self-assembly processes is the actual performance requirement for the resulting structure, rather than limiting attention just to forming a particular structure.

A further generalization of designability is to continuous measures of the quality of the resulting structure. That is,

instead of asking that the structure have behaviour identical to that desired, it may be sufficient if the resulting behaviour is just close to the desired value. Highly designable structures are still useful in this context, in that they are most likely to still be constructed in spite of variations in the components. However, for comparing various options the measure of designability must include not only *how many* L produce the desired behaviour but also *how well* they do so:

$$d(G) = \sum_{G'} q(G', G)|\{L|L \to G'\}| \tag{4}$$

where $q(G', G)$, in the range 0 to 1, denotes how well the structure G' resulting from L performs the behaviour desired for G. When $q(G', G) = 1$ when $G' = G$ and is 0 otherwise, this definition of $d(G)$ reduces to equation (1).

As a final note, from the viewpoint of constraint satisfaction problems, designability is related to studies of situations that give rise to *stable solutions* [27], i.e., solutions that remain solutions even with a few changes in the constraints. Failing this, solutions should at least be easy to fix when conditions change [28], corresponding to simple repairs that require only adjusting a few components.

3.2. The energy gap: reducing noise sensitivity

As self-assembly operates, the components perform a biased statistical sampling of the possible global structures on the way to forming the final structure. This process allows a variety of complex structures to be formed. Unfortunately, this exploration can also lead to continual and undesired changes in the final global structure. In some situations it will be possible to terminate the exploration by an explicit command (e.g., a broadcast signal sent to a collection of tiny robots). In other cases, this will not be possible: either because the continual changes are driven by uncontrolled environmental noise or are needed to keep the system adaptable to further unexpected changes, e.g., additional weight added to a group of robots supporting a structure. This latter example illustrates a possible trade-off in the design of structures: making them stable against undesired changes but still allowing them to change when appropriate.

Designability addresses the sensitivity of the assembly process to errors in the local specification, or instructions, as described in section 3.1. Another important property is the extent to which desired global structures can resist continual environmental noise once formed. This property of self-assembled structures can be formalized by their energy gap, the difference in energy, due to interactions among the components, between the global structure with the smallest energy for those components and that structure with the second smallest energy:

$$\text{gap}(G, L) = \min_{\{G'|E(G',L)>E(G,L)\}} E(G', L) - E(G, L). \tag{5}$$

Corresponding to the first generalization of designability described above, when the focus is on functionally identical structures, this definition for the gap generalizes to computing the minimum over $\{G'|E(G', L) > E(G, L)$ and $G' \not\equiv G\}$.

Structures with relatively large energy gaps will be more robust with respect to environmental noise than those with smaller gaps. One way to characterize this value [11] for a given structure G is with the average of the energy gap associated with all component configurations that produce G, i.e.,

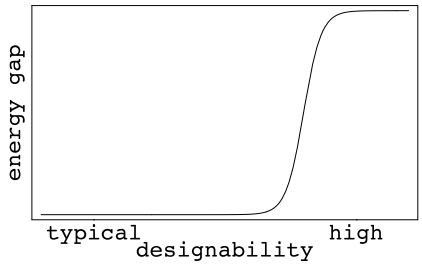

Figure 2. Schematic illustration of one way robustness of global structures can vary with their designability.

$$\text{gap}(G) = \frac{1}{d(G)} \sum_{\{L|L\to G\}} \text{gap}(G, L). \tag{6}$$

This average will be a useful characterization when most values of $\text{gap}(G, L)$ are fairly close to the average.

Designability reflects the behaviour of a given global structure with respect to different sets of components. Thus it characterizes the effect of errors or other changes in the set of components. By contrast, the energy gap characterizes a given set of components with respect to the different global structures that set could form.

Significantly, self-assembly processes with skewed distributions of designability can also produce relatively large energy gaps for the highly designable structures, as illustrated in figure 2. This schematic example illustrates one possibility: an abrupt transition in the energy gap as designability increases. Such a situation would be another example of the transition phenomena commonly found in constrained systems, both physical [29] and computational [30]. Another possibility is for a smooth increase rather than an abrupt change.

This association is easily understood since small changes in the components will usually result in only small changes in the energies of new configurations. This is because the overall energy $E(G, L)$ is usually due to the combination of interactions among pairs of components, based on their location in the global structure. For structures consisting of many components, changing a few components will only modify a small fraction of the total interactions, and typically give a relatively small shift to the energy associated with each global structure. That is, a small change from L to L' will usually result in corresponding small changes in energy from $E(G, L)$ to $E(G, L')$. Such shifts are illustrated in figure 3, which shows the energies associated with four global structures for L and L' as the black and grey points, respectively. Figure 3(*a*) shows a large energy gap so small changes in the energies of all the global structures do not change the one with the minimum energy. On the other hand, small changes in a situation with a small gap are likely to change the minimum energy structure, as illustrated in figure 3(*b*). Thus large gaps are likely to be associated with structures that are energy minima for many different local specifications, i.e., structures that are highly designable.

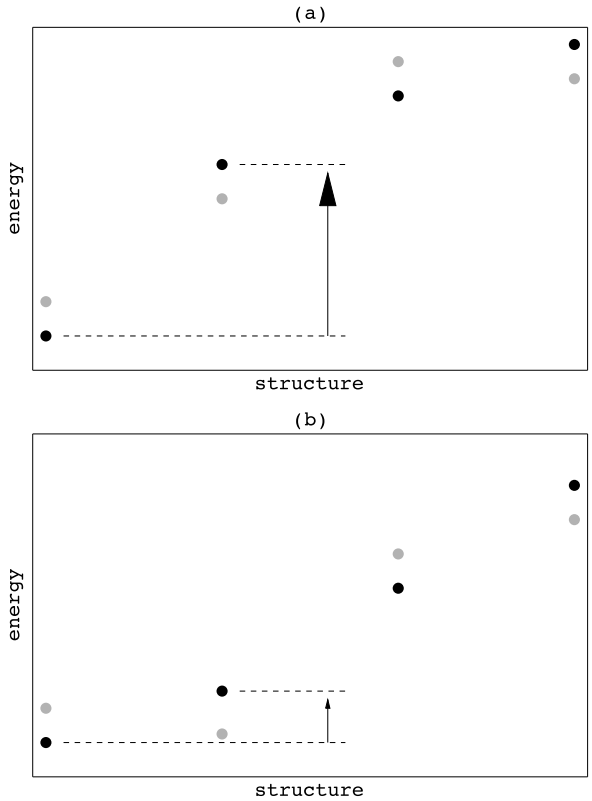

Figure 3. Schematic illustration of the association between large energy gaps and high designability. Black points are energies for different global structures for a given set of components. Grey points are the shifted energy values due to a small change in the component set. The arrows indicate the original energy gaps. With a small gap, the shifts are likely to change the minimum energy structure as illustrated here.

Further insight into the possible changes in energies arises from the nature of the constraints involved in a structure. In many cases these constraints are somewhat frustrated, i.e., they cannot all be simultaneously satisfied completely. For example, we may want a structure that is light and strong. Typically in these cases a small change to a global structure G, producing a *neighbour* structure $G_{\text{neighbour}}$, will introduce changes in the constraints involving just a few parts of the structure, giving rise to a change in energy $E_{\text{neighbour}}$ characterized by the energies involved in just a few constraints. If G is a low-energy structure, such changes are likely to introduce new frustrations and raise the energy, i.e., G has a lower energy than its neighbours and is a minimum in the energy with respect to small changes. Adjusting to these new frustrations will often require further changes, involving additional parts of the structure, to again find a new structure G' with a relatively low energy and, in particular, lower energy than its neighbours.

This discussion suggests two qualitatively distinct causes of the minimum energy gap. First, if G satisfies the constraints particularly well, its energy will be considerably less than any local minima, so the gap will be determined by the typical energy scale of changes in a few constraints, $E_{\text{neighbour}}$. Second, if there are several structures that adjust reasonably well to the frustrated constraints in different ways, the energy differences among these local minima will

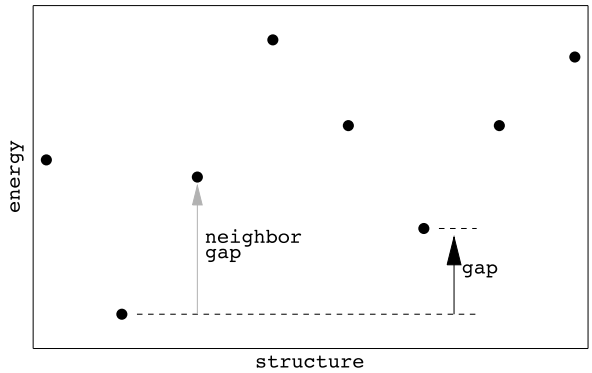

Figure 4. Schematic illustration of energies associated with various global structures. The horizontal axis is ordered so neighbouring structures, i.e., those with similar shapes, are close together. In this case the energy gap is determined by a local minimum shown by the solid arrow. The grey arrow indicates the gap due to a small change in structure.

determine the gap, which will then be considerably less than $E_{\text{neighbour}}$, as illustrated in figure 4.

When this distinction holds, we can expect large gaps, characterized by $E_{\text{neighbour}}$, to arise when some structure can satisfy the constraints particularly well. In other cases, the gap will be smaller and characterized by energy differences among alternate ways of satisfying frustrated constraints. Such situations will result in two distinct regimes for the energy gap as illustrated in figure 3.

3.3. Diversity: avoiding local minima

Designability and the energy gap determine the robustness of the final structure with respect to variations in the components, interactions or environment. Another important question is how long self-assembly takes. While structures can be explored rapidly, especially for molecular sized components, the number of possible structures often grows so rapidly with the number of components that they cannot all be examined in any reasonable amount of time. Fortunately, the biases introduced in the exploration by the interactions can greatly reduce the number of structures that need to be examined.

In some cases, these interactions can also give rise to *local minima*, structures with lower energy than all their neighbouring structures but still with relatively high energy. Such situations lead to sluggish behaviour in both physical [31] and computational [30] systems because the system can remain in a local energy minimum a long time. In such cases, the bias in the exploration toward lower energy structures tends to keep the system in the local minimum.

Models of protein folding suggest this need not be a severe limitation for simple interactions [3]. More generally, the extent to which local minima are a problem depends on several factors. First, for some applications a structure corresponding to a local minimum may be quite adequate, even though it does not satisfy the constraints as well as the true minimum. Second, the number of initial structures leading to the true minimum may be much larger than those leading to local minima. This is often the case when the true minimum is associated with a large energy gap and

the individual constraints are relatively weak. Third, if the system allows a fine degree of control over the component behaviours, as with modular robots, it may be possible to explicitly plan a sequence of motions to avoid local minima [32]. Finally, local minima may be fairly rare.

One way to reduce the number of local minima is through another statistical property of systems with many components: with a sufficient number of ways to change a structure, i.e., a sufficiently high dimensional space of configurations, there is likely to be at least one direction in the space allowing the energy to decrease. This observation can be formalized by examining the statistical distribution of stability, particularly the existence of energy minima, in the space of configurations. Specifically, the stability is determined by the eigenvalues of the matrix of derivatives (the Jacobian) of the forces acting on the system, e.g., due to component interactions. Generally, increases in the size of such matrices or the variation in their values lead to instabilities, i.e., the local minimum becomes a saddle point with at least one direction of change giving lower energy [33–36]. In this case, the variation in the matrix values could arise from a diversity of interactions among the components, e.g., due to a large number of relatively weak constraints rather than a few strong ones. These mathematical properties of large matrices involve systems that can change continuously, which will apply to self-assembly tasks where components can move continuously. For systems with discrete positions, with a sufficiently large number of components the behaviour in many cases will be similar to that of a continuous system. For other cases, extensions of these techniques to discrete dynamical systems will be necessary [37].

Finally, if local minima are a problem, it will be necessary to reduce the bias in the exploration to make changes that temporarily increase the energy more likely. This can be done through an annealing process of slowly cooling the system from a relatively high temperature or, for programmable interactions among robots, using the software equivalent: simulated annealing [22]. However, by requiring that the system wait to find a series of improving changes by chance, such techniques can still be quite slow. When the components are fairly sophisticated, e.g., with programmable robots, a more powerful alternative is the use of computational markets [38] that allow the system to exploit locally coordinated groups of changes that move around the energy barriers [39]. In either case, arranging the self-assembly process so that the desired structures are highly designable, with large energy gaps, while the other structures at local minima are just typical structures with smaller gaps, will allow this annealing process to readily avoid local minima while being less likely to disrupt the final desired structure. Making use of this possibility will require a more extensive analysis of the design possibilities to not only enhance the designability of the desired structure but also reduce the designability of competing structures at the local minima.

4. Decomposability: enhancing robustness

The nature of designability and energy gap distributions affect the robustness of self-assembly. Particularly favourable cases occur when highly designable structures are associated with large energy gaps. While such distributions are seen in models of protein folding [11], an important practical question is how widespread such distributions are.

While the full extent of this question remains to be explored, a particularly simple case is when the system decomposes into several independent parts. That is, supposing the local specification consists of parts $L = (L_1, \ldots, L_k)$ and the global structure consists of corresponding parts $G = (G_1, \ldots, G_k)$. Independence means the energy is the sum of that from the independent parts, i.e.,

$$E(G, L) = \sum_{j=1}^{k} E(G_j, L_j). \tag{7}$$

With this decomposition, if G is the minimum energy global structure for L, then each part G_j must be the minimum for L_j, otherwise another choice for the jth structure will reduce the energy further. This observation means the number of local structures L for which G is the minimum energy configuration, i.e., the designability $d(G)$ of G, is just the product of the designability of the individual parts:

$$d(G) = \prod_{j=1}^{k} d(G_j). \tag{8}$$

On the other hand, the smallest energy gap would be due to a change in just one of the parts, since changes in any other parts will increase the energy further. Thus,

$$\text{gap}(G, L) = \min(\text{gap}(G_1, L_1), \ldots, \text{gap}(G_k, L_k)). \tag{9}$$

These relations result in enhanced tails for the distributions of designability and the gap. In particular, for a wide range of distributions for the individual parts, the product in equation (8) gives rise to a lognormal distribution [40, 41] for $d(G)$. This distribution has a particularly long tail, giving a few highly designable cases as illustrated schematically in figure 1.

Similarly, $\text{gap}(G, L)$ will be governed by the extreme value distribution [42], again with a relatively extended tail. Significantly, these relations can also act to enhance the relation between those cases with high designability and relatively large energy gap. This is because G will be highly designable when most of the parts G_j are so. Further, to have a relatively large gap, the gaps associated with all of the parts $\{\text{gap}(G_j, L_j), j = 1, \ldots, k\}$ must be large. Thus if designability and gap size are somewhat correlated for the parts, these values can become even more related at the high end of the distributions for the combined structures.

Of course interesting self-assembled structures will not have completely independent parts. Nevertheless, this example provides insight into structures with some dependence between the parts, i.e., situations where the energy is nearly decomposable into a sum of contributions from different parts. In such cases equations (8) and (9) will be approximately correct and still lead to distributions with long tails. This extension to nearly decomposable structures is significant because many structures, both natural and engineered, are nearly decomposable [25]. That is, they

consist of a set of parts with relatively strong interactions within each part and weaker ones between them. The example of independent parts is the extreme case where there are no interactions between parts. Furthermore, this near decomposability often extends through a series of levels in the structure, giving hierarchical structures where interaction strengths decrease with the distance between components in the hierarchy, i.e., the number of levels up the hierarchy required to find a common ancestor of the components. At each level in such a hierarchy, equations (8) and (9) will apply approximately, leading to distributions with enhanced tails, at least provided the interactions between parts are relatively weak. This argument suggests highly designable structures will occur in a variety of self-assembly processes.

5. Discussion

Self-assembly of highly designable structures is particularly robust, both with respect to errors in the specification of the components and environmental noise. Thus we have a general design principle for robust self-assembly: select the components, interactions and possible global structures so the types of structures desired for a particular application are highly designable.

Applying this principle requires two capabilities. The first is finding processes leading to highly designable structures of the desired forms. That is, even when a few highly designable structures exist, there remains the question of identifying those structures so as to make use of them for particular applications. Or conversely, finding choices of components and interactions for which desired global structures are highly designable. Identifying such processes uses properties of the tails of statistical distributions, which are more difficult to characterize than those of their central parts. However, some specific examples have been identified, e.g., lattice-based models of protein folding [11] and analyses of the genetic code [43], that suggest evolution has taken advantage of this design principle. Whether or not such simplified models accurately capture the behaviour of natural protein folding, they show such distributions exist in systems with fairly simple interactions and components. Furthermore, this framework is well suited for genetic algorithms [44] to find appropriate processes where possible local specifications and global structures correspond to genotypes and phenotypes, respectively.

The second requirement for applying this design principle is the ability to create the necessary interactions among the components. For simple components, the range of possible interactions may be fairly limited and hence further restrict the search for suitable processes. More complex components, such as tiny robots, can have arbitrary interactions programmed into the components, subject only to restrictions on the timely availability of required information, which tends to enforce the use of local interactions. For example, the interactions could include negotiation among the components [19] or arbitrage opportunities in market-based systems [38, 45]. The ability to program desired interactions is particularly significant in allowing designed systems to more accurately reflect simplifications in the models than would be the case for describing naturally existing assembly situations. Thus, like the development of computational ecologies [20], designed self-assembly provides an example of how prescriptive use of simple models relating global to local behaviours can result in more accurate analyses than their approximate descriptive use for naturally existing systems. Moreover, the flexibility of programmed interactions makes this design principle particularly applicable to collections of robots.

Achieving a general understanding of the conditions that give rise to highly designable structures is largely a computational problem that can be addressed before actual implementations become possible. Thus, developing this principle for self-assembly design is particularly appropriate in situations where explorations of design possibilities takes place well ahead of the necessary technological capabilities [46, 47]. Even after the development of precise fabrication technologies, principles of robust self-assembly will remain useful for designing and programming structures that robustly adjust to changes in their environments or task requirements.

Acknowledgments

I have benefited from discussions with Arancha Casal, Don Kimber, Rajan Lukose and Eric Mao.

References

[1] Muthukumar M, Ober C K and Thomas E L 1997 Competing interactions and levels of ordering in self-organizing polymeric materials *Science* **277** 1225–32
[2] Nelson J C *et al* 1997 Solvophobically driven folding of nonbiological oligomers *Science* **277** 1793–6
[3] Sali A, Shakhnovich E and Karplus M 1994 How does a protein fold? *Nature* **369** 248–51
[4] Martin T, Obst U and Rebek J Jr 1998 Molecular assembly and encapsulation directed by hydrogen-bonding preferences and the filling of space *Science* **281** 1842–5
[5] Bowden N, Terfort A, Carbeck J and Whitesides G M 1997 Self-assembly of mesoscale objects into ordered two-dimensional arrays *Science* **276** 233–5
[6] Smela E, Inganas O and Lundstrom I 1995 Controlled folding of micrometer-size structures *Science* **268** 1735–8
[7] Berlin A A and Gabriel J K 1997 Distributed MEMS: New challenges for computation *Comput. Sci. Engng* **4** 12–6
[8] Brooks R A 1991 New approaches to robotics *Science* **253** 1227–32
[9] Storrs Hall J 1996 Utility fog: The stuff that dreams are made of *Nanotechnology* ed B C Crandall (Cambridge, MA: MIT Press) pp 161–84
[10] Yim M 1994 Locomotion with a unit-modular reconfigurable robot *PhD Thesis* Stanford University
[11] Li H, Helling R, Tang C and Wingreen N 1996 Emergence of preferred structures in a simple model of protein folding *Science* **273** 666–9
[12] Heath J R, Kuekes P J, Snider G S and Williams R S 1998 A defect-tolerant computer architecture: opportunities for nanotechnology *Science* **280** 1716–21
[13] Stupp S I *et al* 1997 Supramolecular materials: Self-organized nanostructures *Science* **276** 384–9
[14] Elghanian R *et al* 1997 Selective colorimetric detection of polynucleotides based on the distance-dependent optical properties of gold nanoparticles *Science* **277** 1078–81
[15] Chen I-M and Burdick J W 1995 Determining task optimal modular robot assembly configurations *Proc. Conf. on Robotics and Automation (ICRA95)*

[16] Kotay K, Rus D, Vona M and McGray C 1998 The self-reconfiguring robotic molecule *Proc. Conf. on Robotics and Automation (ICRA98)* p 424
[17] Murata S *et al* 1998 A 3-D self-reconfigurable structure *Proc. Conf. on Robotics and Automation (ICRA98)* p 432
[18] Pamecha A, Ebert-Uphoff I and Chirikjian G S 1997 Useful metrics for modular robot motion planning *IEEE Trans. Robotics Automation* **13** 531
[19] Yim M, Lamping J, Mao E and Chase J G 1997 Rhombic dodecahedron shape for self-assembling robots *Technical Report* P97-10777 Xerox PARC
[20] Huberman B A and Hogg T 1988 The behaviour of computational ecologies *The Ecology of Computation* ed B A Huberman (North-Holland: Amsterdam) pp 77–115
[21] Mackworth A 1992 Constraint satisfaction *Encyclopedia of Artificial Intelligence* ed S Shapiro (New York: Wiley) pp 285–93
[22] Kirkpatrick S, Gelatt C D and Vecchi M P 1983 Optimization by simulated annealing *Science* **220** 671–80
[23] Mitchell M 1996 *An Introduction to Genetic Algorithms* (Cambridge, MA: MIT Press)
[24] Perlis A J 1985 Another view of software *Proc. 8th Int. Conf. on Software Engineering (ICSE85)* pp 395–6
[25] Simon H 1969 *The Sciences of the Artificial* (Cambridge, MA: MIT Press)
[26] Russell V A *et al* 1997 Nanoporous molecular sandwiches: pillared two-dimensional hydrogen-bonded networks with adjustable porosity *Science* **276** 575–9
[27] Wallace R J and Freuder E C 1998 Stable solutions for dynamic constraint satisfaction problems *Principles and Practice of Constraint Programming (CP98)*
[28] Ginsberg M L, Parkes A J and Roy A 1998 Supermodels and robustness *Proc. 15th Natl Conf. on Artificial Intelligence (AAAI98)* (Menlo Park, CA: AAAI Press) pp 334–9
[29] Wilson K 1979 Problems in physics with many scales of length *Sci. Am.* **241** 158–79
[30] Hogg T, Huberman B A and Williams C 1996 Phase transitions and the search problem *Artif. Intell.* **81** 1–15
[31] Goodstein D L 1975 *States of Matter* (Englewood Cliffs, NJ: Prentice-Hall)
[32] Latombe J-C 1991 *Robot Motion Planning* (Boston: Kluwer)
[33] Gardner M R and Ashby W R 1970 Connectance of large dynamic (cybernetic) systems: critical values for stability *Nature* **228** 784–5
[34] Hogg T, Huberman B A and McGlade J M 1989 The stability of ecosystems *Proc. R. Soc.* B **237** 43–51
[35] May R M 1972 Will a large complex system be stable? *Nature* **238** 413–4
[36] McMurtrie R E 1975 Determinants of stability of large randomly connected systems *J. Theor. Biol.* **50** 1–11
[37] Hogg T and Huberman B A 1985 Attractors on finite sets: The dissipative dynamics of computing structures *Phys. Rev.* A **32** 2338–46
[38] Miller M S and Drexler K E 1988 Markets and computation: agoric open systems *The Ecology of Computation* ed B A Huberman (North-Holland: Amsterdam) pp 133–76
[39] Huberman B A and Hogg T 1995 Communities of practice: performance and evolution *Comput. Math. Organization Theory* **1** 73–92
[40] Aitchison J and Brown J A C 1957 *The Log-Normal Distribution* (Cambridge: Cambridge University Press)
[41] Crow E L and Shimizu K 1988 *Lognormal Distributions: Theory and Applications* (New York: Dekker)
[42] Woodroofe M 1975 *Probability with Applications* (New York: McGraw-Hill)
[43] Vogel G 1998 Tracking the history of the genetic code *Science* **281** 329–31
[44] Forrest S 1993 Genetic algorithms: principles of natural selection applied to computation *Science* **261** 872–8
[45] Clearwater S H 1996 *Market-Based Control: A Paradigm for Distributed Resource Allocation* (Singapore: World Scientific)
[46] Drexler K E 1992 *Nanosystems: Molecular Machinery, Manufacturing and Computation* (New York: Wiley)
[47] Merkle R C 1991 Computational nanotechnology *Nanotechnology* **2** 134–41