Thesis for the degree of Doctor of Philosophy

DNA-Mediated Self-Assembly of Nanostructures

Theory and Experiments

Björn Högberg

Departement of Engineering, Physics and Mathematics
Mid Sweden University
Sundsvall, Sweden 2007

DNA-Mediated Self-Assembly of Nanostructures Theory and Experiments
Björn Högberg

PhD Thesis nr 15
Department of Engineering, Physics and Mathematics
Mid Sweden University
SE-851 70 Sundsvall
Sweden

ISBN: 978-91-85317-39-4
ISSN: 1652-893X, vol. 15

Mittuniversitetets Kopiering
Sundsvall, Sweden 2007

Till Joel och Ragnar

DNA-Mediated Self-Assembly of Nanostructures Theory and Experiments

Björn Högberg
Department of Engineering, Physics and Mathematics
Mid Sweden University

Abstract

This thesis concerns the problem of controlling the construction of complex, functional structures at the lengthscale between chemistry and microtechnology, a problem that is claimed to be *the* problem of nanotechnology. Periodic assembly will not be able to address this problem whereas programmable, or algorithmic, self-assembly might provide a potential solution. Here, I address the question of how the properties of the building-blocks are related to the periodicity of the resulting assembly. By introducing two parameters: the bond uniqueness and number of unique structures, the structural complexity of an assembly system can be estimated. I also give a method for the design of building-blocks after these parameters have been set.

Nanoparticles coated with single stranded DNA could be used to implement programmable self-assembly. However, I argue that a DNA coated nanoparticle by itself cannot be used as a programmable self-assembly building block because it does not have directed bonds. A general scheme for assembling and purifying nanoparticle eight-mers with eight geometrically well-directed bonds is presented.

In a process to make programmable self-assembly building-blocks using nanoparticles, the first goal is the production of dimers with different DNA-functions on the two component particles. I report on the fabrication of anisotropically functionalized dimers of nanoparticles of two different sizes. As a result of their anisotropy, these demonstrator building blocks are shown to assemble into curved structures.

Further, I present the current status of our experiments where DNA-scaffolded origami is used in conjunction with nanoparticles and proteins. By using DNA-modified nanoparticles that hybridize onto the origami, precise control of particle positions at the nanoscale could in principle be achieved. I report on our recent experiments on this system and discuss open problems and future applications.

Keywords: programmable self-assembly, algorithmic self-assembly, DNA, gold nanoparticles, unique structures, bond uniqueness, complexity, nanostructures, bottom-up, nanotechnology, proteins, bio-nanotechnology

List of papers

I. Programmable Self-Assembly Unique Structures and Bond Uniqueness, B. Hogberg and H. Olin, *Journal of Cumputational and Theoretical Nanoscience*, **3** (3), p. 391 (2006)

II. DNA Coated Nanoparticle Eight-mers as Programmable Self-Assembly Building Blocks, B. Hogberg, J. Helmersson, S. Holm and H. Olin, *Proc. Foundations of Nanoscience 2,* p. 219, ScienceTechnica, Snowbird, Utah (2005)

III. Study of DNA coated nanoparticles as possible programmable self-assembly building blocks. B. Hogberg, J. Helmersson, S. Holm and H. Olin, *Applied Surface Science* **252** p. 5538 (2006)

IV. Anisotropically DNA-Functionalized Nanoparticle Dimers, B. Hogberg and H. Olin, *Accepted for publication in European Physics Journal D*

V. DNA-scaffolded nanoparticle structures, B. Hogberg and H. Olin, *Accepted for publication in Journal of Physics: Conference Series*

Related writings, not included:

Comparison Between Unique Addressing and Programmable Self-Assembly, B. Hogberg and H. Olin, *Proc. Foundations of Nanoscience 2,* p. 37, ScienceTechnica, Snowbird, Utah (2005)

A Method for Automated Tile Systems Design, B. Hogberg, M. Olsen and H. Olin, *Proc. Foundations of Nanoscience 3,* p. 215, ScienceTechnica, Snowbird, Utah (2006)

Programmable self-assembly Theoretical aspects and DNA-linked nanoparticles, B. Hogberg, J. Boo J. Liu-Helmersson and H. Olin, book chapter in: *Systems Self-Assembly: Multidisciplinary Snapshots*, Elsevier, *in press* (2007)

Method and computer program for the self-assembly of a nanostructure, H. Olin, B. Hogberg and L. Glans, Patent Appl. WO2005102913

The following papers are not included in this thesis, they are appended as a part of my licentiate thesis:

High-T_c Superconducting Junctions for Integrated Circuits Dept. of Microelectronics and Nanoscience, Chalmers University of Technology, Tehnical report nr. GIPR - 370, Goteborg (2002)

1. Small Scale Integrated Technology for HTS RSFQ Circuits, M.Q. Huang *et al.*, *IEEE Trans. Appl. Supercond.*, **11**, p. 558 (2001)

2. Submicron $YBa_2Cu_3O_x$ ramp Josephson junctions, P. V. Komissinski, B. Hogberg, A. Y. Tzalenchuk, and Z. Ivanov, *Applied Physics Letters*, **80**, p. 1022 (2002)

3. Novel in-situ fabricated Josephson junctions: Trilayer on s Substrate Slope, *IEEE Trans. Appl. Supercond.*, **13**, p. 794 (2003)

Comments on the Papers and My Contribution

Papers I, III and IV are peer-reviewed journal articles. Papers II and V are peer-reviewed conference proceedings. The papers II and III overlap each other, however, paper II covers the basic ideas in more detail than paper III, and is included for clarity.

I. The theoretical work is almost entirely due to me, as well as the writing of the main bulk of the paper. Both authors discussed and commented on the manuscript.

II. The idea was developed by H.O. and me. The experiments were mainly performed by me. Some of the electrophoresis was performed by J.H. I wrote the paper. All authors discussed and commented on the manuscript.

III. The idea was developed by H.O. and me. The experiments were mainly performed by me. Some of the electrophoresis was performed by J.H. I wrote the paper. All authors discussed and commented on the manuscript.

IV. The new ideas introduced in this paper are due to me. The experimental work as well as the writing of the paper was performed by me. Discussion and commenting on the manuscript was done by both H.O and me.

V. I performed the experiments and wrote the paper. Both authors discussed and commented on the manuscript.

The ideas presented in chapter 6 are due to me.

List of Figures

Contents

Chapter 1

Introduction

THIS THESIS IS AS SUMMARY TYPE THESIS. My main research contribution lies in the peer-reviewed articles appended at the end. These articles contain all the technicalities, data and conclusions of my research. Therefore I see no need to repeat all the technicalities of the papers in this summary. Instead, my intention has been to write a text that can appeal to a larger community than the hard-core self-assembly scientists. It is intended to evoke interest and hopefully attract some new people to the area of nanoscale self-assembly. Those who want to know all the details and learn more, are referred to the papers. This general summary is therefore deliberately less technical than normal scientific papers and I have tried to put readability an pedagogical considerations before absolute stringency at all times.

Because of the way I have chosen to organize this thesis, the concepts are not naturally introduced in a chronological order, rather, citations are given where appropriate for the context. Therefore I think it would be in order to start with a quick summary of the historical background and some of the most influential work that this thesis, and the entire field, is based upon: The idea to use DNA as a construction material first came from Seeman [1]. Adleman [2] demonstrated the idea that DNA can actually be used to compute and a few years later, Winfree [3] came up with the basic ideas for programmable, or algorithmic, self-assembly with DNA. Pioneering work has also been performed by the groups of Reif and LaBean [4, 5], Yan [6], Shih [7], Klavins [8], Turberfield [9], Jaeger [10] among others. Worth mentioning at this point is also Rothemund [11]. This short list is far from complete, our introduction in paper I gives a somewhat more elaborate account. For a thorough overview of the history of the field see some of these

excellent reviews: [12, 13, 14, 15].

The first three chapters of the thesis are of theoretical nature. They summarize findings from papers I to III and in some regards, findings from all papers. I begin in chapter 2 with a general survey of some simple, but important, questions, like why we study self-assembly and why the type of bonding interactions are so important to obtain a good assembly. In self-assembly for technological applications, periodicity and order are unwanted things. I try to explain why in chapter 3. In this chapter I also start to introduce some of my contributions to the field, like the classification of assembly types in terms of a parameter that I call "bond uniqueness". In chapter 4 I return to the subject of building-blocks. What must they look like to form the things we want? As it turns out they must be pretty complex themselves. This is why our idea, presented in papers II to IV, is so important if one wants to make self-assembly using nanoparticles. In the same chapter, the concept of unique structures, and a method to design building-blocks, is introduced and I try to convince the reader of its usability when designing programmable self-assembly systems.

Chapter 5 deals with the experiments that we have made in our lab and highlights important points from papers II through V. I begin by introducing the reader to the wonderful world of DNA, followed by a discussion about how to obtain those "complex" building-blocks that chapter 4 showed us that we need. I show that our method to make DNA-nanoparticle building-blocks really works by making them assemble into curved structures. Further, I introduce the reader to the fascinating new field of DNA origami, and discuss some of our recent experiments about sticking nanoparticles to these origamis (presented in paper V). This is a work in progress and the results are somewhat inconclusive.

To get the readers imagination going I finish this thesis with three unpublished ideas that might lead to new results. First, I present a connection between my theoretical self-assembly work and the notion of statistical complexity, an idea that I do not know what it might be good for. The other ideas I do know what they might be good for but I do not know if they are experimentally possible: a nanoparticle container for drug delivery or similar purposes, and a nano DNA-sequencer made of DNA-origami, nanoparticles and a viral bacteriophage portal motor.

Chapter 2

The importance of self-assembly

BEHIND EVERY COMPLEX OBJECT WE CAN THINK OF lies a long path of self-assembly. Take the human brain as an example in my opinion, the most complex object ever found in the universe. One human brain contains more synapses than there are stars in the galaxy, moreover, the synapses are not merely scattered in a random fashion, like the stars, but ordered in an intricate pattern that makes thought possible.[16] The brain, and life itself is self-assembled and self-regulated. How can we expect to beat the complexity presented to us by nature, if we do not learn to use this potentially powerful technique?

2.1 What is self-assembly?

Is not everything self-assembly? In a sense yes. Some scientists talk about "self-assembly" of galaxies, and everything we observe in our society, can to an extreme be said to have self-assembled form a pool of chemicals. This would be taking it too far to be a useful definition for this thesis. I will therefore concentrate on spontaneous processes where the equilibrium is an of aggregation small entities into a larger, more ordered assembly.

Self-Assembly (wide definition) A reaction where the equilibrium state is a larger aggregation of the initial, smaller building blocks.

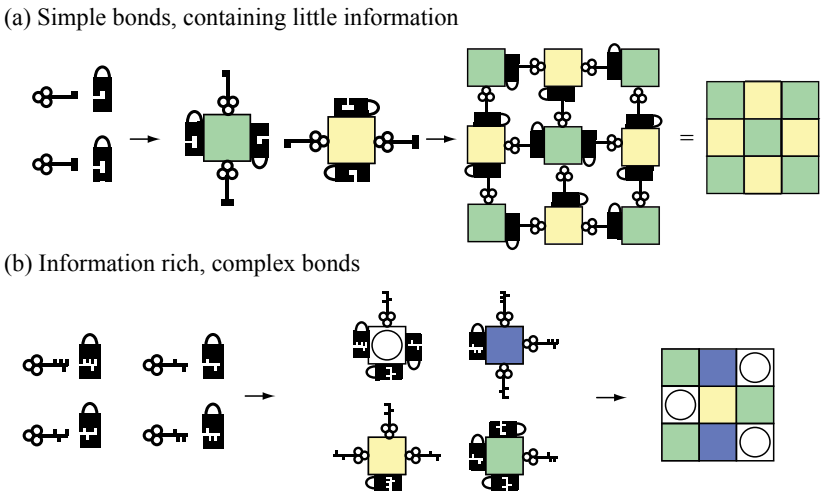


Figure 2.1: Self-assembly where a bunch of keys and keyholes represent the bonds. The more complex the keys are, the larger the variety of keys one can make and more complex assemblies can be fabricated. Complex, information rich bonds, increase the degree of recognition.

So burning of C in O_2 to produce CO_2 is self-assembly? Yes, unfortunately. However, to distinguish ourselves from chemists*, the scientific community of nanoscientists has evolved a concept of self-assembly that might be defined as something like this:

Self-Assembly (narrow definition) A reaction, driven mostly by diffusion, where large molecules or macroscopic objects are aggregated via weak attractive forces, into even larger supra-molecular, or macroscopic objects.

Let us ignore the difficulty in setting the limit as to what is a large molecule and what is a small one, and use this definition in the rest of the thesis.

2.1.1 Why weak forces?

By using weak, reversible bonds, the degree of recognition increases. In a self-assembly system containing a large number of different types of building-blocks, the bonds that are formed must be highly specific. Compare a self-assembly system to a bunch of keys and keyholes like in fig. 2.1. Let each

*Chemists have actually been working with self-assembly for centuries and in some sense, nanoscience is just chemistry on a larger scale.

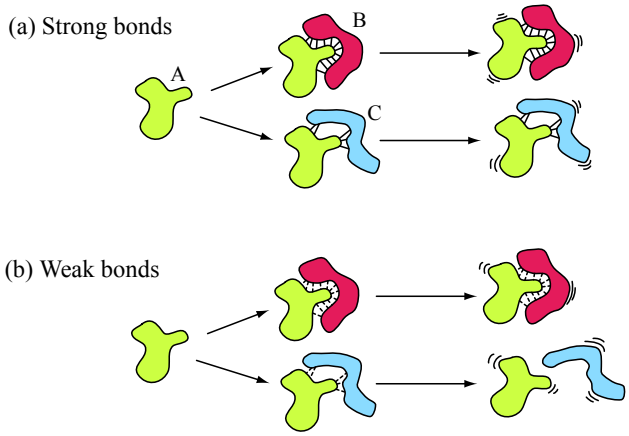


Figure 2.2: The importance of weak reversible bonds for a good recognition effect in self-assembly. In (a) the building-block A randomly encounters either B or C, where C is an unwanted, less favorable reaction. With time nothing changes since the partial bonds are so strong. In (b) however, the same unwanted reaction is quickly reversed because the bonds are to weak to hold together partly mismatched building-blocks during vibrations induced by thermal motion or stirring.

building-block contain both keys and keyholes, each key will fit a keyhole on another building-block so that the blocks may form a bond. Now, if we only have access to a very limited amount of types of keys and keyholes [fig. 2.1(a)], we are very limited in the types of self-assembly we may synthesize. If we want to construct large, complex assemblies with few errors, we simply need a large amount of different key-types and keyhole-types. This is something that I like to call *bond complexity*. Bonds that require a lot of information to describe, complex keys [fig. 2.1(b)], can be made in a larger variety of ways than bonds that require less information, *i.e* using simple keys. The degree of recognition is higher for complex bonds.

In real, physical, self-assembly systems, the bonds can be covalent bonds, hydrogen bonds, surface tension type of bonds or almost any kind of force that one can come up with. Remember that in self-assembly, according to our definition, we are dealing with supramolecular or macroscopic building-blocks. At this scale, *one* bond (one key and keyhole) is just a simplified notation for something that is in fact a large amount of molecular bonds. Because of this, all bonds may end up being only partially matched in practical self-assembly. Such partial matches are the enemy of perfect assembly. Therefore it is important that partially mismatched bonds do not mess up the entire assembly process. This is where weak reversible bonds come in. A

weak reversible bond that is partially matched will easily be broken again. In this way the only important reaction becomes the perfectly matched one, see fig. 2.2.

To sum up, we need complex, information rich bonds. These types of bonds are in fact in all practical cases, composed of a large array of smaller molecular bonds. Each one of these smaller molecular sub-bonds must be weak enough to be able to break if the other parts of the total bonding array are not matched.

2.1.2 Let us ignore dynamic self-assembly

Note that by using the definition on page 4 I will not be discussing what has become known as "dynamic" self-assembly.[17] In dynamic self-assembly energy is continuously used up to drive a process. There is a constant flow of energy through the system, as opposed to the equilibrium type of self-assembly mostly dealt with in this thesis.

In equilibrium self-assembly, parts are mixed and the system relaxes into an equilibrium where the building-blocks will have aggregated into some (hopefully) useful structure, and all important motion comes to a halt. In dynamic self-assembly things continue to move around all the time. An example of dynamic self-assembly might be the patterns formed by a flock of migratory birds. They "self-assemble" into a pattern but are constantly using energy to maintain the pattern. Another example is the living cell. This little machine is constantly boiling with activity and continues to dissipate energy until it dies. The development of multicellular organisms and other more complicated dynamic self-assembly processes are also often refereed to as *self-organization* to distinguish it from what I call equilibrium self-assembly.[18] It should be noted that there is no stringent technical definition of what is dynamic self-assembly and what is equilibrium self-assembly. Some ambiguities exist and the task of classifying different self-assembly types, is to a large extent an open problem.

Although dynamic self assembly is very interesting, I believe that one must fully understand equilibrium self-assembly first, in order to start to grasp the concepts of its dynamic counterpart. In fact in most examples of dynamic self-assembly, like the growth and maintenance of living organism, equilibrium self-assembly plays a crucial role. I would like to go so far as to say that dynamic self-assembly is nothing more than interacting subsystems of equilibrium self-assembly. Take the cell as an example once more. In the cell, practically all essential processes involves attaching proteins or DNA/RNA at the right place. It is only when we take all these processes

together that we start to see something like the dynamics of the system. At the root it is still all about building-blocks that are swimming around and establishing bonds between each-other *i.e.* equilibrium self-assembly.

2.2 Self-assembly in nature

Most of our inspiration as nanoscientists comes from nature. Nanotechnology has a great deal to learn from bioscience but it might also be the other way around. "Unless you can t build it yourself, you don t understand it" is a qoute that I think is due to Langton [19], and this has a great deal of truth in it. By actually trying to build nanostructures and nanomachines, our understanding of self-assembly will greatly increase and new explanations for the marvelous complexity we see in nature are bound to come. The study of self-assembly therefore has a profound fundamental importance since it will in principle, teach us more about the origin of life. If you ask me, such an example has in fact recently been found and that is the connection between DNA-origami and the ribosome, see the following section.

Some of the processes we see in living things are complicated and involves molecular motors driven by hydrolysis of nucleotide triphospates. Examples of this include the action of the enzyme DNA-polymerase during synthesis of a new DNA-strand. On the other hand, there are quite a few clear-cut examples of self-assembly in nature, and in this section I give an introduction to a few of these cases.

2.2.1 The ribosome

The ribosome is a molecular complex of RNA and proteins that is responsible for the synthesis of new protein chains in the cell. It reads, with the help of tRNA, the genetic code of a mRNA strand and puts together a growing chain of amino-acids. The size of the ribosome is about 25 nm. The ribosome is fascinating in many ways and for nanoscientists it is something of a dream-machine. It is a complex and highly functional "nanorobot" and it is produced by self-assembly in its simplest form. Take the building-blocks, proteins and RNA, mix it in a tube under the right pH and salt-concentration, *et voilá*, you have made ribosomes.

Fig. 2.3 shows the principle of ribosome assembly. The ribosome in the bacteria *Escherichia coli*, is made up of a large and a small subunit. RNA and proteins are the basic building-blocks in the subunit assembly [fig

Although important parts are also about molecules dissociating self-dissasembly.

2.3(a)]. The assembly process of the subunits themselves has been studied extensively. [20] It has been found that the ribosomal RNA first folds up in a complicated secondary structure [fig. 2.3(b)]. Thereafter, the proteins attach to this RNA scaffold. By excluding certain proteins in different experiments one has been able to draw an assembly map, showing the order of assembly (fig. 2.3(c)). This map should be interpreted as follows: A protein (denoted by Sx in fig. 2.3) can only be assembled to the growing ribosomal subunit if all the proteins above have already been assembled.

Why does the ribosome self-assembly work? It is a very complicated structure and its assembly is taking place in the cell where a lot of other disturbing proteins are floating around. The reason why it works can probably be found, surprisingly enough, by looking at some of the most recent experiments in artificial self-assembly, a technique called DNA-origami, see section 5.3.1 in chapter 5. The method where a long information rich molecule is mixed with a large amount of smaller constituents that attach to this scaffold, seems to be very robust. In DNA-origami, both the scaffold and the smaller constituents are nucleic acids. In the ribosome, the smaller constituents are proteins and the long scaffold is a nucleic acid. This connection between artificial self-assembly and an assembly found in nature is very interesting and I think it merits some serious thought.

2.2.2 Virus particles

Viruses are malicious nanomachines that specialize in the art of using the cells as copy-machines. A virus is nothing more than a piece of genetic information in the form of RNA or DNA and a protein as protection. A typical virus life cycle proceeds in four different steps. (1) The virus manages to get into a cell and is disassembled to release its RNA or DNA. (2) The virus genome is replicated by the cell replication mechanism. (3) Viral proteins are manufactured by the host cell translation machinery. (4) The produced viral proteins and RNA/DNA self-assemble to form new virus particles. This virus multiplication causes the cell to lyse (break open) and release of the progeny virus particles follows.

One of the first experiments in macromolecular self-assembly was the mixing of the constituent building-blocks that form the *tobacco mosaic virus* (TMW). When the dissociated RNA and proteins were mixed in a test tube, fully functional, infectuous TMW were formed.[22] The TMW is a very simple, rod like virus, in which the RNA forms a scaffold for the assembly of 2130 copies of a single protein. See fig. 2.4(a). More recent experiments have shown that many viruses use more elaborate assembly schemes, often

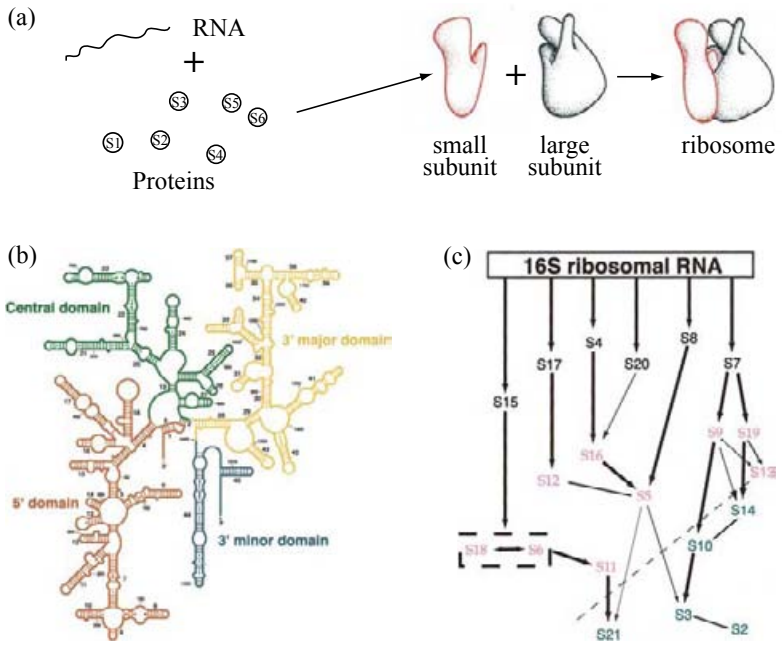


Figure 2.3: Ribosome assembly. (a) When ribosomal RNA strands are mixed in solution with the appropriate proteins, a ribosomal subunit will form by self-assembly. Two types of subunits then self-assemble to form a ribosome. (b) The secondary structure (folding pattern) of the 16S ribosomal RNA from e-coli. (c) Assembly map of e-coli ribosome subunit assembly. Each protein is labeled by an S followed by a number. The map shows the order of assembly, so S17 must be attached to the growing ribosome before S12 for example. ((b) and (c) by G.M. Culver [20])

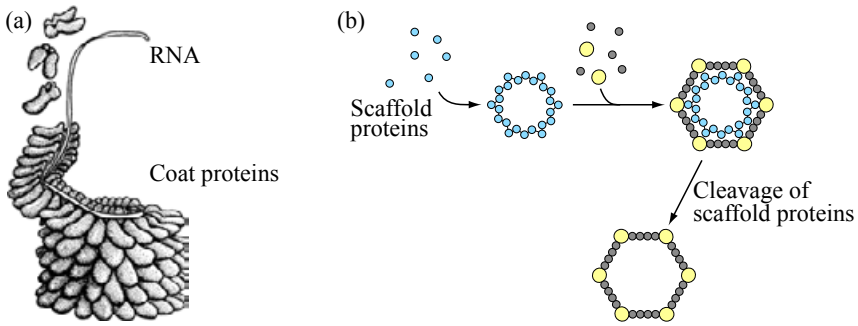


Figure 2.4: Self-assembly of protein particles. (a) Tobacco mosaic virus is a rod-shaped virus that is formed by self-assembly of coat proteins that attaches directly to the RNA-genome and folds it into a helix. (b) Self-assembly pathway of many types of viruses: First a simple scaffold is assembled and then the actual virus particle proteins forms a compartment by assembling on the scaffold. Subsequently, the scaffold is dissolved and the mature virus particle, or *capsid* is ready to be packed by DNA or RNA. (a) reproduced from [21]

involving the assembly of a temporary protein scaffold, see fig. 2.4(b).

Viruses thrive on self-assembly. They have no complicated cellular machines[1] to help them build their capsules and they cannot afford to carry so much RNA or DNA with them to encode for such machines. They must rely solely on efficient self-assembly of their proteins within the cell plasma. Because of this, viruses are a great source of inspiration for nanoscience.

2.2.3 Prions

Prions are proteins that have been misfolded in an extreme way. A prion protein that is misfolded causes nearby proteins of the same kind to misfold as well. The misfolded proteins assemble in a long protein fibre, causing various neurological diseases for example bovine spongiform encephalopaty (BSE or "mad cow disease"), see fig. 2.5. Because of the misfolded proteins ability to deform other, normal proteins, the prions are infectious.

Prion fibres, are thus formed in a two step assembly process: (1) conformational change of an incoming protein and (2) the binding of the shape-transformed protein to a chain of misfolded proteins. Prions give an important hint to nanoscience: Maybe structural information could be "copied" from one nanostructure to another in a similar way? If it could be done, we would probably soon build nanodevices using prion-like self-assembly.

[1] Some viruses actually do build machines , see sect. 6.3

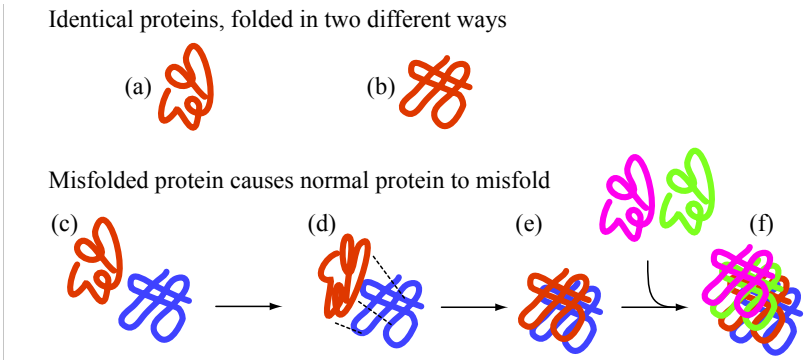


Figure 2.5: The assembly of a prion fibre. Proteins (a) and (b) have identical amino-acid sequences. In the cell the protein always folds into shape (a) and very rarely to shape (b). (c) When a protein of shape (a) encounters a protein of shape (b) it starts to refold, (d). After it has refolded, (e), it takes the shape of protein (b) and is firmly attached to that protein. Adding more proteins of type (a) leads to the assembly of a long fibre of (b)-shaped proteins.

2.3 Self-assembly in technology

Traditionally, the construction of a technological device starts at the drawing table of an engineer. The engineer designs the structure of the device, *i.e.* makes a blueprint, and then gives this blueprint to the guys at the workshop who then manufacture it. If one wishes to construct very small devices, like electronic components, the workshop is replaced by large, complex machines in a cleanroom. These machines, take the blueprint as an input, and project its structure down to the smallest possible scale using focused and highly ordered beams of light, electrons or even ions. This is called lithography. In principle it is the same thing as stone carving, but using extremely small hands. Technologies for the production of really small things that follows this pattern are called *top-down technologies*.

Top-down technologies A top-down technology is a technology to produce complex patterns and devices using machines that transfer and shrink large scale drawings into micro and nanoscale patterns.

Top-down technologies are great. They have helped us to build computers, phones and satellites, among other things. However, when it comes down to nano, using top-down technology quickly becomes unfeasible. There are three main reasons for this:

1. As the scale is decreased, the resolution of a top-down technology

becomes a real problem. It is very hard to make a pattern at the nanoscale using energy (light, electrons) that has a wavelength of similar length-scale as the structures one wishes to make.

2. Lithography is inherently two-dimensional. The patterns produced by top-down methods are two dimensional and using the third dimension requires a lot of effort like stacking 2D layers on top of each-other. The alignment of consecutive two-dimensional layers becomes a limiting factor.

3. Top down methods are always more or less serial. For example, the making of some 10^{23} devices by e-beam lithography takes an enormous amount of time because each pattern has to be exposed by a beam of electrons with a certain finite scanning rate. If each copy of the pattern can be exposed in a millisecond, the machine would be busy for 10^{12} years.

Enter bottom-up!

Bottom-up technologies A bottom-up technology is a technology to produce complex micro- and nano-scale patterns and devices using self-assembly of small building-blocks.

I like to say that top-down is the art of building big and expensive machines to produce small patterns. Bottom-up on the other hand, is the art of creating small patterns by mixing materials and then use big and expensive machines to look at the products you have made. This is not really fair, since the big expensive machines you need for looking at your products (microscopes) are really used in the top-down approach as well. Hence, top-down definitely needs more machines.

I would like to stop here by giving my opinion about the nature of nanotechnology. The words "nanotechnology" and "nanoscience" are used extensively in the debate today. It has almost come to a point where everything is nanotechnology. Terms that encompasses everything quickly looses its meaning. If you are talking about nanoscience today, you could be talking about everything between paints, that uses nanoparticles as a component, to the nanoscale-sized gates in the Pentium transistors. I would like to claim, that nanotechnology is all about building things in a new way, using self-assembly. Not merely a notation for *everything* that has something in it that can be measured in nanometers. That is, nanoscience is the same as *bottom-up*.

2.4 But what is it really good for?

At this point some readers might argue: Okay, now he has told me what self-assembly is and what it is not. Still no clues about what those "complex patterns" are to be used for. I agree, I have not gone into any detail as to what end we will use self-assembly and bottom-up in the future. I do not intend to go into any details about this because it would be tantamount to trying to predict the future, I shall, however, briefly describe some areas where self-assembly is believed to yield new exciting products. More on this in chapter 6, where I present some of my ideas for future work.

2.4.1 Electronics

The first area that came in my mind when I started to work on self-assembly was the creation of nanoelectronic circuits. Imagine the amount of information we could store on a square centimeter if we could make truly nanoscale patterns for electronics. Another issue is the speed of electronics. The smaller components, the faster the circuit. And least but not last, the cost of production: If self-assembly becomes feasible, the price per unit could become vanishingly small. A mole of Pentiums is better than just a few billion Pentiums.

2.4.2 Biosensors

If we can create complex nanoscale patterns with high accuracy, we would certainly be working at the scale of molecular biology. The possiblity of directly manipulating proteins and nucleic acids, and doing measurements on single molecules would open up a path of great opportunities for biomedicine. Imagine a device where protein activity is turned on or off at the switch of a button or a device where proteins are aligned to do molecular work in a production-line manner. The possibilities are numerous and very appealing for the future of biotechnology.

2.4.3 Nanomachines, bio- or not.

The last exampe, proteins doing work together is closer to another category of possible devices: Nanomachines. Nanomachines already exist in nature, the ribosome and many enzymes are examples of devices that can be appropriately referred to as nanomachines. If we could start to build nanomachines of the same complexity as the ribosome, the applications would be endless in biosciences, material sciences and many other areas.

2.5 Summary of Chapter 2

Self-Assembly A reaction, driven mostly by diffusion, where large molecules or macroscopic objects are aggregated by weak forces at the equilibrium, into even larger supra-molecular, or macroscopic objects.

Information rich bonds are needed to have a high degree of specificity and recognition so that the self-assembly system can form complicated structures.

Weak reversible, bonds Because the bonds must be complex, there will always be a possibility of forming partial, incorrect connections between building-blocks. These partially bonded moieties must fall apart spontaneously so that that they do not disturb the self-assembly process.

Dynamic self-assembly A class of self-assembly where everything is moving, all the time. Energy is continuously dissipated. The living cell as a whole, is such a system. Also called *self-organization*.

Nature provides a large variation of self-assembly processes, like the assembly of ribosomes, viruses and prion fibres, and is a great source of inspiration for nanoscience. The understanding of self-assembly is of fundamental importance for both life-science and nanotechnology.

Bottom-up fabrication *is* self-assembly from a technology perspective. It is also a narrow (and good) definition of what nanotechnology is.

Chapter 3

The importance of disorder

DISORDER IS THE NEW ORDER. This phrase is not taken from a fashion magazine, it is my humble simplification of the conclusions from this chapter. A common misconception about "order" is that it is a good thing. However, if matter in the universe was highly ordered, life would not be able to exist, in particular, self-assembly would not work. In this chapter I shall try to explain what this means. At the same time I shall also explain *bond uniqueness*, a concept introduced in paper I.

3.1 Disorder, not chaos

What is order? There is a good definition for something very ordered: it is periodic. When something is periodic, you know exactly what to expect next, that is order! If we are standing on a chess board and look down to read the color of our square, we know exactly what our neighboring squares will look like. If our square is black, the horizontal/vertical neighbors will be white and the diagonal neighbors black. This is true for all black squares of the board (ignore edge squares for now). Thus, the pattern really does not tell us anything, since we cannot know where on the board we are. The pattern of a chessboard contains no structural information, or at least, very little (it does tell us that it is black and white).

The opposite case to our chessboard would be a pattern were we find a maximum of structural information. In this case the pattern is completely chaotic, it could look like TV-noise [fig. 3.1(b)]. If we were standing somewhere on a completely chaotic pattern, chances are that we could figure out where we are. This spot is probably unique. For example we could be standing on a gray pixel in fig. 3.1(b), by looking around at the colors represented

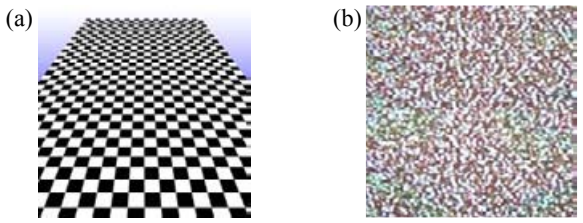


Figure 3.1: (a) Order is periodicity and isotropy. It is also boring and void of information. (b) Total disorder, chaos, is complete randomness. It is the carrier of maximum amount of information, alas, it is information void of meaning.

by our neighboring pixels, we could probably pinpoint our location in the pattern (provided we have a map). A completely chaotic pattern contains a high degree of structural information.

Luckily for science, we have a measure that is very good at discerning order form chaos, it is called *algorithmic information content**, and was introduced by three scientists, Solomonoff, Kolmogorov and Chaitin, independently of one another.[23, 24] Algorithmic information content (AIC) is conceptually easy to understand. To get the AIC of a pattern, just write the shortest possible computer program that can reproduce this pattern and count the number of bits that this program code contains. For example for a $n \times m$-squares chessboard pattern we could write the following simple program:

```
repeat(
     repeat(  black box; white box;  ) n/2 times;   new row
     repeat( white box; black box; ) n/2 times;   new row
) m/2 times;
```

Now, if we would like to reproduce the "chaos-pattern" of fig. 3.1(b) by writing a computer program that prints out the pattern, that would be a very lengthy piece of code. Clearly that program code must be much longer than the short one we used to reproduce the checkerboard pattern. Thus, a chaotic pattern has a larger algorithmic information content than a simple periodic pattern.

Up to now I have been referring to patterns like the one in fig. 3.1(b) as "chaotic", according to algorithmic information theory, such patterns are more appropriately called *random*. Before we had algorithmic information theory and the concept of algorithmic information content, there was no

*Also called program-size complexity or Kolmogorov complexity.

good way to tell whether a pattern was random or not. How can we tell the difference from a pattern that is completely random from one that looks random but in fact is not random at all but just generated from a short computer program in a very non-random way? We can do this by analyzing the AIC of the pattern. If the information in the pattern is not really random, we can compress it with some non-lossy image compression algorithm and represent the pattern by a shorter piece of code. However, if the pattern is truly random, no compression is possible and the information in the pattern can not be represented by anything shorter than the pattern itself. All of the above is valid in principle. In practice, there is still no way to see whether a finite pattern is truly random, and no way to calculate the absolute AIC, only relative to other patterns. But in principle, algorithmic information theory has given us a definition of total randomness (total chaos), and that is uncompressability, *i.e.* maximum information content.

The living world around us is full of information. Take a book for example, it is completely packed with it. The least one could call a book is periodic, sentences hardly ever repeat. Still the algorithmic information content of a book is rather low, much lower than a disordered pile of printouts of TV-noise. An aquarium filled with goldfish and growing seaweed is also rich in information, only very small traces of periodic order can be seen, like in the pattern of scales on some of the fish. For the most part, it is disorder. But it is not chaos. Algorithmic information theory tells us nothing about real, meaningful, information, just how periodic or non-periodic patterns are. The things we enjoy, structures we are built of and the fabric of life is not ordered/periodic neither is it chaotic/random, so what is it? People like to call it complexity.

3.2 What is complexity?

Complexity, according to many researchers (see [26] for an overview), can be viewed as something in between periodic order and complete randomness, see fig. 3.2. *Ordered complexity* and *emergent order* are terms that has been used to grasp this concept. In this thesis I will simply use, *complexity* and *complex* when discussing this intermediary between complete order and complete chaos.

A great effort has been put into finding a good definition of complexity and an accompanying measure of complexity, so far the issue is still largely an open problem. Worth noting here is the concept of *statistical complexity* introduced by Crutchfield and Young [27, 28]. It is based on the notion

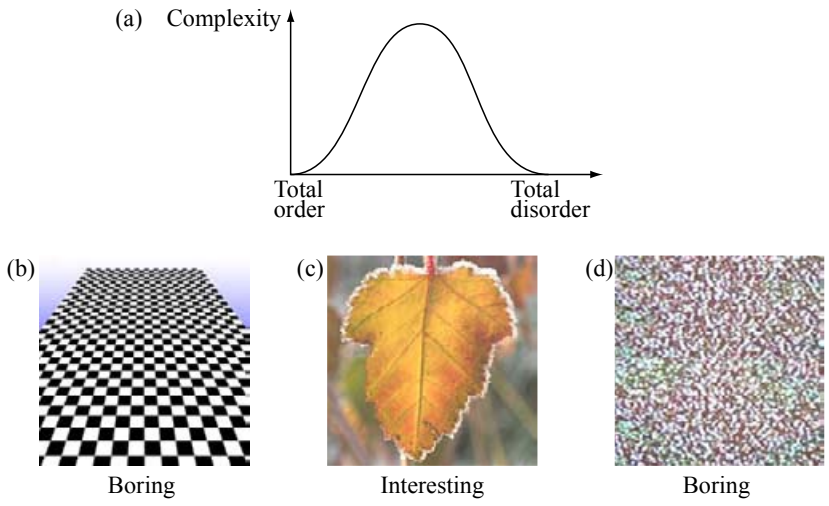


Figure 3.2: The common-sense concept of complexity. (a) Schematics illustrating what a measure of complexity should look like. (Freely adapted from [25]) (b) Order is lack of meaning, as is complete randomness (d). In between them both lies life and complexity filled with meaning and knowledge (d). Variatio delectat!

that *statistically* complete order and complete disorder are *easily* described, whereas true complexity is difficult to describe statistically. Other types of complexity measures take the whole history of the object into account, denoting objects as complex if it requires large amounts of computational work to produce them. Bennett s *logical depth* [29] and the *thermodynamic depht* introduced by Lloyd and Pagels [30] are the most well known examples of this type of complexity measures.

Common for all these notions of complexity, is the ability to note that the description of complete order or complete chaos is in fact *simple*. For an ordered pattern one just needs a few very simple rules. The same is true for a completely disordered pattern, one just needs a simple random number generator. Complexity is periodicity with variations, a book becomes interesting when the repetitive stream of similar letters are arranged to form aperiodic patterns, sentences. A Pentium circuit can do effective calculations only because the eternal field of transistors is now and then

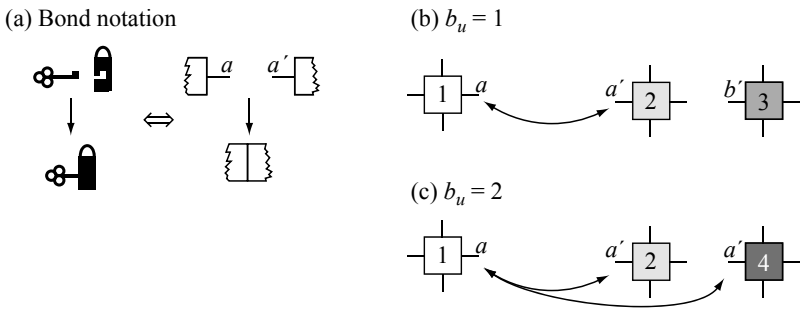


Figure 3.3: The bond uniqueness, b_u. (a) Instead of the lock-and-key notation used earlier. I will use letters like $a, b...$ to denote keys and $a', b'...$ to denote keyholes. a and a' are thus complementary and may form a bond a a'. (b) When the bond uniqueness is equal to one, only one building-block will be able to bind to a certain bond. In this case, building-block number 2 is the only one that fits bond a. (b) When the bond uniqueness is larger than one, a bond may harbor more than one type of building-block. In this case block 3 has been exchanged for block 4. Since 4 also has a a' bond, both building-block 2 and 4 will be able to bind to bond a. The b_u of bond a is thus 2.

interrupted by a wire or two. *Variatio delectat* (variety gives joy).

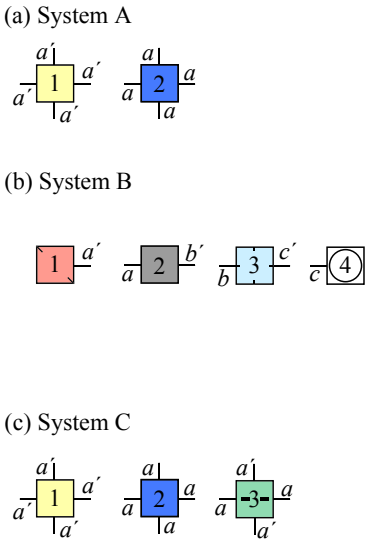
3.2.1 Making complex structures by self-assembly

Now, in order to produce functional nanodevices, we will need complex structures. Structures that are not periodic nor completely random, but a sort of intermediary between the two, following the discussion above. There are three principal ways of assembling a nanostructure. I will go through these classes by the aid of some new concepts: bond uniqueness and unique structures.

3.3 Introducing b_u, the bond uniqueness

The bond uniqueness, b_u, is a measure of how specific the bonds in an assembly system are. The higher the bond uniqueness, the less specific are the bonds. Looking at fig. 3.3(b) and (c), we find an example of the definition of bond uniqueness: when a bond may connect to a single type

The choice of the name *bond uniqueness* is unfortunate because of this. A better name might be *bond ambiguity*. However, since we use the term bond uniqueness in paper I, I will use that name here.



Bondtype	# possible blocks that may bind.
a	1
a'	1

Mean: b_u=1

Bondtype	# possible blocks
a	1
a'	1
b	1
b'	1
c	1
c'	1

Mean: b_u=1

Bondtype	# possible blocks
a	2
a'	2

Mean: b_u=2

Figure 3.4: The bond uniqueness of a *system* of building-blocks is calculated by taking the mean over the b_u s of all the bonds in the system.

of building-block, the b_u is equal to one (fig. 3.3(b)). When a bond may connect to two types of building-blocks, the b_u is equal to two (fig. 3.3(c)).

The b_u is defined for a system of building-blocks. Which building-blocks should be considered when determining the b_u? One should only consider the building blocks that participate in the assembly process in question, the building-blocks that are mixed in a particular experiment. Such a collection is called a *system* of building-blocks. If one for example mixes blocks 1, 2 and 3 the system will be the one displayed in fig. 3.3(b) and bond a will have bond uniqueness equal to one.

The bond uniqueness of a system of building-blocks is calculated by taking the bond uniqueness of all the bonds present in the system and taking the mean value over all the bonds. Take fig. 3.4(a) as an example. The bonds of type a can only connect to one type of building-block, the same is valid for the bonds of type a . Thus, we get that the mean bond uniqueness of the system is $b_u = 1$.

The systems of building-blocks shown in fig. 3.4 all produce different

When I write bond I do not mean a connection between two building-blocks in this case. For simplicity, the term bond in this thesis, will also be used to denote *sites* that provide bonding opportunities (*i.e.* a key , or keyhole , alone can be called a bond).

types of structures. When the building-blocks of system A are mixed they will produce a checkerboard pattern. The system B will produce finite tetramers when mixed and the system C can produce a large variety of different structures. What are the differences between these systems? Depending on how the bonds are distributed on the building-blocks they all belong to different classes of self-assembly: Crystal-type (syst. A), Unique addressing-type (syst. B) and Programmable (or Algorithmic) type self-assembly (syst. C). In the following sections I give descriptions of these classes.

Another interesting property with the concept of b_u is that it is in some way correlated to the structural complexity of the generated assemblies. More of this in chapter 6.

3.4 Assembly at $b_u = 1$

At $b_u = 1$ the assembly process is always completely determined by the bonds only. One bond one building-block. No ambiguity at all. This is an advantage, it makes the assembly process very robust against errors. However, this property also has a drawback: once the building-blocks are defined so are the resulting structures. If one wishes to modify the structure of the final assembly, one has no other choice but to redesign some or all of the building-blocks. There is no way of modifying the final structure by, say, mixing the building-blocks in a different order or some other experimental scheme. The structure is defined by the building-blocks only, and the building-blocks need to be modified if the resulting assembly is to be modified.

3.4.1 Crystal assembly

At $b_u = 1$ there are two types of assembly, the first type is the one in which a *unit cell* (a small collection of building-blocks) is reproduced over and over in a periodic pattern, this is called *crystal-type self-assembly*. The system in fig. 3.4(a) is such a system. It has $b_u = 1$ and the property that new building-blocks can always be added at the edges of the assembly. Crystal assembly systems always produce periodic patterns, see fig. 3.5.

3.4.2 Unique addressing assembly

The other possibility when $b_u = 1$ is an assembly where each building block has its unique position, or address, this is called *unique addressing-type self-*

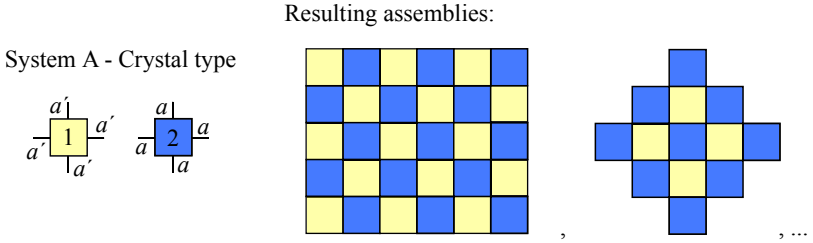


Figure 3.5: Assemblies resulting from system A in fig. 3.4(a). This is *crystal-assembly.*

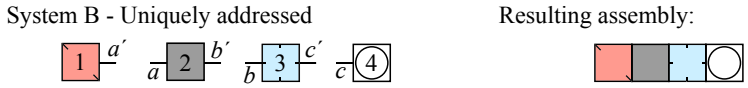


Figure 3.6: Assembly resulting from system B in fig. 3.4(b). Each building-block has its unique position, its unique address, within the resulting structure. This is *unique addressing*-type assembly.

assembly. Recall that in crystal assembly, new building-blocks can always be added at the edges of the assembly. If a system of $b_u = 1$ lacks this property, *i.e.* it creates closed edges, where no open bonds exists, it belongs to another class of self-assembly: *Unique addressing-type self-assembly.* Unique addressing-type assembly, always creates completely *aperiodic* patterns, so that one building-block only occurs once, see fig. 3.6.

3.5 Assembly at $b_u > 1$

This is an interesting option. Since b_u is larger than one, it means that the assembly process is no longer completely determined by the bonds alone. One bond does not always correspond to one building-block alone. In this way, the surrounding of one building-block is not always the same and the assembly can be made aperiodic. Unique-addressing assembly also produces aperiodic structures, however in that type of assembly each building-block only occur once in the finished structure. In so called *programmable self-assembly* , aperiodicity is possible even though the building-blocks occurs more than once in the finished assembly. In fig. 3.7 I show a few possible assemblies that may result when mixing the $b_u = 2$ system C from fig. 3.4(c). If all assemblies in fig. 3.7 are possible results from system C, how does one

In this thesis, the term *programmable* self-assembly has the same meaning as the term *algorithmic* self-assembly used by many authors[3, 31]

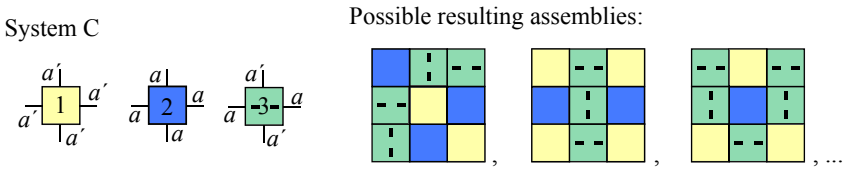


Figure 3.7: Assemblies resulting from system C in fig. 3.4(c). The patterns are aperiodic and building-blocks occur more than once. This is *programmable* self-assembly. Also called *algorithmic* self-assembly.

control which one we get? One of the points of doing nanoscale engineering by self-assembly is to have architectural control. Now I have presented a system that can assemble into a multitude of patterns, so how do we control the assembly?

3.5.1 Making assembly with $b_u > 1$ deterministic

The process of assembling at $b_u > 1$ can be made deterministic and controllable by introducing a constraint on the assembly process. This constraint is called cooperative bonding [3], (or ternary cyclic rules in [8]), and it can be described like this: Each building-block that assembles must make two bonds with two, already assembled building-blocks, in the assembly in order to attach, see fig 3.8. So a building-block that may only match one bond will not be attached, like the building-blocks 2 and 3 in fig. 3.8(a). This process can actually be implemented in a lab by setting the temperature at a value where objects that have made two bonds stick, and objects that have only made one bond shake loose, see fig. 3.9. Compare with fig. 2.2 on page 5.

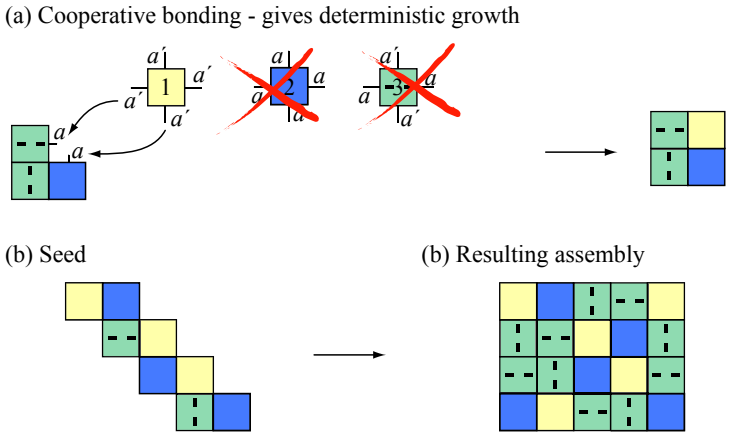


Figure 3.8: The way assembly at $b_u > 1$ can be made deterministic. If an incoming building-block is to attach to the growing structure it must bind with two bonds. (a) Only building-block 1 has two "a'"-type bonds next to one another and thus only building block 1 may bind to the assembly by making two bonds. The other possible choices, 2 and 3, may only make one bond. (b) By using a seed structure in combination with the cooperative bonding, the resulting assembly (c), is completely determined.

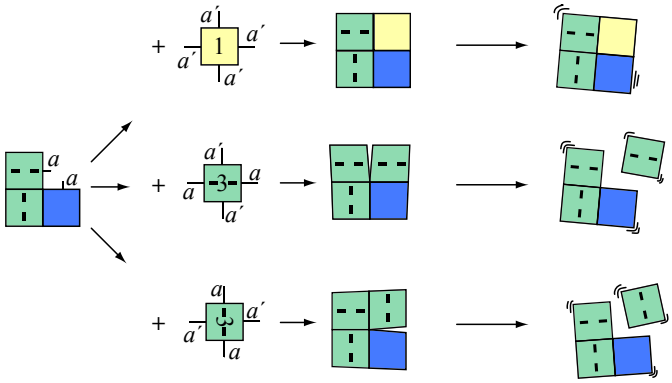


Figure 3.9: Cooperative bonding in practice can be achieved through temperature control. Here, an assembly has three options for growth. However, the temperature is too high to sustain assemblies containing singly bonded parts so these quickly fall off. The only sustainable structure is the one where two bonds are made (top).

3.6 Summary of Chapter 3

Algorithmic Information Content A measure of the degree of randomness. The measure is the number of bits in the shortest computer program that reproduces the structure. Also called Kolmogorov complexity or program-size complexity.

Statistical complexity A common name for complexity measures that look at the complexity of structural correlations. Periodicity and randomness are easy to describe statistically low statistical complexity. Complex structures are difficult to describe statistically high statistical complexity.

Complex structures Structures with a mix of disorder and periodic order. Have intermediary algorithmic information content, but the highest statistical complexity. Useful structures are always complex.

Bond uniqueness A measure that describes how deterministic a set of self-assembly building-blocks are. $b_u = 1$ means: one bond only one building-block will fit. $b_u > 1$ means: one-bond more than one building-block will fit.

Crystal-type self-assembly $b_u = 1$ Produces periodic assemblies

Unique addressing-type self-assembly $b_u = 1$ Non-periodic assemblies where each building-block only occurs once. Can produce complex structures but only by using a large number of building-block types.

Programmable self-assembly $b_u > 1$ Produces complex structures with few types of building-blocks. Also called *algorithmic self-assembly*.

Chapter 4

Building-blocks and possible assemblies

The properties of the building-blocks define the process of self-assembly to a large extent, although other factors, such as cooperative bonding, also have a great influence over the resulting assembly. If we assume that the assembly process is of the programmable (or algorithmic) self-assembly type, including the cooperative bonding effect, what types of building-blocks do we need to implement it in the lab? If we could make a certain system of building-blocks, what would the resulting assembly look like? In this chapter, I will try to summarize the results from paper I and some of the theoretical aspects in papers II and III that deal with these issues.

4.1 Will any kind of building-block be useful?

If the process is supposed to be of the programmable self-assembly type, giving complex structures in a controlled way, the short answer to this question is "no", the building-blocks need to have a certain minimal complexity.

Anisotropy. Assume that all the building blocks in an assembly system are completely symmetric, with only one type of bond* on all faces of each building block. In this system there exists no way of breaking symmetry and thus all the patterns will be completely symmetric and periodic. The

i.e. bonding *option*, see note on page 20.

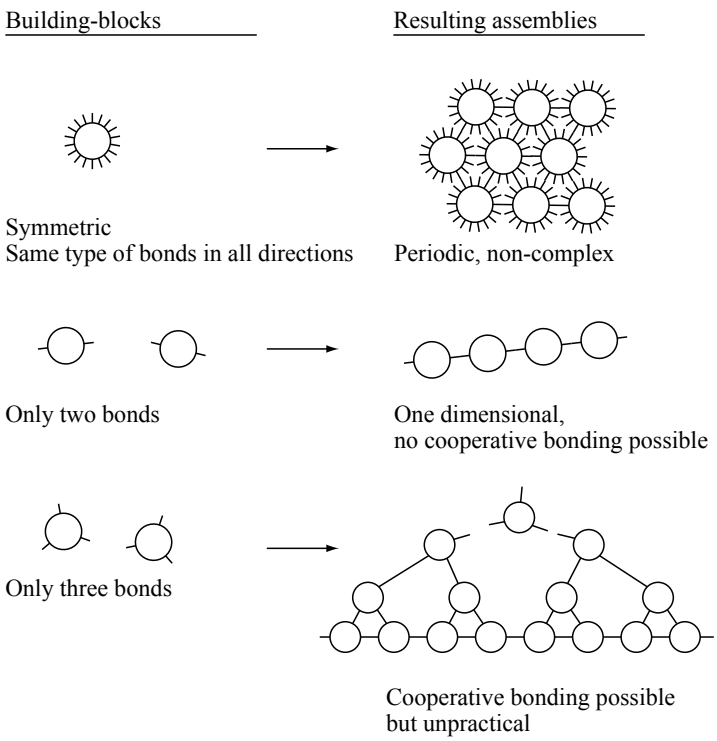


Figure 4.1: The reason why anisotropic building-blocks with at least four bonds are needed for programmable self-assembly. Note that when the building-blocks have three bonds, cooperative bonding is possible, but the assembly process quickly "dies" because the number of open bonds get reduced by one each time a building block attaches.

building-blocks must therefore be *anisotropic*, with the possibility of displaying different bonds in different directions.

At least four bonds. If the building-blocks do not have enough bonds, this will severely limit the types of assemblies obtainable. In particular, in order to create two-dimensional or three-dimensional assemblies with the use of cooperative bonding for programmable self-assembly, the building blocks need to have at least four bonds, see fig. 4.1. A more detailed account of the argument for this can be found in section 2 of paper II.

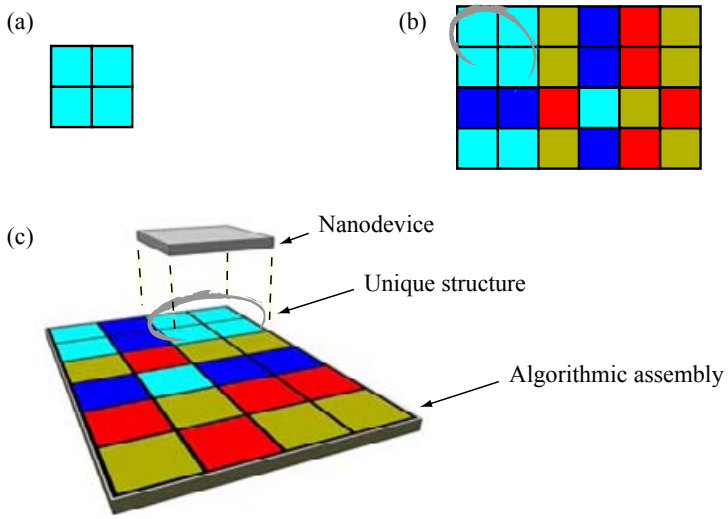


Figure 4.2: The patch of tiles (a) can be found in one place, and one place only, in the assembly (b). The encircled area in the assembly (b), thus constitutes a unique structure. (c) Practical importance of unique structures. The unique structures can be used as addresses in a two-step self-assembly process: First an assembly is made using programmable- (or algorithmic-) self-assembly and then nanodevices are assembled on top of this assembly, using the created unique structures as addresses. The right combination of "glues" represented by all the surfaces of the unique structure would be recognized by the surface of the nanodevice, making it assemble on the right place.

4.2 Unique structures of an assembly

I will now introduce the concept of a *unique structure*. A unique structure in an assembly is a patch of building-blocks that stick together is such a way that it is not repeated anywhere else in the assembly, like the patch of blueish tiles in fig. 4.2. *Tiles* is a term that is often used to denote square building-blocks for programmable self-assembly. [32, 31]

Besides being a good measure of the structural complexity of an assembly, there is also a technological reason why we should be interested in unique structures. The unique structures that are created in a programmable self-assembly process can be used to guide nanodevices to certain locations. Consider for example an experiment where a programmable self-assembly system is used to construct an aperiodic lattice of DNA. This lattice might then be used to position nanodevices on the unique structures, see fig. 4.2.

4.3 The number of unique structures obtainable

Consider an assembled structure made up of 24 small building-blocks. If we could find 6 unique structures in that assembly, each one consisting of 4 building-blocks, then we would say that the *number of unique structures*, denoted S, is equal to 6. The *size of the unique structures*, denoted η, would be 4 (*i.e. 4 tiles*) and the *total number of tiles*, denoted N, would be 24. If we say that unique structures may not overlap, then it will always be possible to find these numbers in an automatic way for any type of assembly. In fig. 4.3 I show how the numbers S and η are calculated in principle.

4.3.1 What use are S and η?

Armed with the concepts of unique structures and bond uniqueness, we can now make some statistical conclusions about the possibilities of different types of assembly systems. In paper I, I deduce this expression:

$$S = N \left(\frac{D \log b_u}{\log \left[\frac{b_u^D}{w} \left(N^{1/D} - (\frac{N}{S})^{1/D} + 1 \right)^D \right]} \right)^D \tag{4.1}$$

- S The maximum number of unique structures obtainable
- N Total number of building blocks used in the assembly
- b_u The bond uniqueness, introduced in chapter 3
- w The number of different building-block types involved in the assembly
- $D = 2$ when dealing with two-dimensional assembly and $= 3$ when dealing with three-dimensional assembly

Expression (4.1), equation (7) in paper I, is a relation that gives the maximum number of unique structures a certain assembly system may obtain. The assembly system being defined by the parameters N, w and b_u. In essence, this equation is a quantitative measure of the trade-off that exists between w and b_u on one hand and S and η on the other hand. Lets stop at this for a moment.

In programmable self-assembly there are two options for creating a large number of unique structures.

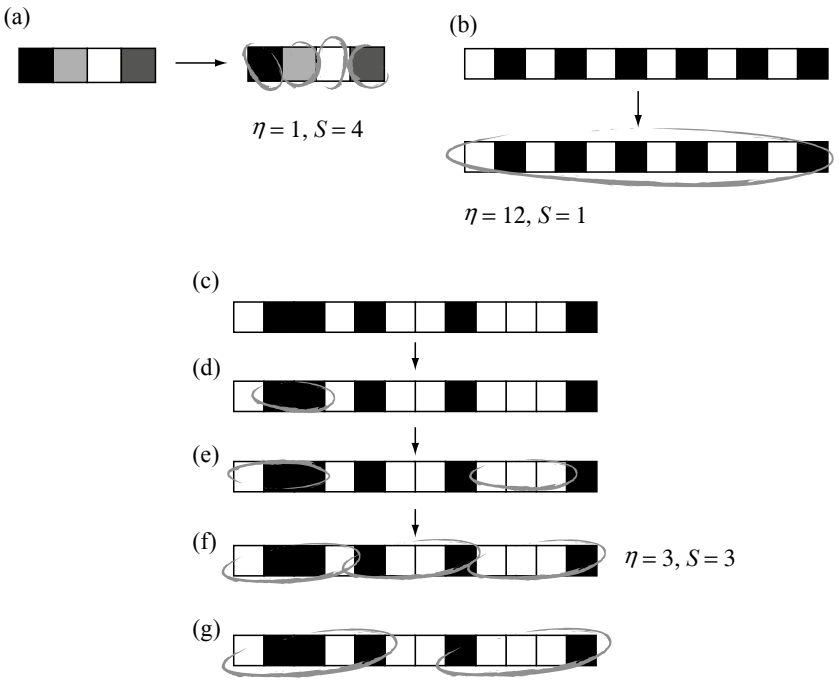


Figure 4.3: Principle for calculating the number of unique structures, S, and the corresponding size, η of the structures in an assembly. (a) If the assembly is totally aperiodic, *i.e.* uniquely addessed, each building-block is a unique structure. So in this case η is always one and the number of unique structures, S, is equal to the number of building-blocks in the assembly. If the assembly is periodic, (b), then there exists only one "unique" structure and that is the whole assembly (So the size of the unique structure η is equal to the number of building-blocks). If the assembly is of the programmable self-assembly type, the structure contains both periodicity and aperiodicity like the structure in (c). In this case the number of unique structures can be determined by increasing the η in steps and see how many unique structures one finds. In this case the maximum number of unique structures is 3 and that is found when $\eta = 4$. If η is increased to 5, the number of unique structures is reduced to 2 because of lack of space (unique structures may not overlap by definition).

1. Either one designs a system using very few types of building-blocks (low w and low b_u). This gives assemblies where the regularity is quite large and the resulting η is thus quite large, since a large number of building-blocks are needed to define a unique structure.

2. The other option is to design systems with a large number of building blocks (large w and large b_u), in this case the size of the unique structures can be made small, since the resulting lattice is more irregular.

By using expression (4.1) this trade-off can be estimated quantitatively.

Example (ripped from paper I) Consider an example where one wants to construct a scaffold for nanodevices. Lets say that we have the technology to produce 9 types of building blocks that each measures 13 nm×13 nm. They have b_u=3. How many uniquely addressable structures can we optimally produce in an area measuring 1 μm×1 μm. The total number of tiles we have room for is thus $N \approx 5900$. Using equation (4.1) with $D = 2$, and solving numerically, we get that the maximum number of unique structures is approximately 380 so $\eta = N/S \approx 15.5$. It should thus be possible to construct a scaffold comprising some 370 addressable sites, each containing 16 tiles. (If one uses $\eta = 15$, which is less than the estimated value, the assembly system will not be able to cover the entire area with unique structures, some will necessarily be repeated.) The unique structures / addressable locations are thus in the order of 50 nm×50 nm. If one wishes to make them smaller, more tile types are needed.

4.4 How to design an assembly system

Let us say we want to produce the system of building blocks in the previous example. Could the design of the building blocks be done in an automated way? The answer is yes. In section 4 of paper I, I show how this in principle can be made. The discussion is somewhat technical and I will not go into details here but merely summarize the most important concepts.

The design method uses the bond uniqueness, b_u, and the number of building-block types, w, as inputs. These numbers are used to calculate how many different types of bonds that must be used in the design. When the number of different bond-types have been determined, a list is made of all the possible combinations of two bonds. These combinations of two bonds are called *corners* since a corner of a building-block can be labeled

by its two adjacent bonds. From the list of corners, the building-blocks are assembled as combinations of corners.

To summarize, a complete assembly system design method could look something like this:

- *Define the assembly problem* by stating the desired number of unique structures that will be needed and their physical size, *i. e.* set the desired values of S and N.
- *Estimate the approximate b_u and w* by using equation (4.1).
- *Get the number of bond-types* by using the values of b_u and w as an input for the building-block design method
- *Building-blocks are designed* by "spreading out" the bond-types, as described in the method in paper I.
- *With the designed building-blocks, design the seed that gives the most unique structures by simulating the assembly.*
- *From simulations - get a map over the expected assembly and design the nanodevices to assemble on the correct unique structures.*

4.4.1 Feasibility of making this *in vitro*

The scheme outlined above is one way a programmable self-assembly technology could work in principle. In practice, however, assembly errors seem to be a major problem. In experiments with DNA-tiles, *i.e.* DNA building-blocks see sect. 5.3.1, the error rate is always much larger in programmable self-assembly[33], than in crystal type assembly[34] and unique adressing type assembly[11, 4]. There are methods to implement error-correction in programmable self-assembly[35], however, these methods rely on introducing more building-blocks and in principle they render the assembly more periodic, in order to keep it aperiodic at a larger scale. If one wants to keep the unique structures small, these error corrections methods are not so helpful. See discussion at the end of paper I.

Moreover, clear-cut examples of algorithmic, or programmable self-assembly occurring in nature are scarce, almost non-existing. Examples of unique-addressing (and crystal) type assembly however, are abundant in the living

Strictly speaking, ref. [11] (DNA-origami, see. sect. 5.3.1) is not about assembly of *tiles*, it is however an example of a unique-addressing type, DNA self-assembly.

Some viruses are believed to have a length controlling mechanism during assembly that can be called programmable self-assembly in the sense I have described. It is called

world. Is this an indication that a programmable self-assembly technology might be hard to implement in reality? Maybe unique addressing is the way to go?

the *vernier* mechanism. In vernier type assembly two proteins of different length attaches to one another and continues to attach as long as ends are protruding. When the length of the two proteins complexes match up the assembly stops, a simple type of prime-number modulo counter.

4.5 Summary of Chapter 4

Complex building-blocks Anisotropic building block with at least four bonds of arbitrary type directed in different directions. Needed for programmable self-assembly. Details described in paper II.

Tile Another word for building-block. The term is used extensively in the DNA self-assembly litterature and in paper I.

Unique structure An area of an assembly that is not repeated anywhere else in the assembly. The total number of such structures in an assembly is denoted by S in paper I.

Size of unique structure The number of building-blocks needed to define a unique structure. Denoted by η in paper I.

Number of tile types The number of different types of tiles (or building-blocks), used in an assembly process. Denoted by ω in paper I.

A trade-off relation between the number of tile types, ω, the size of a unique structure, η, and the total number of unique structures, S, is derived in paper I.

A method describing the design of an assembly system given the number of tile types, ω, and the bond uniqueness, b_u is introduced in paper I. Together with the trade-off relation described above, assembly systems that give a certain desired number of unique structures can be designed automatically.

The feasibility of a programmable (*i.e.* algorithmic) self-assembly technology is not certain. In order to produce small (= technologically important) unique structures in an assembly, a large number of building-block types will be needed. As a consequence, errors will be a problem. Moreover, complex self-assembly in nature seems to use unique-addressing and not programmable self-assembly.

Chapter 5

How to implement this in the laboratory

PRECEEDING CHAPTERS SHOULD HAVE provided the reader with the basic concepts of programmable self-assembly. In the following chapters I will introduce the ideas behind the experimental work that was performed as a part of this dissertation. Keeping it simple, I will try to avoid the technical details, since they can be found in the appended papers. All of the experimental work has involved the use of deoxy-nucleic acid, (DNA), either as a "glue" alone or as both a glue and a structural component. It is therefore appropriate to start off with a brief motivation for the use of DNA.

5.1 Why DNA?

DNA, the molecule of life! What has it got to do with nanotechnology? As it turns out, a great deal. By using DNA, we are able to control matter at a scale between chemistry and microtechnology. This is mainly due to the inherent programmability of the molecule itself and to our very thorough knowledge of its behavior and interactions. RNA is a similar molecule and it has been used for nanotechnology applications[10]. However DNA is chemically more stable and has more predictable folding properties than RNA.[14] Therefore, DNA remains the molecule of choice for many nanoscientists. Let us start the discussion with a brief introduction to the workings of the DNA molecule.

5.1.1 The structure and mechanisms of DNA binding

DNA is a *polymer* a large molecule consisting of many and similar smaller building-blocks, called *monomers*. The DNA monomers are called nucleotides. Each nucleotide has a phosphate group, a deoxyribose sugar and a nitrogenous *base*, see fig. 5.1(c) and (d). There are four variants of the bases in DNA: Adenine, Guanine, Cytosine and Thymine (or A, G, C and T for short). Each DNA polymer or DNA *strand* may have any sequence of these bases, for example AAAGACCT. The good thing about DNA from a self-assembly perspective is that base A stick to base T and base G stick to base C (and vice versa), through a weak chemical bond called hydrogen bond. So a strand like AAAGACCT will stick to the strand TTTCTGGA. In fact, the DNA-strands are directional and bind only to strands directed in the opposite direction, see fig. 5.1(c). It is customary to write a DNA sequence beginning with its 5 -end and ending with the 3 -end. So the strand (5 -)AAAGACCT(-3) will in fact bind to the strand (5 -)AGGTCTTT(-3). Strands like these two are called Watson-Crick complements.

Each individual bond in DNA is very weak, the G-C bond being slightly stronger than the A-T bond. The *melting temperature* of a double stranded DNA is the temperature where the double helix is separated into its two different single-stranded components due to thermal motion, the DNA is said to *denature*. When two single stranded DNA s meet and form a double-stranded molecule, the strands are said to *hybridize*. A longer double-stranded molecule has a higher melting temperature than a short one.

Short (less than 100 bases) DNA strands, so called oligonucleotides, are easy to manufacture in a dedicated DNA synthesiser machine. The raw materials are four bottles of A, G, C and T-nucleotides usually extracted from fish sperm. The sequences can be chosen at will, making it very easy to design experiments with DNA. Several suppliers provide custom oligonucleotides for approximately €1 per base, per μmole, and a test tube with the DNA can usually be delivered by mail a few days after one has made an order by writing down the desired A, C, G and T-sequence on their homepage.

5.1.2 So why use it?

The highly specific bonding-interactions makes DNA a perfect glue for self-assembly. Because the bonds are also relatively weak and reversible (a requisite for self-assembly, see fig. 2.2), the predictable interactions and the availability of well-known models for DNA-hybridization are also helpful.

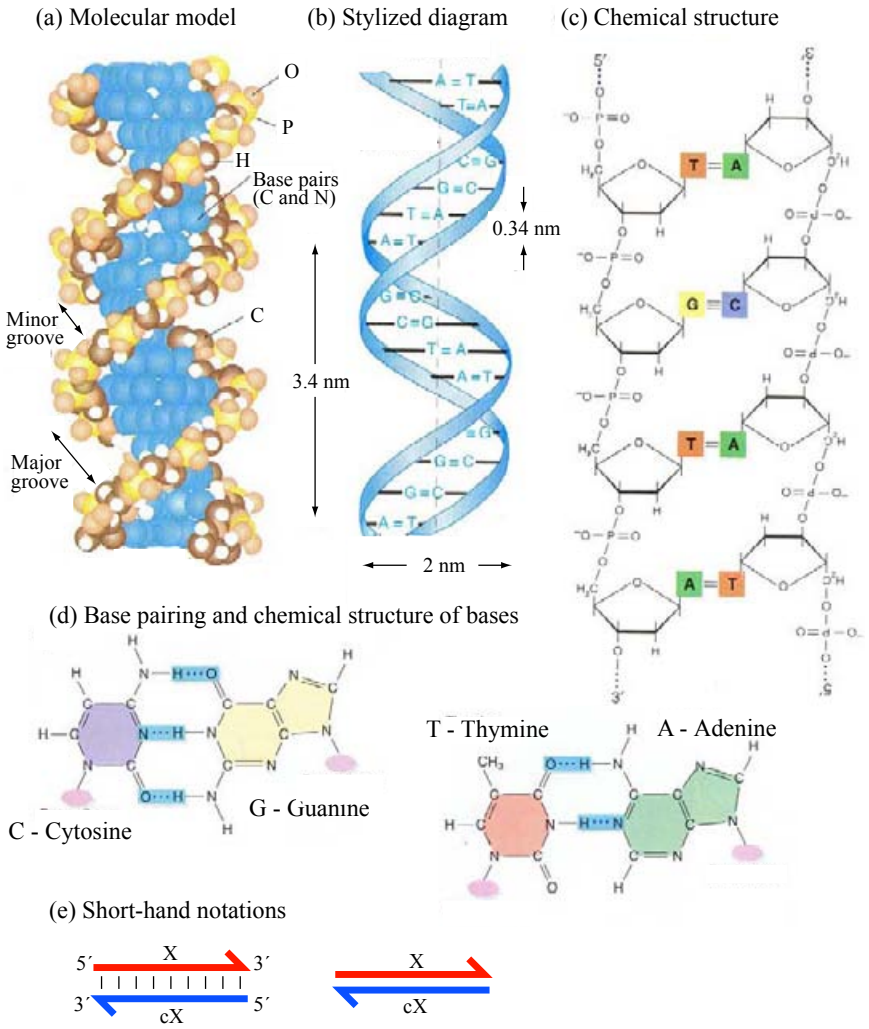


Figure 5.1: The structure of DNA with physical dimensions. (a) Space filling, three dimensional model of a double stranded DNA molecule. (b) Stylized diagram showing the base pairing principle. (c) Chemical structure of double stranded DNA. Note that the rightmost strand is vertically flipped compared tho the strand on the left. The structure of the bases themselves are found in (d) where the hydrogen bonds have been highlighted in blue. (e) Short-hand notation for interacting DNA-strands. The red strand, X, is the Watson-Crick complement to the blue strand, cX. The arrows point from the 5 -end to the 3 -end. ((a) through (d) adapted from [36])

Some question that often turn up at conferences and other scientific meetings is the following: "Why do you guys use DNA? Surely we must be able to come up with something better suited for nanofabrication. I mean, DNA is a genetic *information carrier*, not a building block. And also, isn t it dangerous to play with DNA?" I would like to clarify some points about DNA by addressing the issues raised one by one:

Why do you guys use DNA for self-assembly purposes?

Because it has highly specific, and predictable, bonding interactions. The bonds are weak and reversible. Moreover, DNA is small enough to make it useful for nanoscale applications. Finally, synthetic DNA is readily available for a few dollars to anyone with an internet-connection and a mailbox.

Surely we must be able to come up with something better suited for nanofabrication.

Yes, probably, someday. However, considering the large timescale nature has had to develop DNA, it is very likely that the knowledge needed to create something better than DNA will occupy many hundreds of graduate students for quite some time. We would like to start experimenting with programmable self-assembly *now*.

I mean, DNA is a genetic information carrier*, not a building block.*

Wrong. The genetic DNA is in fact a building block. Sure, it is a very information-rich building-block, but a building-block nevertheless. If you look at the processing of information in the developing cell, its all about attaching (and detaching) things to the DNA, that is, assembly.

And also, isn t it dangerous?

Nothing is ever completely risk-free of course, but the biohazard risk in using a lot of small fragments of synthetic DNA is almost vanishing. First of all, bacteria and viruses have very intricate genomes evolved over eons. The probability of us producing the DNA of an efficient killer virus by accident is astronmically

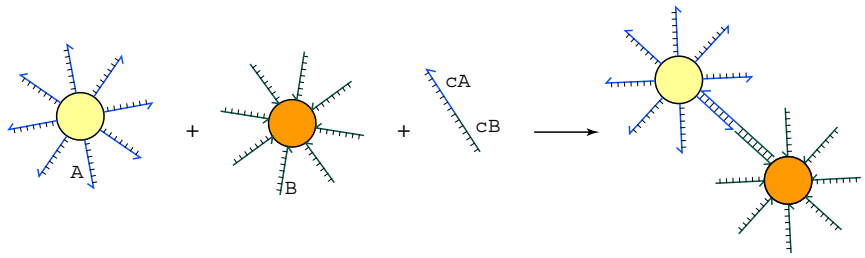


Figure 5.2: The principle of assembling nanoparticles with DNA. Two types of DNA-coated nanoparticles, coated with DNA `A` and `B` respectively are mixed with a *linker strand*, half of which is complementary to `A` and the other half complementary to `B`. The hybridization of the linker strand makes the particles stick together.

small. And secondly, even if we did, free DNA is very much exposed to degradation since all organisms on this planet (and many enzymes present even in air) efficiently destroy any DNA fragments that they encounter.

5.2 Nanoparticle self-assembly using DNA as glue

Despite its many advantages, DNA also has some real drawbacks. For our purposes the most obvious one has been the fact that DNA is an "electronically dead" material. If DNA self-assembly ever is going to be useful to make nano-electric circuits, the DNA will eventually have to be mixed with conducting materials and semiconductor materials (and maybe even superconductors) as active parts. That is why we introduced the concept of making programmable self-assembly with nanoparticles. This work is described in detail in papers II, III and IV.

5.2.1 The problem breaking the symmetry

DNA strands can easily be modified chemically with a number of different end-groups. In particular it is possible to attach a sulfhydrol-, or *thiol*-group (-SH) to either the 5 - or 3 -end of a DNA strand. If such thiol-modified DNA is mixed with gold particles, the thiol-group will adsorb to the gold surface, forming a covalent bond between the nanoparticle and the DNA strand.

In 1996, Mirkin *et al.* [37] and Alivisatos *et al.* [38] simultaneously came up with the idea of assembling gold nanoparticles that had been functionalized with strands of complementary DNA, see fig. 5.2. Both groups saw this as a method that could be used in the future to assemble nanostructures.

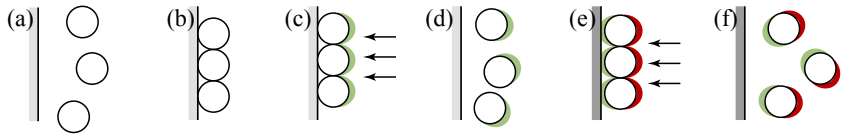


Figure 5.3: Functionalizing nanoparticles on two sides. Principle used in [39] and [40]. It is unclear how a similar method could provide more than *two* bond-types. Explanation of scheme principles: (a)-(b) Particles are attached to a surface. (c) The face not adjacent to the surface is exposed to functionalization. (d) Particles are released from first surface. (e) Particles are attached to another surface by using the first applied functionalization. (e) The other face of the particles is now subjected to functionalization no. 2 and finally released (f).

However, the problem with functionalized nanoparticles is that they are inherently very symmetric. This tend to make the assemblies non-complex and periodic. It is not until recently that people have started to realize that in order to make functional nanodevices we need to be able to assemble non-periodic, asymmetric structures (cf. chapter 3). As discussed in chapter 4, the building-blocks needed for programmable self-assembly must be quite complex. In particular, they need to be anisotropic with different bonds pointing in different directions. The problem is the following: How do we attach four (or more) types of DNA on four (or more) different faces of a nanoparticle with a diameter in the order of 10 nm?

One method might be to attach the particles to a surface and then functionalizing the free surface, reversing the process one could functionalize the other surface, see fig 5.3. For larger particles (1μm in diam.) Bao *et al.* have successfully applied *two* different patches of gold on opposite sides of the particles. [39]. In principle, similar techniques might be used for nanoparticles, and a recent experiment with nanoparticles [40] follows this line of ideas. These types of schemes for asymmetric functionalization suffers from an evident drawback: only *two* sides may be functionalized. As described earlier we need four or more bonds on each building-block.

5.2.2 Solution More than one particle per building-block

To the problem of making complex anisotropic building-blocks we introduce another solution. Instead of trying to fit four or more bonds on a single particle, fig. 5.4(a), we make a building block out of four or more nanoparticles each one carrying a specific DNA, see fig. 5.4(b). The trick is to break the symmetry by subsequent steps of linking and purification.

Figure 5.4: Instead of trying to fit four bonds on a single nanoparticle (a) we introduce a method to make a building block out of four different particles, (b), each one with its own DNA. The method can in principle be extended to make more advanced building-blocks, like a 3D building-block made up of eight nanoparticles.

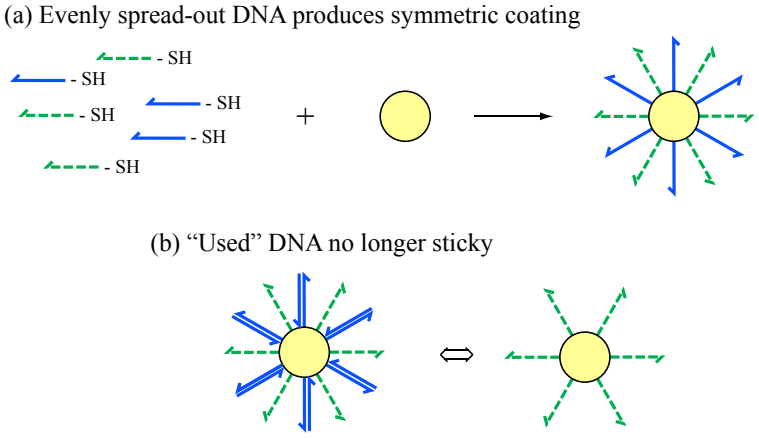


Figure 5.5: Prerequisites for the use of our method. DNA modified with a thiol group (-SH) will bind covalently to gold. (a) If two types of thiolated DNA are mixed with gold particles, a coating of DNA will be created on the nanoparticle surface that is a homogeneous mix of the two DNA-types. (b) If one of the DNA-types (here the solid one) is ”used up” by hybridization with its complement in some recation like linking, the unused DNA (the dashed one) will be the only active DNA left and the particle will be functionally equivalent to a particle with only one DNA. This property is used in the method to produce nanoparticle building-blocks (fig. 5.6).

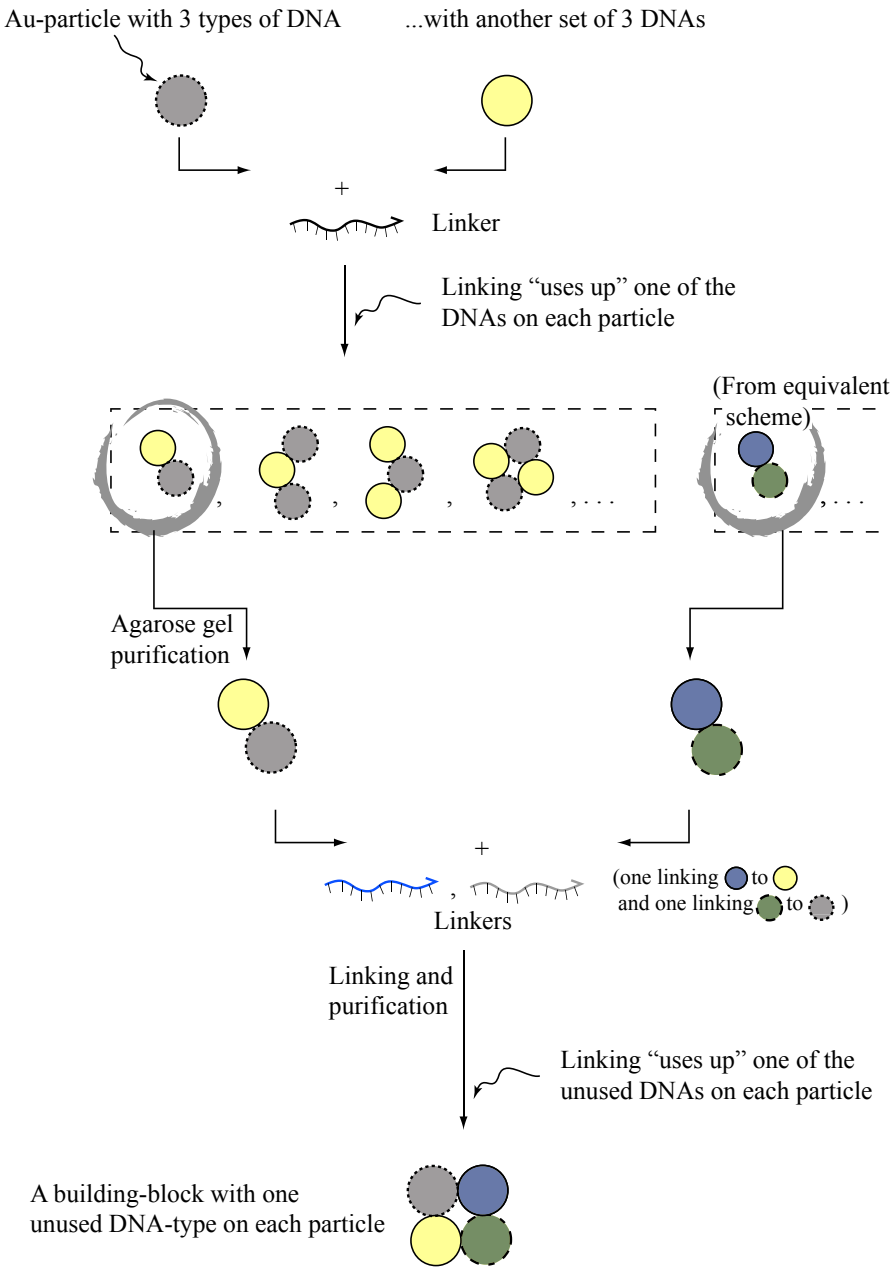


Figure 5.6: Scheme for the fabrication of anisotropic nanoparticle building-blocks for programmable self assembly. Starting from nanoparticles symmetrically coated with a mix of several DNA-types, an asymmetric building-block is achieved by consecutive linking and purifying.

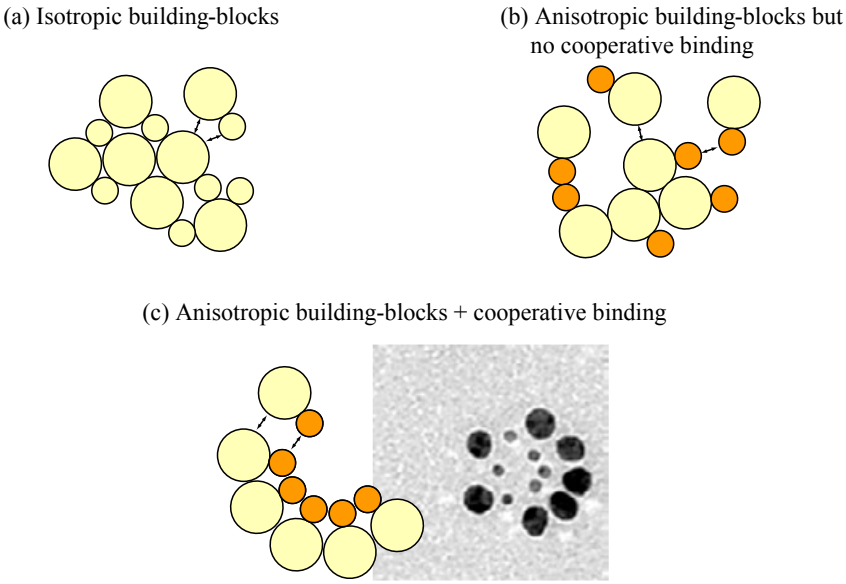


Figure 5.7: The reason why the results in paper IV show that the method for producing building-blocks (fig. 5.6) is working. (a) If the resulting building-blocks would have had the same DNA all over, then the resulting assembly would have been disordered. (b) If the resulting building-blocks would indeed have been isotropically functionalized, with two different types of DNA on each part of the dimer, but there was no cooperative binding effect, the results would have been hard to discern from the case (a). As it turns out, see (c), the results in paper IV (micrograph) are only explainable if our building-blocks are indeed anisotropic and that they assemble via cooperative binding.

The following points are important to understand our method. First, if two (or more) types of thiol-modified DNA are mixed with gold particles, both types of DNA will attach to the gold and spread-out evenly, see fig. 5.5(a). Second, note that, if the strands of one of several types of DNA on a nanoparticle are hybridized with their complement, that type of DNA is inactive ("used-up") and will no longer affect the assembly, see fig. 5.5(b).

The method to make anisotropic nanoparticle building-blocks is briefly introduced in fig. 5.6 and described in more detail in papers II and III.

5.2.3 Results Cooperative binding of anisotropic dimers

The method introduced above has been used to make dimer building-blocks. Production of dimers and purification by agarose-gel electrophoresis were shown to work in papers II and III. In paper IV I developed a slightly

modified scheme for DNA attachment and linking, so that only one type of dimer building-block could be used to prove that the dimers are actually anisotropically functionalized. To make the results easier to interpret the dimers were made from one large (20 nm diameter) and one small (10 nm) particle. The dimer building-blocks in paper IV are shown to assemble into curved structures, see fig. 5.7. These curved structures arise as a result of cooperative binding of the DNA s on both of the particles and because of the anisotropy of the dimer building-blocks. If the cooperative binding effect had not been present, and the dimers would not have had two types of DNA, each on one end of the dimers, the structures would not necessarily have turned out curved. Instead more random aggregates would have been found. Thus, the results in paper IV show that our method to produce programmable self-assembly building-blocks is feasible.

5.3 Using DNA-origami as a scaffold

As noted earlier, nanoparticles will probably have to be introduced in any DNA-self assembly technology that proposes to make nanoelectronics. Besides making the assembly directly with nanoparticles, as disscussed up until now, there is also a second possibility and that is using scaffolded assembly. By this I mean a process, where a pure DNA nanostructure is first assembled followed by the assembly of the nanoparticles onto this scaffold.

The work summarized in this section is presented in paper V. This paper is really a description of a work in progress. As will become evident, the attachment, and imaging, of gold nanoparticles on DNA seem to be a complicated issue.

5.3.1 DNA-nanostructures and DNA-origami

Over the last years, scientists have learned to "knit" with DNA.[12, 13] By mixing several strands of appropriately designed DNA strands, it is possible to obtain quite large periodic nanostructures built up of DNA alone, see fig. 5.8. These assemblies, or DNA crystals, can be as large as several tens of micrometers in diameter.[34] Winfree early recognized [3], that these *DNA-tiles* could be made into programmable self-assembly building-blocks and thus providing a path to complex nanostructures. *

*Quite often, tiles of other geometries than the cross-shape in fig. 5.8 are used in these experiments, but the basic principle is the same.

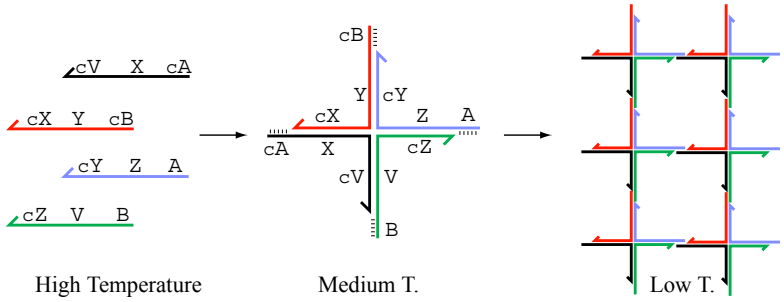


Figure 5.8: The principle of DNA-tile assembly. By designing strands that fit together in a cross-like pattern when hybridized, one gets a perfect self-assembly building-block, a *DNA-tile* (middle). As the temperature is decreased the sticky ends on the tiles (`cA, A, cB, B`) will make the tiles assemble into a large lattice. The technique can be varied to make programmable self-assembly building-blocks.

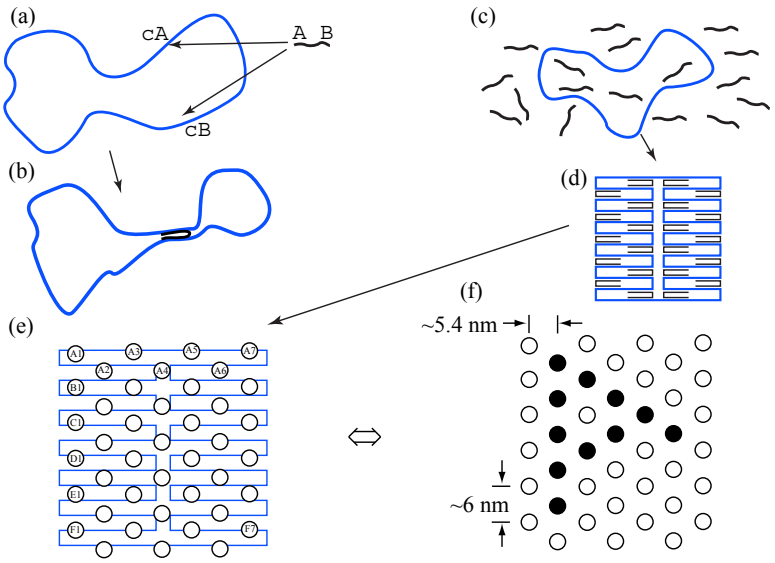


Figure 5.9: In the DNA origami method by Rothemund [11], a long circular DNA strand from a virus is folded by the help of a large number of small *staple*-strands. (a) If a short strand `A B`, is designed to hybridize to the large strand at two locations, half of it at `cA` and the other half at `cB`, then this short strand will "staple" the long strand like in (b). By designing a large number of such small strands and mixing them with the long strand (c) the staple strands will force the long strand to fold up into the desired structure (d). Because each staple-strand has its predefined position in the final structure (A1-F7 in (e)), the staples can be used as addresses to produce patterns like the letter "A" in (f) with a resolution of 5.4×6 nm.

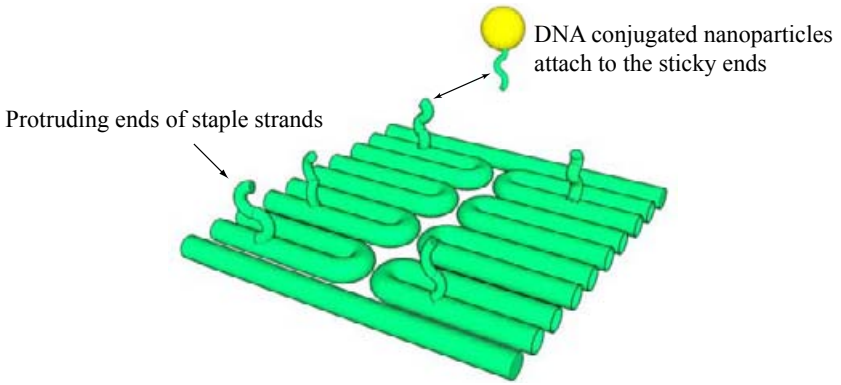


Figure 5.10: Using a DNA-origami as a scaffold for the assembly of nanoparticle patterns.

However, the fabrication of complex, aperiodic, DNA-nanostructures has been harder than expected and the produced assemblies are normally full of errors.[33]

Given the large error rates found in algorithmic DNA-assemblies, and our conclusion that such large error rates are hard to avoid if one wants to build complex assemblies with small unique structures (see sect. 4.4.1 and sect. 5 in paper I), the main goal of nanotechnology to produced complex, nanoscale patterns by self-assembly, remains largely unsolved. This was at least the case until march 2006 when Rothemund published a new method to organize DNA, called DNA origami.[11]. DNA origami is in principle a unique addressing type of self-assembly process, see page 21, each piece of DNA has only one place to go a unique address. The trick introduced by Rothemund (inspired by earlier work of Yan, LaBean, Feng and Reif [41]) was to add a scaffold to the assembly process. This scaffold, a long strand of DNA from a virus, was made to fold by the hybridization of a large number of small *staple* strands, see fig. 5.9.

Note that this type of assembly is similar to how the ribosomes in the cell are assembled, see sect. 2.2.1.

5.3.2 Results - The DNA-origami as a scaffold

In our lab we have reproduced the results of Rothemund, producing DNA-origami rectangles from DNA and virus genome acquired from other suppliers than the ones used by Rothemund. Since the viral genome we obtained probably originates from a slightly different clone, our reproduction of the

experiment is an indication that the method is robust. Our idea, also hinted at by Rothemund in the original paper, was to use the origamis as scaffolds for the assembly of nanoparticle patterns, see fig. 5.10. I have used 5 nm and 1.4 nm diameter gold particles functionalized with a DNA sequence that was complementary to a part of the prolonged staple sequence. However, this approach seems to suffer from some technical difficulties that have to be solved. Paper V describes the approaches I used to try to fit nanoparticles in patterns on the origamis. In summary, no patterns of nanoparticles could reproducibly be demonstrated by atomic force microscopy (AFM) in liquid or transmission electron microscopy (TEM). The main reasons for this are likely the following:

1. *The effective diameter of the nanoparticles is to large, due to electrostatics, to give the desired patterns.* The gold nanoparticles must be made slightly charged in order to avoid their aggregation in solutions containing salts. Obviously, this also prevent them from becoming "aggregated" in the form of a dense pattern on the origami. In previous work with nanoparticles on DNA-scaffolds, the particle to particle distance is normally not less than 20 nm.[6, 42]

2. *AFM imaging with particles around is a mess.* The DNA-origamis are imaged with AFM in liquid on a *mica* substrate. However, in samples containing particles the AFM imaging in liquid becomes difficult. Particles adhere to the AFM tip and the particles on the sample get a very large effective diameter due to interactions with the tip, further decreasing the resolution of the experiment. Dry AFM is not an option. When dried, samples become obscured and blurry, probably by salts and drying deformations of the DNA.

3. *TEM imaging is tricky because of drying effects and poor substrates.* In transmission electron microscopy, a drop of the sample is placed on a very thin film that is transparent to an electron beam. This film is usually made of carbon. The DNA-origamis tend to lye very flat on mica, but unfortunately they do not on carbon. A probable cause for this is that the carbon films are more or less hydrophobic. Moreover, the drying of the samples likely induces further structural artifacts. In combination these effects make the TEM imaging of DNA-origamis unreproducible producing structural images that do not ressemble the

Mica is a mineral that can be cleaved just before the experiment, revealing a fresh, atomically flat surface. In a buffer containing Mg^{2+} ions the DNA adheres strongly to the mica surface, making it a very good substrate for these kind of experiments.

ones obtained with AFM. It should be noted, however, that this work is in progress.

DNA-origami as a protein scaffold

Whereas the attachment of gold nanoparticles on an origami seems to be a technically difficult task, this is not the case for the attachment of proteins. In paper V I describe the use of staples modified with a biotin molecule to guide the assembly of a protein called streptavidin to specific sites on the origami. By using this technique I was able to produce nanoscale patterns of walking men, see fig. 3 in paper V.

5.4 Summary of Chapter 5

DNA is useful for nanotechnology applications because it is a small structure that has got highly programmable sites for interactions. It is also cheap and lab procedures for its reactions are well known.

DNA can be used in nanotechnology applications as a smart *glue* alone, or as both building-block and glue.

Symmetry needs to be broken when making nanoparticle building-blocks for programmable self-assembly.

A method to produce asymmetric programmable self-assembly building-blocks using DNA-functionalized nanoparticles, is introduced. The method produces building-blocks that are composed of several nanoparticles, having one *active* DNA type per particle.

Dimer building-blocks have been produced in the lab. Their assembly properties, self-assembling in curved structures, show that the method for building-block production works.

DNA-origami is a method where a large number of small DNA-strands (*staples*) are used to direct the folding of a long DNA-strand (*scaffold*). The long strand is a virus genome and the short ones are custom made artificial DNA-strands.

Nanoscale patterns can be made using DNA-origamis as a scaffold. Nanoscale patterns made from proteins are demonstrated. Creation of origami-scaffolded patterns made from gold nanoparticles is an open problem.

Chapter 6

Ideas for the future

In this chapter the focus will be on the future and I will present some thoughts on extending the research presented in this thesis. Many of the concepts presented here have not been published. The goal of this chapter is twofold. Firstly, I believe that this thesis will merit from a description of what one may expect to come out of continued research in the field. Secondly, this chapter is partly written for myself as a sort of mindmap of things that I should keep in mind if I get the opportunity to continue this line of research. The ideas presented here are speculative, some may be unfeasible, they are included here for the purpose of acting as a survey of the expectations of the field.

6.1 b_u as a complexity measure

The ideas presented here were first thought to be a part of paper I. As this paper grew larger, however, it was omitted due to lack of space. Furthermore, the practical importance of introducing a statistical complexity measure might be limited. This was also pointed out by one of the reviewers of an early version of paper I. Nevertheless, there is *something* substantial in what follows and I hope that, given the right critical input, the thoughts may evolve into a theory someday.

For a brief introduction to what *statistical complexity* is I refer the reader to section 3.2. Is there any practical use for a measure of true complexity? Most often the answer to that is: no. There is, however, something deeply philosophical about the notion of complexity. We all know what true complexity is when we see it, yet there is no general theory of complexity. Is that not annoying?

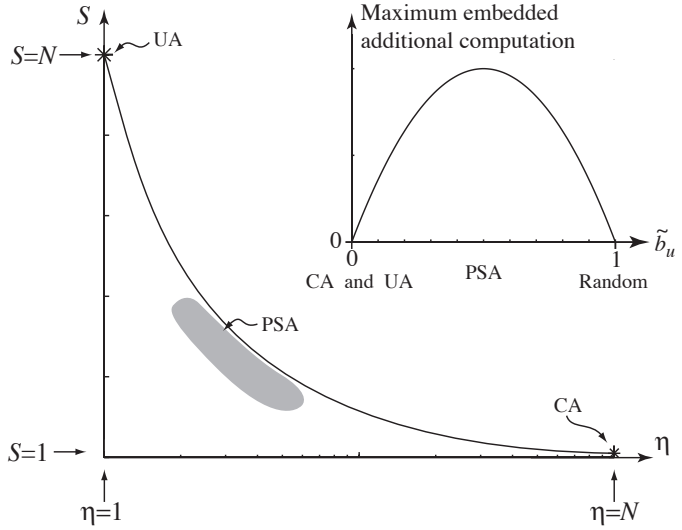


Figure 6.1: A plot of S (total number of unique structures) versus η (building blocks per unique structure) for different kind of assemblies at constant number of total building blocks, N. At the top left corner unique addressing (UA) is found, where each building block constitutes a unique structure. At the opposite corner crystal type assembly (CA) is found, where only one unique structure is possible (the whole assembly) because of the periodicity of the pattern. In the middle different kinds of PSA (Programmable Self-Assembly) are situated. The solid line represents the upper bound on $S \leq N/\eta$. The inset shows the qualitative dependence (eqn. (6.1)) of the amount of embedded additional computation in an assembly with respect to the reduced bond uniqueness b_u of the building blocks.

In this section I use the term *tile* to denote a basic assembly building-block.

6.1.1 $\tilde{b}_u$ and additional computation

The bond uniqueness parameter, b_u, is strongly influencing the structural complexity of the generated assembly patterns. In what follows I try to quantify this statement.

The *reduced bond uniqueness*, b_u is defined $b_u = (b_u - 1)/(w - 1)$, where w is the number of tile-types used in the assembly. b_u is $= 0$ when the assembly is crystal or uniquely addressed and $= 1$ when the assembly is random, *i.e.* any tile fits any bond. For PSA (programmable self-assembly) assemblies $0 < b_u < 1$.

Recall that deterministic complexity (introduced by Solomonoff, Kolmo-

gorov and Chaitin [23]) measures the amount of randomness, or information, in a structure whereas statistical complexity [27, 29] is based on the notion that both randomness and periodicity are equally simple to describe statistically. Objects are considered complex (having a high statistical complexity) if the amount of computation required to produce them is large.

Additional computation is defined as the amount of computation needed to correctly place a tile at a growth site, *in excess of bond matching computation*. To see if an open bond matches another open bond is fundamentally equivalent to performing c_b `AND` operations, where c_b is the bond complexity (number of bits necessary to describe the bond, cf. page 4). This computation must be performed in any assembly processes. The additional computation embedded in PSA processes is the ability to select the correct tile given more than one open bond (in principle, this is equivalent to performing nested `AND` operations). Here, I propose to use the additional computation as a statistical complexity measure. The amount of additional computation needed to select the appropriate tile type to attach (bind) to an assembly is proportional to the number of incorrect tiles that needs to be sorted out times their bond complexity.

The number of bonds an incoming tile must make in a PSA process is denoted τ (called the temperature parameter in [31])*. The number of tile types that the assembly process needs to take "into consideration" at each assembly step is the bond uniqueness times the τ-parameter, $b_u\tau$. At least τ out of these $b_u\tau$ tile types will be the same, namely the correct tile types that fit all bonds that need to be filled. The number of incorrect tile types the process must consider is thus $(b_u - 1)\tau$ or, using the reduced bond uniqueness, $b_u(w-1)\tau$. This value needs to be corrected for the fact that, as b_u increases, the fraction of incorrect tile types is decreasing. (When b_u reaches w all tile types will fit and there are no incorrect tile types to consider.) The fraction of "available" incorrect tile types is $1 - b_u$ so the maximum additional computation, C, is thus given by

$$C = c_b\tau(w-1)b_u(1-b_u) \tag{6.1}$$

where $w, \tau > 1$, $c_b \geq 1$ and $0 \leq b_u \leq 1$. The maximum additional computation, in other words the maximum structural complexity, is maximal around $b_u = \frac{1}{2}$ (PSA) and vanishing for $b_u = 0$ and $b_u = 1$ (CA-, UA- and random-type assemblies) see fig. 6.1 cf. fig. 3.2 on page 18.

*This is a measure of the degree of cooperativity needed to form the assembly. In the previous chapters, τ has been assumed to be equal to *two* for programmable self-assembly (*two* bonds needed to deterministically define an incoming tile).

Can this be applied to any kind of pattern? It might, the w (number of tile types) is simply the number of "colors" found in the pattern and principle b_u can be estimated from any pattern, assembly or not, by calculating the number of unique structures found and their size, and plugging that into equation (4.1) to calculate b_u. Could this be useful estimations of the statistical complexity of patterns by analyzing the unique structures? Maybe not as a practical tool. However, I m not sure that it is a completely useless insight, in terms of understanding complexity.

6.2 Drug delivery

Artificial nanostructures have been considered for theraputical use. In many cases they are considered safer to use than biological nanostructures, like those made from cloned viruses. Nanomachines or more passive nanodevices that deliver drug agents, proteins or gene-segments to cells are one of the goals for the emerging field of nanomedicine. [43, 44] Recent examples include, a passive nanoparticle device, made of copolymers, for the simultaneous delivery of DNA and drugs [45], and a more active, nanomachine-like, container based on a triblock copolymer vesicle [46]. In this section I propose a nanocontainer, built with DNA-coated nanoparticles, that could be used as a carrier for drug delivery. In addition to providing a vehicle for the drug transport, the containers I propose could be made to pop-open, releasing its contents, at a given chemical signal in the form of a DNA-molecule.

Gold nanoshells (silica particles with a thin layer of gold) have been proposed as a treatment for certain types of cancer, [47], and hollow gold nanoparticles as a vehicle for drug delivery [48]. Another approach that uses metallic particles is the use of magnetic iron-oxide nanoparticles to direct drugs to a site by magnetic forces. [49] In all, there seem to be no great concerns about the toxicity in the use of gold and iron *in vivo*. Note also that though most of our previous work has been on *gold*-nanoparticles, there are no reasons why the methods could not work with nanoparticles of a more biodegradable material, like for example polymer nanoparticles. So when I discuss *particles* in this section, I mean particles of any desired material.

6.2.1 Strand invasion

Strand invasion is the effect in which a hybridized DNA strand gets "invaded off" by another strand taking the place of the original strand. The effect is due to a random walk, induced by thermal motion, along hybridized base pairs, eventually leading to the effect that the strand that fits with the

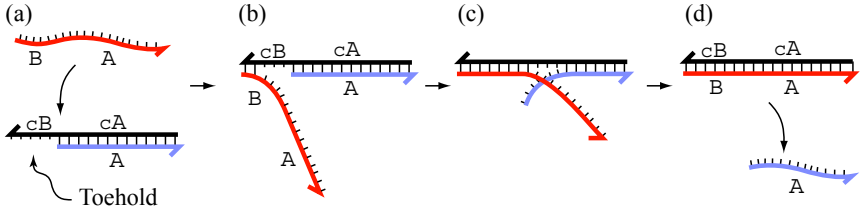


Figure 6.2: The phenomenon of strand invasion. In (a) the blue and the black strands are only partly complementary since a part of the black strand is still single stranded. This is called a *toehold*. If a strand that is complementary to the toehold as well as the rest of the black strand (the red strand) is added to the solution, it will bind to the toehold (b). By thermal motion the blue and the red strand will perform a random walk along the black strand (c) that eventually leads to the dissociation of the blue strand (d). The red strand is used to "unzip" the blue strand from the black one.

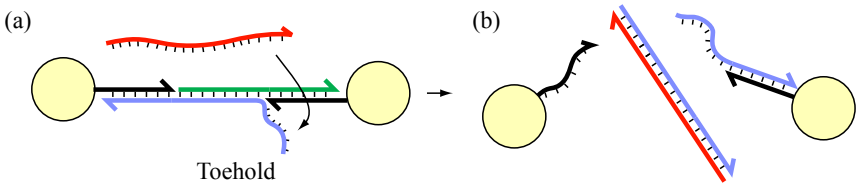


Figure 6.3: Using *strand invasion* to break up nanoparticle bonds by a given DNA-signal. (a) The link that keeps nanoparticles assembled can be set up with a toehold, such like that of the blue strand. Upon addition of the red DNA (*signal*) the blue strand will be invaded off (b) and the particles are no longer bonded.

highest number of bases will "win" over the strand that fits a smaller number of bases, see fig. 6.2

Since the phenomena of strand invasion can be used to detach strands of DNA from a hybridized assembly[50], without raising the temperature above the melting temperature, it could be used to break up an assembly of nanoparticles, see fig. 6.3

6.2.2 A container for drug delivery

The initial experiments with our anisotropic dimers, sect. 5.2.3 and paper IV, show that it should be possible to construct a hollow spherical vessel out of nanoparticles that are glued together with DNA. If such a vessel were prepared in a solution containing a high concentration of a drug agent, the interior of the vessel would be filled with that agent, fig. 6.4(a). Because of the strand invasion-effect, these vessels can be made to open up when

exposed to the right DNA signal, fig. 6.4(b).

Unused DNA of the outer nanoparticles could be used to bind to target sites that show complementary DNA. Also, it should be feasible to bind proteins to the outer particles, or even include the proteins in the assembly of the "drugball" like in fig. 6.4(c). Such proteins can act as antibodies, attatching at target sites on specific cells, fig. 6.4(d). After the attatchment of the drug carrier at its intended site, the drug can be released in a controlled way by exposing the drug carrier to the correct type of DNA, that is one that can open the container by strand invasion, fig. 6.4(e). The signal DNA could be injected, like in the case of gene therapy [51] or in a more futuristic scenario, be endogenous (*i.e.* DNA from the cell itself). One could imagine a case where the nanocontainer with its drug lie dormant in a cell, releasing its cargo only at a given signal in the form of a specific mRNA, for example, resulting in a kind of cell reprogramming.

6.3 A nano DNA-sequencer

Rapid sequencing of entire genomes is a killer-application for nanotechnology. To be able to screen genetic information in as simple a way as making a pH-test will be a revolution that will speed up medical and biological research and take our knowledge in these fields to unprecedented levels. The first draft of the human genome took several years to complete and cost about $300 million. Today the cost for sequencing a genome of human size is about to reach $100,000 and the quest for "the $1000 genome" is on. [52, 53] A wide variety of strategies exist to speed up the sequencing of DNA, from improvements on the original ideas in use today [54] to more speculative ideas, like the one I am proposing here, built on direct read-out on individual DNA-molecules [55].

The direct read-out methods, so called "revolutionary" sequencing techniques, are a long way from practical use, perhaps more than 10-15 years. Direct optical measurements on fluorecent-tagged DNA strands passing through a fluid channel [55], and motion based detection of a polymerase in action using optical tweezers [56] are two of these proposals. However, the conceptually simplest of these new sequencing techniques is *nanopore sequencing*. In nanopore sequencing a single stranded DNA molecule is made to pass linearly though a hole. This hole should have a diameter of no more than a few nanometers, prohibiting any other large molecules to pass through at the same time as the DNA molecule. One of the first demonstrations of this was the use of a biological pore, the membrane protein α-hemolysin, anchored in

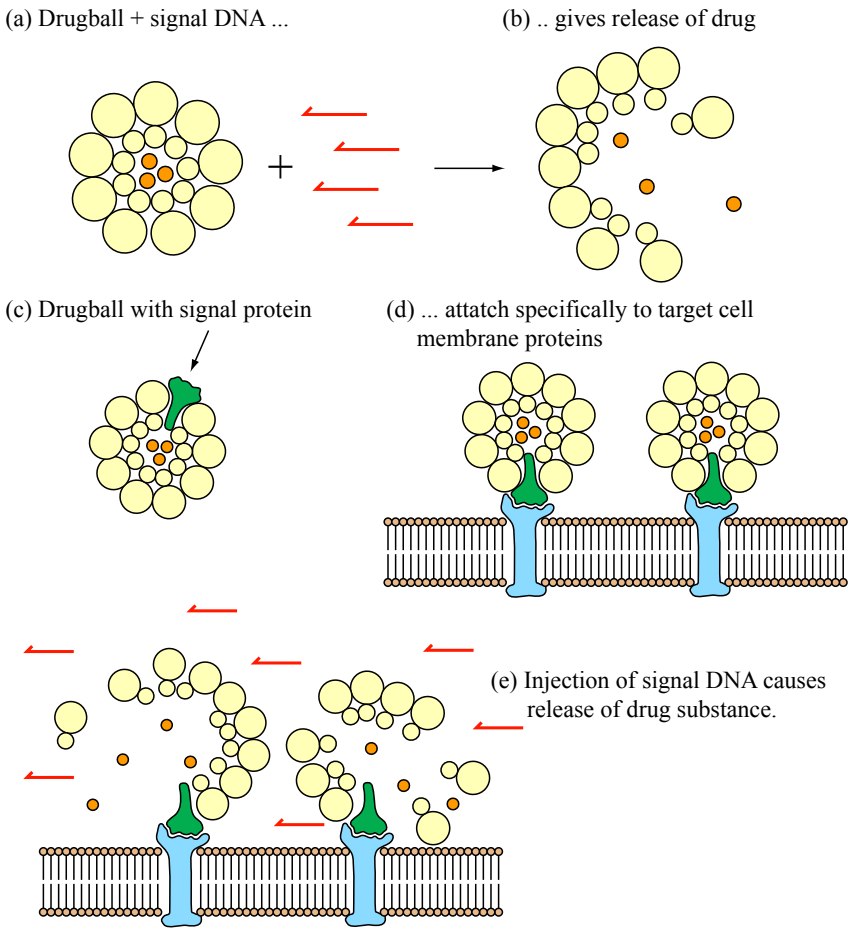


Figure 6.4: Drug delivery using nanoparticle assemblies. Strand invasion could be used to open up a spherical shell of nanoparticles that contains a drug substance (orange). (a) When a signal DNA comes into contact with the spherical shell, parts of the linkage get invaded off and the shell breaks up, (b), releasing the substance. If the spherical shell is assembled with a signal molecule, (c), it could target specific cells by selective attachment to membrane proteins, (d). If signal DNA is injected after the shells have located their target cells, a high local concentration of drug can be released at the target cell surface, (e).

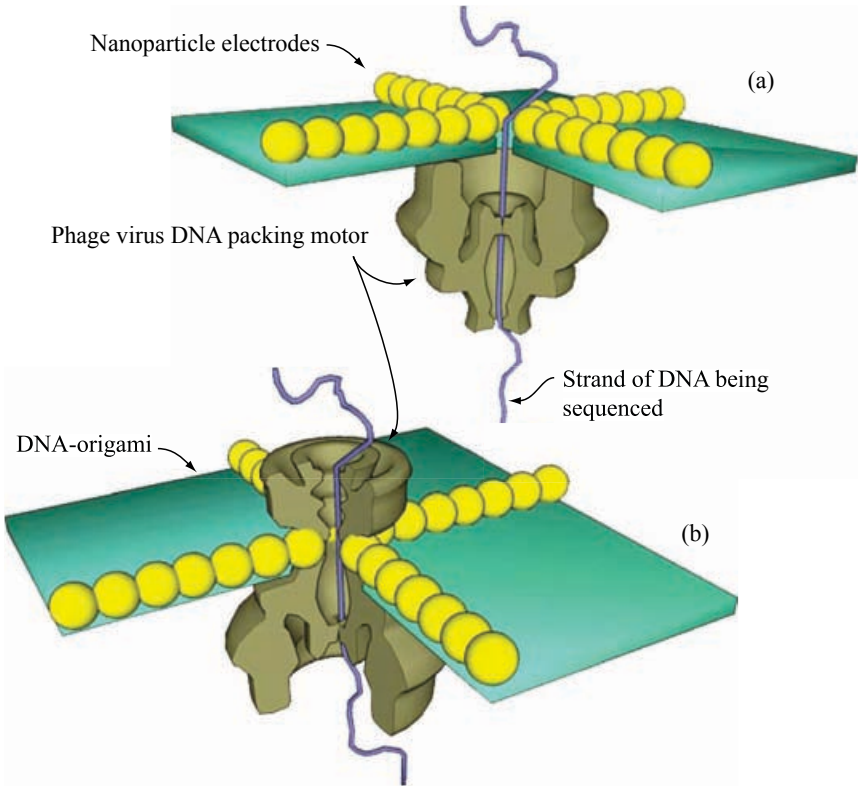


Figure 6.5: A DNA sequencing device made of DNA-origami scaffolded electrodes and a phage virus portal motor. In (a) a phage-portal is mounted on one side of the origami and the electrodes on the other. The origami must be prepared so that it has a small hole in the center. A better approach might be to incorporate part of the electrodes into the protein motor like in (b). This way, the DNA strand being measured is led passed the electrodes in a more stable configuration.

a lipid bi-layer. Using this setup, Kasianowicz *et al.* [57] were able to measure changes in the ion-current from one side of the membrane to the other, as DNA-molecules passed through. More recently, solid state nanopores, fabricated by focused ion-beam etching, have been used to reproduce this experiment. [58] It is unlikely that just measuring the ion-current blockade will provide the necessary resolution to measure individual base-pairs, however, the addition of measurement electrodes at the pore is estimated to yield single nucleotide resolution. [59]

6.3.1 A phage-portal motor in a DNA origami

In order to reproducibly obtain a sequencing device based on a pore with electrodes, I think it is necessary to use a bottom-up approach, where, in principle, direct control over atomic positions is possible. I therefore propose the following device: Take a protein assembly from nature with a suitable pore size, preferably a portal motor from a phage virus (see below), make it assemble with a DNA origami prepared with a hole in the center and with nanoparticle electrodes leading up to that hole, see fig. 6.5. Because of the addressability of DNA origamis, this idea is feasible.

To connect the nanoparticle electrodes to an external measurement equipment, the whole DNA origami assembly could be made to self-assemble on large hole in a solid state membrane prepared with gold connectors that are coated with DNA. Since this hole and the connectors could be made fairly large (∼50 nm) and with a high tolerance for manufacturing spread, these "holders" could be prepared fairly easily by a normal electron-beam lithography process.

Solid-state nanopores are more robust and insensitive to environmental changes than biological nanopores and therefore considered more suited for a sequencing technology. On the other hand, I would like to argue that the kind of atomic scale control that will be needed to reproducibly measure individual molecules can only realistically be achieved via a self-assembly technology. And the most obvious self-assembly technology would be to use DNA and proteins as the building-blocks. There exist in nature, a protein complex that would be perfect for this purpose: the DNA-packaging motor of some so called bacteriophage viruses. The structure and function of these protein machines have been studied by x-ray crystallography and cryo-electron microscope tomography. [60, 61, 62, 63, 64] Their purpose in nature is to aid in the virus assembly by packing the viral head (the *capsid*) full of the DNA virus genome, see fig. 6.6.

To my knowledge, the use of this viral motor has not been proposed as a tool for DNA-sequencing before. Why use a viral, DNA-packaging motor instead of a simple membrane protein like the aforementioned α-hemolysin?

1. *Control of translocation speed* The DNA strand moves through the pore by the aid of a molecular motor driven by ATP. By controling the concentration of ATP, the movement of the DNA strand can be controlled. By adding or removing ATP from the solution, a chemo-electric feedback circuit could be employed to tune the translocation speed while measuring.

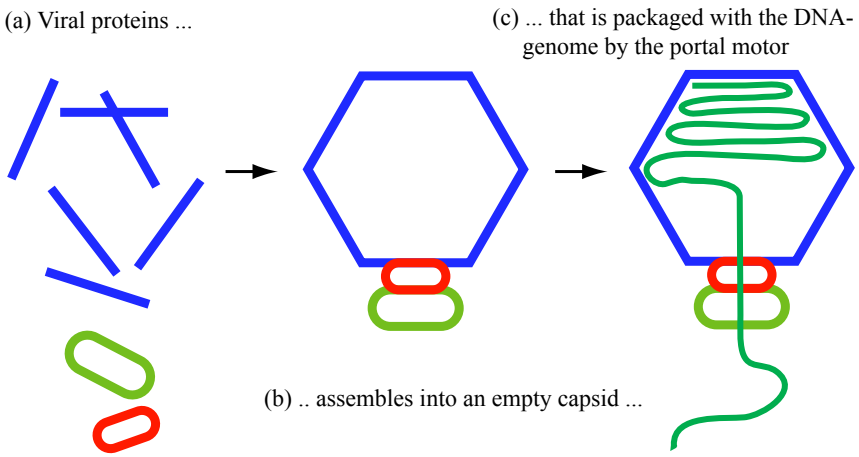


Figure 6.6: Simplified model of bacteriophage assembly. (a) Viral proteins produced from the viral genome by the host cell are self-assembled to form an empty capsule, or capsid, (b). In (c) the genome of the virus in form of double-stranded DNA is finally packed into the head by the phage portal motor, powered by adenosine triphosphate (ATP) hydrolysis.

2. *Slower translocation* In the present day schemes for pore-sequencing, the strands are moving too fast through the pores in order to make precise measurements on the nucleotides, and although schemes have been suggested to slow down the DNA translocation, they also lead to a lower ion-blockade signal.[65] With the use of a viral motor the translocation speed can be made as low as one wants.

3. *Reduction of structural fluctuations* Structural fluctuations of the DNA strand inside the nanopore are a likely source of noise, and the calculations by Lagerqvist *et al.* [59] supports this assumption. Solid-state nanopores and membrane pores like α-hemolysin do not provide the "steady grip" on the DNA strand that viral packaging motors must have in order to pack DNA into a virus capsid to a pressure that exceeds by ten times the pressure in a champagne bottle [61].

6.3.2 Problem? The known portal motors work with ds-DNA.

The bacteriophage portal motors pack double-stranded DNA (dsDNA) only. I have not found in the literature any description of a ssDNA (single-stranded DNA) virus that pack DNA in the same way as the dsDNA phages. The

ssDNA phages are more symmetric in shape and use a more self-assembly like process to fill with DNA: After capsid assembly, a protein binds to the ssDNA and helps it bind to the inner walls of the capsid.[66] In the nanopore sequencing literature, there seems to be a consensus that measurements must be done on single-stranded DNA (ssDNA). This might be a reasonable assumption since it is the code of a single strand that we seek. Knowing the seqence of A-T and G-C pairs in a dsDNA without knowing the individual strand sequences is useless for gene researchers. But is this really a problem? I would like to argue that by doing measurements on dsDNA, a much greater control can be achieved. Why are there no ssDNA portal motors in nature? Simply because the ssDNA is too floppy to be "pushed" into something. The flexibility and structural disorder of ssDNA will make a mess of any effort to make single nucleotide measurements in a pore. The dsDNA however, is a sturdy, rod-like structure with a ordered structure. Moreover, if the translocating DNA-strand is rotating as it passes the portal , then the alignment of the passing base-pairs would be fixed, relative to the pore-electrodes. In my opinion, this should greatly increase the signal-to-noise ratio of this device.

There exist in nature, several mechanisms to detect sequences from the outside of a dsDNA, and some of the mechanisms have even been suggested for DNA diagnostic use.[67] If the sequence of one of the strands in a dsDNA molecule can be *electronically* measured by nearby electrodes is an open problem and needs to be addressed by simulations. However, the fact that natural mechanisms do exist, gives a clear indication that such a sequencing scheme might be feasible.

It has been suggested that the phage portal motors themselves are in fact rotating to drive the DNA[60], in this case the bases would rotate as they pass the pore.

6.4 Summary of Chapter 6

The (reduced) bond uniqueness is related to complexity of an assembly or pattern. Can this be an important insight?

Additional computation is introduced as a measure of how much computational effort that needs to be applied to finish the assembly. *Additional*, because it is computation in addition to just matching bonds.

Drug delivery by a nanoparticle container. The results presented in this thesis might be used to make a small container for drug delivery.

The container can be made to pop open at any time by introducing DNA that dissolves the bonds by *strand invasion*.

Sequencing of DNA by measurements on single molecules passing through a pore might be possible. Especially if the strand movement can be controlled and electrodes placed at the pore.

A nano DNA-sequencer device is proposed. Instead of a simple pore, a portal motor from a virus is used. DNA-origami is used to hold the motor, and nanoparticle measurement-electrodes, in place.

Chapter 7

Concluding remarks

How can we control matter at the nanoscale in order to build useful nanodevices? In my opinion, this is the fundamental problem of nanotechnology, and it is still, to a large extent, unsolved. In the preceding thesis I have used this as the starting point. I have tried to guide the reader through the basics of this problem followed by some of my attempts to come up with solutions.

It is not a small problem. When we first ventured into the field of programmable self-assembly, we did not know much, we just thought that it should be a useful way to build nanoelectronics*. Håkan (*i.e.* Prof. Olin) had this idea that instead of "growing flowers from a seed" we should try to "grow Pentium chips". It seemed like a simple idea that just needed a little intellectual input to get going. We soon realized that self-assembly of complex structures is a huge problem. It encompasses physics, chemistry, computation, complexity, microbiology and even the problem of origin of life. While this sometimes got a bit discouraging, because my knowledge of these fields was, and unfortunately still is, limited, it is also why this subject is so much fun. While trying to construct a nanodevice to solve some down-to-earth problem, you soon end up wondering about up-in-the-sky problems like the notion of complexity. And the best thing is; its not about daydreaming! Fundamental issues really matter.

*And note that we did not plan to build bio-nanodevices at first, just electronics. The appreciation for the potential use for bio-applications came with time.

Acknowledgments

A NUMBER OF PEOPLE have contributed to this thesis, both directly and indirectly. First of all I would like to thank my father and mother for giving me a childhood that has taught me to appreciate science and the merits of critical, independent thinking. Growing up in a home where one parent fiddles with brains and the other parent fiddles with atoms, all while discussing world problems, is an excellent starting point for any prospective scientist. My father, Hans-Erik Hogberg, has also been of direct help for this thesis, many tips and helpful insights about everything "chemical" has been very appreciated, as well as the very thorough proof-reading of this manuscript.

Further, I would like to express my deep gratitude to my advisor Håkan Olin. He had the courage to employ me after a brief meeting three years ago when this story began. He was also courageous enough to push me into the field of programmable self-assembly, and for this I am very grateful. This field had almost no active people in Sweden and very few in Europe at the time. He has been a great support, always encouraging, and a great source of ideas and thoughts.

I am grateful for the discussions I have had with my fellow PhD student Magnus Hummelgård who gave many useful tips and prophecies, and for the help with simulations by Martin Olsen. Many thanks goes to Lars Ryen for swift TEM-imaging when needed. I also want to thank Goran Andersson who has been very helpful during my teaching attempts and greatly reduced my workload.

For the help in the lab from Svante Holm, Håkan Norberg and Torborg Jonsson I am very thankful. Without your help, my transition into a molecular biology lab-worker would never have occurred.

I would like to thank my father-in-law, Lars-Åke Lindstrom, for careful proofreading and commenting on this manuscript.

The people at the department have all been very helpful, Lotten Glans, Peter Glans, Nils Olander, Christine Malmgren, Sam Lodin, Tomas Nilsson

and Frank Wikstrm among others. I have appreciated the discussions around the coffee table, like the ever ongoing one about the nature of mathematics and computation.

Finally, I would like to thank my family for all the support, Åsa Lindstrom, Joel Hogberg and Ragnar Hogberg. You are the most complex objects in the universe and I find new reasons to love you every day.

Bibliography

[1] N. C. Seeman. Nucleic-acid junctions and lattices. *Journal of Theoretical Biology*, 99(2):237 247, 1982.

[2] L.M. Adleman. Molecular computation of solutions to combinatorial problems. *Science*, 266(5187):1021 3, 1994.

[3] E. Winfree. On the computational power of DNA annealing and ligation. In R. J Lipton and E. B. Baum, editors, *DNA Based Computers*, volume 27 of *DIMACS series in Discrete Math. and Theor. Comp. Sci.*, pages 199 221. Amer. Math. Soc., Princeton, 1996.

[4] Sung Ha Park, Constantin Pistol, Sang Jung Ahn, John H. Reif, Alvin R. Lebeck, Chris Dwyer, and Thomas H. LaBean. Finite-size, fully addressable DNA tile lattices formed by hierarchical assembly procedures. *Angewandte Chemie International Edition*, 45(5):735 739, 2006.

[5] Hao Yan, Sung Ha Park, Gleb Finkelstein, John H. Reif, and Thomas H. LaBean. DNA-templated self-assembly of protein arrays and highly conductive nanowires. *Science*, 301(5641):1882 1884, 2003.

[6] J. P. Zhang, Y. Liu, Y. G. Ke, and H. Yan. Periodic square-like gold nanoparticle arrays templated by self-assembled 2D DNA nanogrids on a surface. *Nano Letters*, 6(2):248 251, 2006.

[7] M.M. Shih, J.D. Quispe, and G.F. Joyce. A 1.7-kilobase single-stranded DNA that folds into a nanoscale octahedron. *Nature*, 427(6975):618, 2004.

[8] E. Klavins, R. Ghrist, and D. Lipsky. Graph grammars for self assembling robotic systems. *IEEE Proc. Int. Conf. Robotics and Automation*, 5:5293 5300, 2004.

[9] R. P. Goodman, I. A. T. Schaap, C. F. Tardin, C. M. Erben, R. M. Berry, C. F. Schmidt, and A. J. Turberfield. Rapid chiral assembly of rigid DNA building blocks for molecular nanofabrication. *Science*, 310(5754):1661 1665, 2005.

[10] A. Chworos, I. Severcan, A. Y. Koyfman, P. Weinkam, E. Oroudjev, H. G. Hansma, and L. Jaeger. Building programmable jigsaw puzzles with RNA. *Science*, 306(5704):2068 2072, 2004.

[11] P. W. K. Rothemund. Folding DNA to create nanoscale shapes and patterns. *Nature*, 440(7082):297 302, 2006.

[12] N. C. Seeman and P. S. Lukeman. Nucleic acid nanostructures: bottom-up control of geometry on the nanoscale. *Reports on Progress in Physics*, 68(1):237 270, 2005.

[13] K. V. Gothelf and T. H. LaBean. DNA-programmed assembly of nanostructures. *Organic & Biomolecular Chemistry*, 3(22):4023 4037, 2005.

[14] L. Jaeger and A. Chworos. The architectonics of programmable RNA and DNA nanostructures. *Current Opinion in Structural Biology*, 16(4):531 543, 2006.

[15] C. M. Niemeyer. Nanoparticles, proteins, and nucleic acids: Biotechnology meets materials science. *Angewandte Chemie-International Edition*, 40(22):4128 4158, 2001.

[16] G. Cziko. *Without Miracles*. The MIT Press, Cambridge, Massachusetts, 1995.

[17] G. M. Whitesides and B. Grzybowski. Self-assembly at all scales. *Science*, 295(5564):2418 2421, 2002.

[18] P. W. K. Rothemund. *Theory and Experiments in Algorithmic Self-Assembly, Ph.D Thesis*. PhD thesis, Univ. of Southern California, 2001.

[19] whose writings include this influential article: Chris G. Langton. Computation at the edge of chaos - phase-transitions and emergent computation. *Physica D*, 42(1-3):12 37, 1990.

[20] G. M. Culver. Assembly of the 30S ribosomal subunit. *Biopolymers*, 68(2):234 249, 2003.

[21] BIODIDAC digital resource. http://biodidac.bio.uottawa.ca/.

[22] H. Fraenkel-Conrat. The role of the nucleic acid in the reconstitution of active tobacco mosaic virus. *J. Amer. Chem. Soc.*, 78:882 3, 1956.

[23] R.J. Solomonoff. A formal theory of inductive inference. *Inform. Control*, 7:224 254, 1964.

[24] M. Li and P. Vitányi. *An Introduction to Kolmogorov Complexity and Its Applications*. Springer-Verlag, 1997.

[25] J. P. Crutchfield and K. Young. Computation at the onset of chaos. In W. Zurek, editor, *Entropy, Complexity, and the Physics of Information*, volume 7 of *SFI Studies in the Science of Complexity*. Addison-Wesley, Reading, 1990.

[26] Chapter 13 in M. Lockwood. *The Labyrinth of Time*. Oxford University Press, Oxford, 2005.

[27] J. P. Crutchfield. The calculi of emergence: Computation, dynamics and induction. *Physica D*, 75(1-3):11 54, 1994.

[28] J. P. Crutchfield and K. Young. Inferring statistical complexity. *Physical Review Letters*, 63(2):105 108, 1989.

[29] C. H. Bennett. Logical depth and physical complexity. In Rolf Herken, editor, *The Universal Turing Machine a Half-Century Survey*, pages 227 257. Oxford University Press, Oxford, 1988.

[30] S. Lloyd and H. Pagels. Complexity as thermodynamic depth. *Annals of Physics*, 188(1):186 213, 1988.

[31] P.W.K. Rothemund and E. Winfree. The program-size complexity of self-assembled squares. In *Proceedings of 32nd Annual ACM Symposium on Theory of Computing*, pages 459 68, New York, NY and Portland, OR, USA, 2000. ACM.

[32] H Wang. Proving theorems by pattern recognition ii. *Bell. Sys. Tech. J.*, 40(1), 1961.

[33] P. W. K. Rothemund, N. Papadakis, and E. Winfree. Algorithmic self-assembly of DNA sierpinski triangles. *Plos Biology*, 2(12):2041 2053, 2004.

[34] Y. He and C. D. Mao. Balancing flexibility and stress in DNA nanostructures. *Chemical Communications*, (9):968 969, 2006.

[35] Erik Winfree and Renat Bekbolatov. Proofreading tile sets: Error correction for algorithmic self-assembly. *LNCS, DNA Computers 9*, 2943:126 144, 2004.

[36] P.J. Russel. *iGenetics: A Molecular Approach*. Benjamin Cummings, San Francisco, 2006.

[37] C. A. Mirkin, R. L. Letsinger, R. C. Mucic, and J. J. Storhoff. A DNA-based method for rationally assembling nanoparticles into macroscopic materials. *Nature*, 382(6592):607 609, 1996.

[38] A. P. Alivisatos, K. P. Johnsson, X. G. Peng, T. E. Wilson, C. J. Loweth, M. P. Bruchez, and P. G. Schultz. Organization of nanocrystal molecules using DNA. *Nature*, 382(6592):609 611, 1996.

[39] Z. Bao, L. Chen, M. Weldon, E. Chandross, O. Cherniavskaya, Y. Dai, and J.B.-H. Tok. Toward controllable self-assembly of microstructures: selective functionalization and fabrication of patterned spheres. *Chemistry of Materials*, 14(1):24, 2002.

[40] F. W. Huo, A. K. R. Lytton-Jean, and C. A. Mirkin. Asymmetric functionalization of nanoparticles based on thermally addressable DNA interconnects. *Advanced Materials*, 18(17):2304, 2006.

[41] H. Yan, T. H. LaBean, L. P. Feng, and J. H. Reif. Directed nucleation assembly of DNA tile complexes for barcode-patterned lattices. *Proceedings of the National Academy of Sciences of the United States of America*, 100(14):8103 8108, 2003.

[42] J. D. Le, Y. Pinto, N. C. Seeman, K. Musier-Forsyth, T. A. Taton, and R. A. Kiehl. DNA-templated self-assembly of metallic nanocomponent arrays on a surface. *Nano Letters*, 4(12):2343 2347, 2004.

[43] S. M. Moghimi, A. C. Hunter, and J. C. Murray. Nanomedicine: current status and future prospects. *Faseb Journal*, 19(3):311 330, 2005.

[44] G. A. Hughes. Nanostructure-mediated drug delivery. *Dm Disease-a-Month*, 51(6):342 361, 2005.

[45] Y. Wang, S. J. Gao, W. H. Ye, H. S. Yoon, and Y. Y. Yang. Co-delivery of drugs and DNA from cationic core-shell nanoparticles self-assembled from a biodegradable copolymer. *Nature Materials*, 5(10):791 796, 2006.

[46] P. Broz, S. Driamov, J. Ziegler, N. Ben-Haim, S. Marsch, W. Meier, and P. Hunziker. Toward intelligent nanosize bioreactors: A pH-switchable, channel-equipped, functional polymer nanocontainer. *Nano Letters*, 6(10):2349 2353, 2006.

[47] L. R. Hirsch, A. M. Gobin, A. R. Lowery, F. Tam, R. A. Drezek, N. J. Halas, and J. L. West. Metal nanoshells. *Annals of Biomedical Engineering*, 34(1):15 22, 2006.

[48] Y. G. Sun, B. Mayers, and Y. N. Xia. Metal nanostructures with hollow interiors. *Advanced Materials*, 15(7-8):641 646, 2003.

[49] P. Gould. Nanomagnetism shows *in vivo* potential. *Nano Today*, 1(4):34 39, 2006.

[50] B. Yurke, A. J. Turberfield, A. P. Mills, F. C. Simmel, and J. L. Neumann. A DNA-fuelled molecular machine made of DNA. *Nature*, 406(6796):605 608, 2000.

[51] H. Herweijer and J. A. Wolff. Progress and prospects: naked DNA gene transfer and therapy. *Gene Therapy*, 10(6):453 458, 2003.

[52] R. F. Service. The race for the $1000 genome. *Science*, 311:1544 6, 2006.

[53] E. R. Mardis. Anticipating the $1,000 genome. *Genome Biology*, 7(7), 2006.

[54] M. L. Metzker. Emerging technologies in DNA sequencing. *Genome Research*, 15(12):1767 1776, 2005.

[55] E. Y. Chan. Advances in sequencing technology. *Mutation Research-Fundamental and Molecular Mechanisms of Mutagenesis*, 573(1-2):13 40, 2005.

[56] W. J. Greenleaf and S. M. Block. Single-molecule, motion-based DNA sequencing using rna polymerase. *Science*, 313(5788):801 801, 2006.

[57] J. J. Kasianowicz, E. Brandin, D. Branton, and D. W. Deamer. Characterization of individual polynucleotide molecules using a membrane channel. *Proceedings of the National Academy of Sciences of the United States of America*, 93(24):13770 13773, 1996.

[58] D. Fologea, M. Gershow, B. Ledden, D. S. McNabb, J. A. Golovchenko, and J. L. Li. Detecting single stranded DNA with a solid state nanopore. *Nano Letters*, 5(10):1905 1909, 2005.

[59] J. Lagerqvist, M. Zwolak, and M. Di Ventra. Fast DNA sequencing via transverse electronic transport. *Nano Letters*, 6(4):779 782, 2006.

[60] A. A. Simpson, Y. Z. Tao, P. G. Leiman, M. O. Badasso, Y. N. He, P. J. Jardine, N. H. Olson, M. C. Morais, S. Grimes, D. L. Anderson, T. S. Baker, and M. G. Rossmann. Structure of the bacteriophage phi 29 DNA packaging motor. *Nature*, 408(6813):745 750, 2000.

[61] G. C. Lander, L. Tang, S. R. Casjens, E. B. Gilcrease, P. Prevelige, A. Poliakov, C. S. Potter, B. Carragher, and J. E. Johnson. The structure of an infectious p22 virion shows the signal for headful DNA packaging. *Science*, 312(5781):1791 1795, 2006.

[62] L. Tang, W. R. Marion, G. Cingolani, P. E. Prevelige, and J. E. Johnson. Three-dimensional structure of the bacteriophage p22 tail machine. *Embo Journal*, 24(12):2087 2095, 2005.

[63] Y. Xiang, M. C. Morais, A. J. Battisti, S. Grimes, P. J. Jardine, D. L. Anderson, and M. G. Rossmann. Structural changes of bacteriophage phi 29 upon DNA packaging and release. *Embo Journal*, 25(21):5229 5239, 2006.

[64] K. R. Kondabagil, Z. H. Zhang, and V. B. Rao. The DNA translocating atpase of bacteriophage t4 packaging motor. *Journal of Molecular Biology*, 363(4):486 499, 2006.

[65] D. Fologea, J. Uplinger, B. Thomas, D. S. McNabb, and J. L. Li. Slowing DNA translocation in a solid-state nanopore. *Nano Letters*, 5(9):1734 1737, 2005.

[66] R. A. Bernal, S. Hafenstein, R. Esmeralda, B. A. Fane, and M. G. Rossmann. The phi x174 protein j mediates DNA packaging and viral attachment to host cells. *Journal of Molecular Biology*, 337(5):1109 1122, 2004.

[67] I. Ghosh, C. I. Stains, A. T. Ooi, and D. J. Segal. Direct detection of double-stranded DNA: molecular methods and applications for DNA diagnostics. *Molecular Biosystems*, 2(11):551 560, 2006.

Paper I

Programmable Self-Assembly Unique Structures and Bond Uniqueness

Journal of
Computational and Theoretical Nanoscience
Vol. 3, 1–7, 2006

Programmable Self-Assembly—Unique Structures and Bond Uniqueness

Björn Högberg* and Håkan Olin

Department of Physics, Mid Sweden University, SE-851 70 Sundsvall, Sweden

An important problem in nanotechnology is to develop a method for assembling complex, aperiodic, structures. While simple self-assembly will not be able to address this problem, programmable-, or algorithmic-, self-assembly is powerful enough to be a potential solution. Here, we address the question of how the basic properties of the constituent building blocks are related to the periodicity of the resulting assembly. By introducing the parameters unique structures, which gives a measure of the complexity of an assembly, and bond uniqueness, which gives a measure of how the building blocks fit together, we show how to quantify the structural quality of a general assembly system and present relations between the parameters. The introduced methods will be helpful when designing assembly systems to be used for direct fabrication of nanosystems or for nano-scaffolds and addressable arrays.

Keywords: Programmable Self-Assembly, Algorithmic Self-Assembly, Scaffold Assembly, Bond Uniqueness, Tile Systems.

RESEARCH ARTICLE

1. INTRODUCTION

Probably, the single most important problem in nanotechnology is to develop a method for assembling complex structures, such as a nanochip. Today, in microelectronics different types of lithography is used to define the structures. These methods are referred to as top-down methods. However, the continuing route towards smaller structures using top-down methods is increasingly difficult, which is reflected in fabrication costs that are growing much faster than the electronic market.[1]

A bottom-up building approach based on self-assembly has been widely discussed as an alternative method for nanofabrication. In self-assembly the building blocks, that could be atoms, molecules, or larger structures, diffuse around and eventually bind to a specific location. The simplest type of self-assembly, such as self-assembling monolayers or growth of nanowires, will merely lead to a non-complex crystal, and will be of little use to assemble a chip. We refer to this simplest type of self-assembly as *crystal self-assembly* (Fig. 1a). At the other extreme are building blocks that all have a unique address tag that will bind to a corresponding address. Using this *unique addressing self-assembly* method any kind of complex structure might be build, but a large number of building blocks are needed (Fig. 1b). Between these two extremes is *programmable self-assembly* (or algorithmic self-assembly)[2,3] (Fig. 1c).

A formal model for studies of programmable self-assembly is the Tile Assembly Model[4] where the building blocks called tiles are self-assembled into a square lattice. The model is an extension of Wangs theory of tiling,[5] but each of the sides of the square tiles contains a glue that allow binding of another tiles to the sides. The glue is specific so only a tile-side with the same corresponding glue will bind. Growth starts from a seed tile by adding one tile at the time. By designing the tiles in a specific way the growth can be programmed. Programmable self-assembly is powerful, for example, it has been shown to be capable of universal computation.[6] The Tile Assembly Model has been used for investigation of, for example, the minimum number of tiles needed to self-assemble a square of a certain size (program size complexity),[4] the time complexity,[7] optimal size and time complexity,[8] or whether a given tile system uniquely produce a given shape.[9] Furthermore, algorithmic self-assembly systems have been proposed as a tool to study self-replication[10] in a manner similar to von Neumann self-replicating machines.[11]

Experimentally, DNA double-crossover molecules with four sticky ends, which are analogues to the four sides of the Wang tiles, have been demonstrated to assemble into two-dimensional lattices.[12] Another experimental demonstration, of a much more complex pattern, is the algorithmic assembly of DNA Sierpinski triangles.[2] Macroscopic

*Author to whom correspondence should be addressed.

systems based on millimeter-scale plastic tiles that float on a liquid[13, 14] can also be used for experimental realization of algorithmic self-assembly.[3] Recently, DNA-linked nanoparticles has been suggested as a base for programmable self-assembly.[15] Another path, for doing programmable self-assembly, is to first self-assemble a scaffold of technologically simple building-blocks, then, self-assemble the interesting devices on the addressable sites created by the scaffold assembly. Potential scaffold materials are for example DNA-crystals[12, 16–19] mentioned above or DNA-linked nanoparticles[20, 21] as well as more advanced biological systems like proteins[22] or viruses.[23] Using metallization of the biomolecules, the scaffold can also act as an electrical contact between the devices assembled upon the scaffold.[24] For reviews on some of these self-assembly systems see Refs. [25, 26].

Suppose, that a complex nanosystem, say an electronic chip, is going to be built using programmable self-assembly. From an engineering point of view it is important to know the limits of the assembly system given an available technology. This paper describes a solution to this problem by introducing two parameters: *unique structures* which gives a measure of how complex the final structure is and the *bond uniqueness* parameter—which is related to the specificity of the glue or bonds. Together with other building block parameters, such as the number of different types and their total number, we will give limits as well as trade-off relations between different assembly designs. The task of designing an assembly system so that it gives the desired amount of unique structures is discussed in Section 4 where we propose a method for the design of square tile systems.

2. CLASSES OF SELF-ASSEMBLY

In the following we use the word tile to denote a general assembly building block. A tile have one or more functional edges with specific types of glues, or *bonds*.

Depending on the types of open bonds on the self-assembly tiles and on the physical process that makes the tiles assemble, we classify the process according to the type of assembly it can produce.

In *Crystal self-assembly* (Fig. 1(a)) patterns of tiles are reproduced throughout the assembly. Since a crystal self-assembly (CA) repeats itself, the neighborhood of each tile must be identical everywhere in the assembly. This leads to the conclusion that each open bond of a crystal tile can only make a bond with one specific type of tile. This constraint on the assembly process assures that each tile of a certain type has the same type of tile neighbors everywhere.

In a *unique addressing assembly* (UA) (Fig. 1(b)), each tile type only occurs once and the tile position is completely defined.[4] Each bond must be specific as to what type of tile it can accept, like in the CA case above. There must also exist a unique way to arrange the tiles so that there are no more open bonds or so that the remaining open bonds are blocked by the assembly itself.

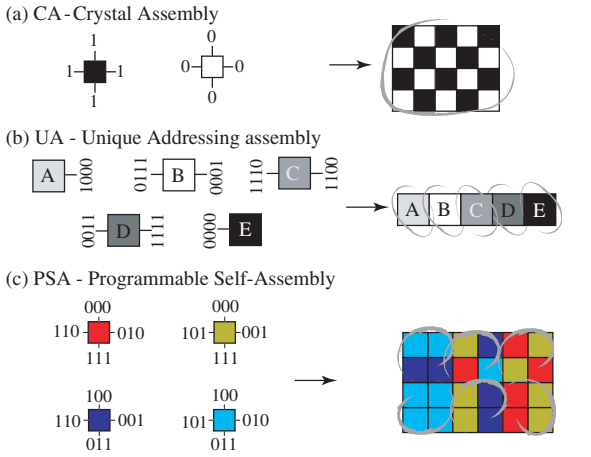


Fig. 1. Classes of self-assembly systems and unique structures (encircled in gray) in generated assembly patterns. (a) Type 1 bonds can connect to type 0 bonds. Each of the tiles bonds can only connect to one other type of tile. This leads to crystal type assembly. The only unique structure is the entire assembly. (b) Each tile has bonds that can connect only to one other type of tile, and each tile has no more than one type of each bond. The resulting assembly type is unique addressing where each tile constitutes a unique structure. (c) In this case each bond can host two different types of tiles. Together with the added criterion that each incoming tile must interact with at least two other tiles in order to assemble, the process is now of the PSA class. This assembly process needs to be nucleated for the assembly to grow. The resulting assembly in (c) has six unique structures of 4 tiles each ($S = 6$, $\eta = 4$).

A *Programmable Self-Assembly* (Fig. 1(c)) (PSA) system produces assemblies where one can find both repeating and unique patterns of tiles. For this to happen, some, or all, of the open bonds of a PSA tile must be able to bond to more than one *type* of tiles. Since an assembled PSA tile must, according to the above definition, accept at least two different types of new neighbors the process is not deterministic and thus not programmable; a single PSA tile alone cannot uniquely define the next tile in the assembly. At least one second neighboring, PSA tile needs to be present to uniquely determine the next tile type. One thus introduce the added criterion that a new tile must bind to at least two, already assembled, tiles; i.e., two tiles is the smallest structure that define the type of the third tile to be assembled. We do not know if this is generally true that programmable semi repetitive systems need this assembly criteria. This feature seems to be the single most important factor for the creation of complex assemblies. In nature, this type of dependent binding has been shown to be an important feature. For example in the self assembly of ribosomes certain proteins do only assemble if the growing ribosome assembly contains a certain pair of previously assembled proteins.[27]

Winfree and Rothemund has made a similar observation in discussions of the need for cooperation in order

RESEARCH ARTICLE

for the assembly process to be programmable.[4] Klavins et al.[28] has proved that no binary grammar can generate a unique stable assembly, however, this proof is valid for self-assembly systems with conformal switching tiles, i.e., building blocks that change state after being assembled.

3. UNIQUE STRUCTURES AND BOND UNIQUENESS

As noted in the introductory discussion above, one issue that makes self-assembly interesting for nanotechnology is the ability to construct scaffolds with uniquely addressable structures. In the following we will show how it is possible to estimate the number of unique structures that a certain assembly system can produce.

In each finite assembly, one can identify a certain number of unique structures, S. A unique structure is a number of connected tiles that is not reproduced anywhere else in the assembly. As an example, the string `ac` is a unique structure in `aaabacabca` while `ab` is not. The minimum number of tiles needed to define a unique structure is denoted η or, number of tiles per unique structure. The symbol N will be used to specify the total number of tiles that constitute the assembly in the discussion that follows. The number of tile *types* will be denoted by w.

The patterns in Figure 1 are examples of a few simple cases where finding S and η is straightforward. In general it is always possible to divide an assembly into S unique structures of η tiles in each structure. The pair (S, η) is defined to be; the maximum possible S, and the η that gives this maximum. A unique structure parser algorithm for finding the values of the pair η and S in any finite assembly, can be constructed. Note that unique structures may not overlap. This way, each unique structure can be interpreted as an address in a scaffold, to be used for subsequent assembly.

Consider the task of assembling a linear structure of S addressable locations. Since each location should be uniquely addressable, the structure constituting the location must be globally unique. One way to proceed is to make exactly S tiles that stick together in only one way (unique addressing, see Fig. 2(a)), or to make some number w of unique tiles where $w < S$ (PSA). If one chooses the later strategy one must figure out a clever way to make the w tile types assemble into S unique structures. Two examples of model PSA systems are found in Figures 2(b) and (c). The prime-tower assembly (Fig. 2(b)) counts to a product of two prime numbers and then stops the assembly. This type of process is similar to the *vernier* process in biology, that is believed to be a length controlling mechanism in linear protein assembly.[29] The counter by Cheng, Goel, and Moisset de Espanés (CGM)[8] is an assembly counter that is optimized to use as few tile-types as possible (optimized for low w) and still be able to produce arbitrarily long sequences. Each row is $\log_2 S$ tiles wide and constitutes a unique structure and $w = 8$.

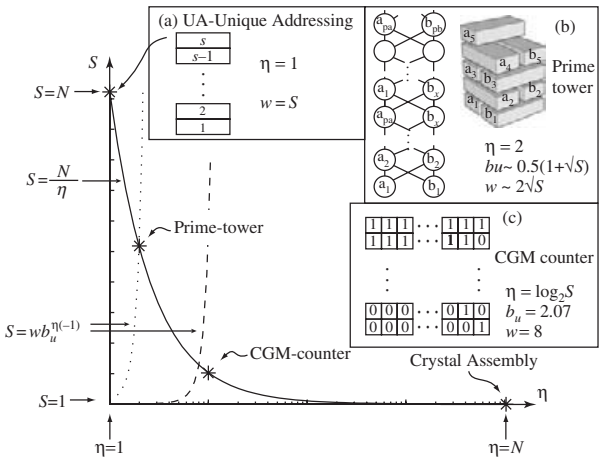


Fig. 2. A plot of S versus η for different kind of linear-, counter type-, assemblies at constant number of total tiles, N. The η-axis is logarithmic. The solid line represents the bound $S = N/\eta$. The insets (a)–(c) show model systems of linear assembly displaying S unique structures. The process in (a) is an example of a one-dimensional UA assembly. (b) Shows a modulo-prime counter. Two sets of tiles, a's and b's, consisting of p_a and p_b tiles each ($p_{a,b}$ are prime numbers). The assembly stops when tiles a_{p_a} and b_{p_b} are next to one another. To the right: physical model of the assembly. Left: a graph displaying the bond structure. In (c) the counter is of the type described in Ref. [8] (referred to as CGM-counter in this paper). This is a pseudo binary counter under the tile assembly model[4] that uses 8 distinct tiles. In the described model systems each row constitutes a unique structure. The prime number counter have $\eta = 2$ and $S \sim N/2$. The CGM counter have $\eta = \log_2 S$ where S is determined from the equation $S = N/\log_2 S$. The dotted and the dashed lines are Eq. (2) written with the values of w and b_u for the prime-counter and the CGM counter, respectively. (The graph is plotted for $N = 100$.)

When the total number of tiles, N, is constant, it is possible to draw some conclusions about S and η for several types of assembly processes. The simplest cases being unique addressing, where S would be equal to N and $\eta = 1$, and crystal type assembly, where S would be 1 and $\eta = N$. Programmable self assembly lies somewhere in between. We start the discussion by considering the number of possible ways one may assemble a substructure containing η tiles.

If the assembly is random, the number of ways that one may combine η tiles chosen from w types is given by w^η. PSA processes are not random however, and the number of ways one may combine tiles into structures of η tiles is limited by the *bond uniqueness* of the assembly system.

The *bond uniqueness*, b_u, for an assembly system is defined as the average number of tile types that each open bond can bind to. For example in the assembly system in Figure 1(c) the bond uniqueness is equal to two because each specific open bond can only harbor two different tile types on average. If the number of bonds per tile is denoted k ($k = 4$ for the square-tile systems discussed below, since each tile has four bonds), the total number of bonds on all the tile types is wk. Each bond on each tile type is given an index i. The bond uniqueness is then calculated in the

RESEARCH ARTICLE

following way:

$$b_u = \frac{\sum_i^{kw} \text{No. of tile types that bond } i \text{ can bind to}}{kw} \quad (1)$$

Returning now to the construction of structures containing η tiles. The first tile in such a structure can be chosen at will from the w available tile types. Once this tile is specified however, the following tile types to be placed next to the original tile, can only be chosen from a subset of the tile types, containing b_u tiles. So a row (η tiles wide) in a linear-, counter type-, assembly can be constructed in $wb_u^{\eta-1}$ number of ways. If each row is to constitute a unique structure we get that the maximum number of rows, i.e., the maximum S for the given η is:

$$S = wb_u^{\eta-1} \quad (2)$$

This equation, together with $N = S\eta$, gives the number of unique structures for linear-, counter type-, assemblies (see Fig. 2).

In two dimensional assembly we consider each unique structure to be a square of η tiles. Once the edge is specified, in the form of an L, the interior is uniquely determined by the assembly system. The corner tile can be chosen among all w tile types and each arm of the L-shaped edge can be chosen in $b_u^{(\eta^{1/2}-2)}$ ways (the structures are assumed to be square and the edges thus $\eta^{1/2}$ tiles long). In general we get that the number of possible ways to build structures containing η tiles is given by the following expression:

$$\text{No. of possible } \eta\text{-structures} = wb_u^{(\eta^{1/D}-D)} \quad (3)$$

where D is a dimensionality constant that is equal to 2 or 3 depending on whether the assembly is two-dimensional or three-dimensional.

Consider a two dimensional assembly with N tiles (assume that it is $\sqrt{N} \times \sqrt{N}$ tiles big). There are $(\sqrt{(N)} - \sqrt{\eta}+1)^2$ locations where a $\sqrt{\eta} \times \sqrt{\eta}$-tiles structure may be located (result from 2D-word pattern matching see for example Ref. [30]). Assume that each one of these structures must be unique and that we use the maximum number of variations of η-structures that the tile system may produce. Then from Eq. (3) we get that

$$wb_u^{(2\eta^{1/2}-2)} = (\sqrt{(N)} - \sqrt{\eta}+1)^2 \quad (4)$$

And in general

$$wb_u^{(D\eta^{1/D}-D)} = (N^{1/D} - \eta^{1/D}+1)^D \quad (5)$$

Now note again that, since the S unique structures that we seek may not overlap, the total number of tiles is η taken S times:

$$N = S\eta \quad (6)$$

Using (6) to solve for S in (5), and taking logarithms we get the following result:

$$S = N\left(\frac{D \log b_u}{\log\left[\frac{b_u^D}{w}((N^{1/D} - (\frac{N}{S})^{1/D}+1)^D)\right]}\right)^D \quad (7)$$

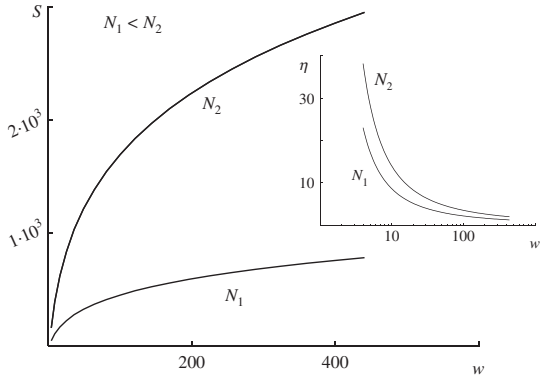


Fig. 3. Characteristics resulting form Eq. (7) showing the number of unique structures, S, in a two dimensional assembly plotted as a function of the number of tile types, w. $b_u = \sqrt{w}$ is assumed. Plotted for two values of total number of tiles N where $N_1 = 1000$ and $N_2 = 6000$. As the number of tile types is increased the growth of S slows down. The inset shows the same curves but instead of S we plot the size of the unique structures, η, on the y-axis (where $\eta = N/S$). The w-axis is logarithmic in this case. As w is increased the size of the unique structures rapidly decreases.

Equation (7) gives the maximum number of unique structures of a PSA process of N total tiles with w tile types and bond uniqueness b_u. This relation gives an optimal number of unique structures, in practice, the value of S could be less than this depending on the periodicity of the seed or nucleation process used. In Figure 3 a graphical interpretation of (7) is given.

3.1. Example of Application

The results above will be helpful when designing nanosystems using self-assembly. Consider an example where one wants to construct a scaffold for nanodevices. Lets say that we have the technology to produce 9 types of building blocks that each measures 13 nm × 13 nm. They have $b_u = 3$. How many uniquely addressable structures can we optimally produce in an area measuring 1 μm × 1 μm. The total number of tiles we have room for is thus $N \approx 5900$. Using Eq. (7) with $D = 2$, and solving numerically, we get that the maximum number of unique structures is approximately 380 so $\eta = N/S \approx 15.5$. It should thus be possible to construct a scaffold comprising some 370 addressable sites, each containing 16 tiles. (If one uses $\eta = 15$, which is less than the estimated value, the assembly system will not be able to cover the entire area with unique structures, some will necessarily be repeated.) The unique structures/addressable locations are thus in the order of 50 nm × 50 nm. If one wishes to make them smaller, more tile types are needed.

4. DESIGN OF A SELF-ASSEMBLY SYSTEM

Once the important design parameters are fixed, how do one proceed with the actual design of the building blocks? We assume that the values of w (number of tile types) and

b_u (bond uniqueness) have been determined to provide the desired amount non-periodicity in the assembly by using Eq. (7) (i.e., the desired amount of unique structures). Lets consider the case where a two-dimensional assembly is constructed from square tiles, each of them having four bonds to assemble into a square lattice. Similar schemes as the one presented here may be used for any type of plane-filling, or space-filling tiles. The bonds are considered to be of a complimentary nature so that a bond may not bind to a bond of the same type but only to its complement. Examples of complementary bond types are DNA-strands and jigsaw-puzzle pieces.

The tiles can be made non-rotatable by dividing the bonds into north-south and east-west pairs.[†] North-bonds must thus be complementary to south-bonds and east-bonds complementary to west-bonds. We denote by n_{NS} and n_{EW} the number of bond-*pairs* that are of the north-south-, and east-west-type, respectively. So, the total number of bonds of the south type for example is n_{NS}, of the north type n_{NS}, and together they form n_{NS} complementary pairs. Assembly growth is possible in any direction.

Remember that the bond uniqueness is the average number of tile-types that each bond can bind to. Now we know that the number of south bonds is n_{NS}, the number of tile-types is w, so each type of south bond can be found on w/n_{NS} tile types on average. By consequence, the north bonds will have an average bond uniqueness of w/n_{NS}. By the same argument the south bonds will have an average b_u of w/n_{NS} and the east and west bonds an average b_u of w/n_{EW} each. The total bond uniqueness is the average over the four E-W-S-N bond classes:

$$b_u = \frac{w}{2}\left(\frac{1}{n_{\mathrm{NS}}} + \frac{1}{n_{\mathrm{EW}}}\right) \tag{8}$$

To relate the values of n_{NS} and n_{EW} with the number of tile types, w, we will now consider the number of possibilities to construct *corner sites*. A corner site is a site where programmable self-assembly can occur, i.e., a site in the assembly where an incoming tile may bind to at least two other assembled tiles. In this discussion we consider square, two-dimensional tiles, so the corner sites are L-shaped trimers, like the one in Figure 4. The bottom left tile[‡] can be chosen at will from the w tiles whereas the tiles to the north and east of this tile can be chosen in w/n_{NS} or w/n_{EW} ways, respectively. So the total number of possible L-shaped trimers is $w^3/(n_{\mathrm{NS}}n_{\mathrm{EW}})$. Looking now at an individual tile, A, the number of L-shaped trimers that can be created by starting out from tile A is determined by

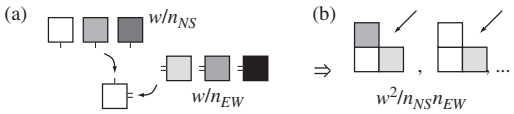


Fig. 4. (a) Each open north-, or east-bond can bind to w/n_{NS} or w/n_{EW} tile types on average. This creates on average, for each starting tile, $w^2/(n_{NS}n_{EW})$ possible corners like the one marked by an arrow in (b). Each of these corner sites must uniquely define the tile type that should bind to the site. If the number of created corners is greater than w for some tile then one tile type must fit two of the created corner sites. This violates the principle of programmable self-assembly and would yield a non-deterministic assembly process.

the bond uniqueness of the bonds of tile A. One thing is certain however, tile A must not be able to create more corner sites than there are tile types. If this would be the case then at least one of the tile types would have to fit more than one of the created corner sites. This would in turn violate the principle of programmable self-assembly that each incoming tile is *uniquely* determined by binding to at least two bonds. Thus we conclude that the number of corner sites each individual tile may create must be less than or equal to w. The average number of L-shaped trimers each tile may create is $\frac{w^3}{n_{\mathrm{NS}}n_{\mathrm{EW}}}/w$, so $w^2/(n_{\mathrm{NS}}n_{\mathrm{EW}}) \leq w$ following the above arguments. However, if one tile generates less than w corner sites, then another tile must generate more than w corner sites in order for the total number of corner sites to be correct. As mentioned, this is not allowed and the only possibility left is thus

$$\frac{w^2}{n_{\mathrm{NS}}n_{\mathrm{EW}}} = w \Rightarrow w = n_{\mathrm{NS}}n_{\mathrm{EW}} \tag{9}$$

The only exception to this is when $b_u = 1$, i.e., when the assembly is of the crystal type and not of the PSA type, then each tile can generate exactly one corner site and $n_{\mathrm{NS}} = n_{\mathrm{EW}} = w$.

The Eqs. (8) and (9) constitute an equation system with the following solution:

$$n_{\mathrm{NS,EW}} = b_u \pm \sqrt{b_u^2 - w} \tag{10}$$

where n_{NS}, n_{EW}, and w must be positive integers. Since all the numbers in (10) are real, we get the following constraint on the values of the bond uniqueness

$$b_u^2 \geq w \tag{11}$$

4.1. Designing Tiles from Corners

Following the results above we can give a general method for designing square, non-rotating, tile systems having a certain number of tile types and bond uniqueness. The method can be extended to non-rotating tile types of other geometries.

A *corner* is defined as a pair of two adjacent bonds on a tile. Square tiles will have four corners, i.e., four pairs of

[†]Note that the tiles are still physically rotatable, but by the way we organize the bond-pairs the tiles will be forced to line up with the rest of the assembly so that every copy of a tile will always have the same rotation.

[‡]The choice of direction of the trimer construction is arbitrary and the results will be the same if one considers L-shaped trimers with different rotations.

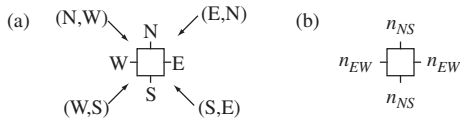


Fig. 5. (a) The corners of a square tile. (b) The number of choices for the types of bonds is n_{NS} for the north and south bonds and n_{EW} for the east and west bonds. Since each pair of bonds at a corner must be unique, $w \leq n_{\mathrm{NS}} n_{\mathrm{EW}}$.

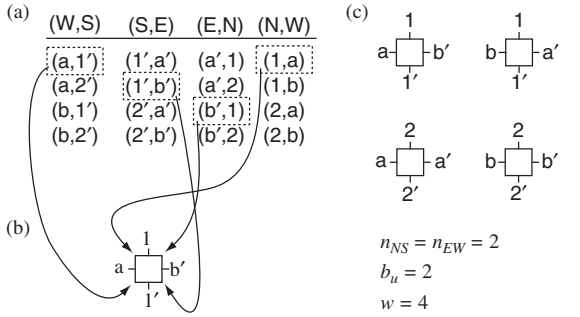


Fig. 6. Example showing the construction of a tile-system with $n_{\mathrm{NS}} = n_{\mathrm{EW}} = 2$. Primed bonds fit the unprimed counterpart. `a,b` are west bonds, `a′,b′` east bonds, and `1,2` and `1′,2′` north and south bonds, respectively. (a) A table showing the total number of possible WS, SE, EN, and NW bond-pairs. (b) Four corner bond-pairs makes one tile. (c) Proceeding in the same manner as in (b) generates a complete tile-system with $w = 4$ and $b_u = 2$. (This is the same system as in Fig. 1(c)).

bonds: (W, S), (S, E), (E, N), and (N, W),[¶] see Figure 5(a). Hexagonal tiles will have six corners and cubes in three-dimensional assembly will have eight corners where each corner will be a bond triplet.

When the number of bondtypes, n_{NS} and n_{EW}, have been decided, one may construct $n_{\mathrm{NS}} n_{\mathrm{EW}}$ corner pairs for each type of corner (WS, SE, EN, and NW), in total $4 n_{\mathrm{NS}} n_{\mathrm{EW}}$ pairs. To construct the tiles, simply pair the corners four and four, see Figure 6 for an example. Because Eq. (9) holds, all corner pairs must be used, and each corner pair must only be used once. If one corner pair were to occur on two tiles, the corresponding corner site would not uniquely define a single tile and the assembly would be random.

4.2. Example

The proposed method for tile design should be proceeded with a check for the technology limits in order to see that the desired requirements can be met. The following is an example of such a check:

Suppose we have a technology to produce 10 nm × 10 nm square tiles using a maximum of 14 bond pairs (suppose for example that we only have 28 DNA-strands to work with). So $n_{\mathrm{NS}} + n_{\mathrm{EW}} = 14$. Is it possible to cover a surface of 1 μm × 1 μm and get uniquely addressable structures that are maximum 4 tiles big ($\eta = 4$, $S = 2500$, $N = 10000$)? By combining Eqs. (8) and (9) we find that $b_u = \frac{1}{2}(n_{\mathrm{NS}} + n_{\mathrm{EW}})$, so $b_u \leq 7$. Since $w = n_{\mathrm{NS}} n_{\mathrm{EW}}$ we know that the maximum number of tile types is $w = 49$. Using Eq. (7) with $D = 2$ we get that the maximum number of unique structures that we can expect is $S = 1796$ so the answer is no, we would need more bond types.

[¶]The notation assumes a counter-clockwise walk around a tile, this is why we use WS and EN instead of SW and NE.

5. DISCUSSION

Depending on the application, structures that show repetitions if rotated may, or may not, count as unique. If we do not regard structures, that show repetitions while rotated, as unique, then we must divide the total number of unique structures by the rotational symmetry of the lattice. In the case of square tiles the factor is four. If one considers the case of constructing nanosystems using assembly of nanodevices the orientation of the nanodevices will often be important and we should thus take this factor into consideration.

Another important issue that will affect the assembled structure is the relative concentrations of the tile-types. In Eq. (7) we assume that the relative concentrations are all equal $1/w$. Any deviation from this will lead to a reduced number of unique structures.

Looking now at the error rate in self-assembly, how is it related to the bond uniqueness? At thermal equilibrium the probability of forming a certain configuration L is given by:[31]

$$\mathcal{P}_L = \frac{\exp(-U_L/k_B T)}{\sum_{L_i} \exp(-U_L/k_B T)} \tag{12}$$

where U_L is the energy of the formation, T is the temperature, and k_B is Boltzmanns constant. Assuming that the energy of all bonds are equally strong and the binding energy is equal to U. Assuming that the correct binding of a tile will have binding energy $2U$ (two bonds fit) and any incorrectly bonded tile (only one bond fit) will have energy U. For each open bond in an assembly there are b_u tiles that may bind to that bond but only one of these tiles is the correct one. Consequently, for a corner site with two open bonds there are $2(b_u - 1)$ possible ways to add an incorrect tile type by making a single bond. Following this discussion, we get that the probability of a correct assembly event is given by:

$$\begin{aligned} \mathcal{P}_L &= \frac{\exp(-2U/k_B T)}{\exp(-2U/k_B T) + 2(b_u - 1)\exp(-U/k_B T)} \\ &\simeq 1 - 2(b_u - 1)\exp(U/k_B T) \end{aligned} \tag{13}$$

The complimentary probability is the probability of errors and as shown in Eq. (13) it is proportional to b_u. If one is concerned about the correctness of the assembled structures it is thus advisable to choose a low value for the bond uniqueness.

Since we have found that, for square tile systems that are made non-rotating, $b_u \geq w$, we can also conclude that the error rate is at best proportional to $\sqrt{w}$. Making complex assemblies with spatially small unique structures (low η, high w) might therefore be harder than previously

RESEARCH ARTICLE

assumed. Proposed methods for error correction exists[32] but since they rely on subdivision of the tiles, in effect creating even more tiles per unique structure, one needs to take this into consideration when evaluating the size of the unique structures. The results in this paper are still valid, the extra tiles needed for error correction is simply a multiplication factor of the number of tile types needed and of the resulting number of tiles per unique structure.

6. CONCLUSIONS

We have presented a method for the parametrization of assembly systems derived from their ability to form unique structures. We have also introduced the concept of bond uniqueness and showed how it influences the number of unique structures that a programmable self-assembly system can create. The structural complexity is heavily dependent on the bond uniqueness of the system. By using the relations obtained in this paper (mainly Eq. (7)) for nanotechnology applications it will possible to vary the parameters w (number of types of building blocks) and b_u (their bond uniqueness) to obtain the desired structural complexity and the desired size (η) of the addressable locations/unique structures. In Section 4 we showed how these parameters influences the number of bond types required, and the design of the actual building blocks to be used for the assembly. The introduced concepts will prove helpful when designing tile systems and evaluating the theoretical limits of a proposed self-assembly technology at an early stage.

References

1. R. Compañó (ed.), *Technology Roadmap for Nanoelectronics*, European Commission, Office for Official Publications of the European Communities, Luxemburg **(2001)**.
2. P. W. K. Rothemund, N. Papadakis, and E. Winfree, *Plos Biology* 2, 2041 **(2004)**.
3. P. W. K. Rothemund, *Proc. Nat. Acad. Sci.* 97, 984 **(2000)**.
4. P. W. K. Rothemund and E. Winfree, in *Proc. 32nd Ann. ACM Symp. Theor. Comp.* New York, NY and Portland, OR, USA **(2000)**, p. 459.
5. H. Wang, *Bell Sys. Tech. J.* 40, 1 **(1961)**.
6. E. Winfree, in DNA Based Computers, *DIMACS series in Discrete Math. and Theor. Comp. Sci.*, edited by R. J. Lipton and E. B. Baum, American Mathematical Society, Princeton **(1996)**, Vol. 27, p. 199.
7. L. M. Adleman, Q. Cheng, A. Goel, and M. D. Huang, *ACM Symposium on Theory of Computing (STOC'01)* **(2001)**, p. 740.
8. Q. Cheng, A. Goel, and P. Moisset de Espanés, in *Proceedings of Foundation of Nanoscience: Self-Assembled Architectures and Devices*, Sciencetechnica, Snowbird, Utah **(2004)**.
9. L. M. Adleman, Q. Cheng, A. Goel, M. D. Huang, D. Kempe, P. Moisset de Espanés, and P. W. K. Rothemund, *ACM Symposium on Theory of Computing (STOC'02)* **(2002)**, p. 23.
10. R. Schulman and E. Winfree, in *Advances in Artifical Life, Proc., Lect. Notes Artif. Intel.* **(2005)**, Vol. 3630, p. 734.
11. D. Mange, A. Stauffer, L. Peparolo, and G. Tempesti, *Proc. IEEE* 92, 1929 **(2004)**.
12. E. Winfree, F. R. Liu, L. A. Wenzler, and N. C. Seeman, *Nature* 394, 539 **(1998)**.
13. K. Hosokawa, I. Shimoyama, and H. Miura, *Sens. Actuators A: Phys.* 57, 117 **(1996)**.
14. N. Bowden, A. Terfort, J. Carbeck, and G. M. Whitesides, *Science* 276, 233 **(1997)**.
15. B. Högberg, J. Liu-Helmersson, S. Holm, and H. Olin, *Appl. Surf. Sci.* **(2006)**, in press.
16. S. H. Park, H. Yan, J. H. Reif, T. H. LaBean, and G. Finkelstein, *Nanotechnology* 15, 525 **(2004)**.
17. D. Dwyer, S. H. Park, T. LaBean, and A. Lebeck, in *Proceedings of the 2nd Conference on the Foundations of Nanoscience*, Sciencetechnica, Snowbird, Utah **(2005)**, p. 187.
18. S. H. Park, C. Pistol, S. J. Ahn, J. H. Reif, A. R. Lebeck, C. Dwyer, and T. H. LaBean, *Angew. Chemie Int. Ed.* 45, 735 **(2006)**.
19. Y. Y. Pinto, J. D. Le, N. C. Seeman, K. Musier-Forsyth, T. A. Taton, and R. A. Kiehl, *Nano Lett.* 5, 2399 **(2005)**.
20. A. P. Alivisatos, K. P. Johnsson, X. G. Peng, T. E. Wilson, C. J. Loweth, M. P. Bruchez, and P. G. Schultz, *Nature* 382, 609 **(1996)**.
21. C. A. Mirkin, R. L. Letsinger, R. C. Mucic, and J. J. Storhoff, *Nature* 382, 607 **(1996)**.
22. Shuguang Zhang, *Biotechnol. Adv.* 20, 321 **(2002)**.
23. A. S. Blum, C. M. Soto, C. D. Wilson, J. D. Cole, M. Kim, B. Gnade, A. Chatterji, W. F. Ochoa, T. W. Lin, J. E. Johnson, and B. R. Ratna, *Nano Lett.* 4, 867 **(2004)**.
24. H. Yan, S. H. Park, G. Finkelstein, J. H. Reif, and T. H. LaBean, *Science* 301, 1882 **(2003)**.
25. G. M. Whitesides and B. Grzybowski, *Science* 295, 2418 **(2002)**.
26. N. C. Seeman, *Nature* 421, 427 **(2003)**.
27. G. M. Culver, *Biopolymers* 68, 234 **(2003)**.
28. E. Klavins, *IEEE Proc. Int. Conf. Robotics and Automation* 5, 5293 **(2004)**.
29. J. S. Lindsey, *New J. Chem.* 15, 153 **(1991)**.
30. A. Carpi and A. de Luca, *Int. J. Found. Comp. Sci.* 15, 355 **(2004)**.
31. B. N. Roy, *Fundamentals of Classical and Statistical Thermodynamics*, Wiley, West Sussex, England **(2002)**.
32. E. Winfree and R. Bekbolatov, *Lect. Notes Comp. Sci., DNA Computers* 9, 126 **(2004)**.

Received: 27 February 2006. Accepted: 9 March 2006.

RESEARCH ARTICLE

Paper II

DNA Coated Nanoparticle Eight-mers as Programmable Self-Assembly Building Blocks

Proc. Foundations of Nanoscience, p.219-226, ScienceTechnica, Snowbird, Utah (2005)

DNA Coated Nanoparticle Eight-mers as Programmable Self-Assembly Building Blocks

Björn Högberg[*†], Jing Helmersson[†,‡], Svante Holm[+], Håkan Olin[*†]

[†] Dept. of Engineering Physics, Mid Sweden University
[‡]Dept. of Physics and Astronomy, California State University Long Beach
[+]Dept. of Biology, Mid Sweden University

Abstract. Nanoparticles coated with single stranded DNA have been shown to efficiently hybridize to targets of complementary DNA. This property might be used to implement programmable (or algorithmic-) self-assembly to build nanoparticle structures. However, we argue that a DNA coated nanoparticle by itself cannot be used as a programmable self-assembly building block since it does not have directed bonds. A general scheme for assembling and purifying nanoparticle eight-mers with eight geometrically well-directed bonds is presented together with some preliminary experimental work.

1. Introduction

DNA functionalized nanoparticles has been a prospect material for the construction of self-assembled structures ever since they were first demonstrated [1, 2]. So far, the main interest in these particles has been to use them in detection methods for small amounts of specific DNA [3]. The structures formed in these, and other [4] experiments, are mainly periodic-, or random agglomerations of particles. In order to fully exploit the potential of self-assembly of DNA coated nanoparticles one needs to address the problem of assembling nonperiodic structures.

2. Programmable Self-Assembly Building Blocks

It has become clear that in order to implement programmable self-assembly (PSA) one needs building blocks of a certain complexity and diversity [5-8, 9]. Besides different types of building blocks, PSA also requires that the building blocks have some minimal geometrical complexity. It is not possible to implement PSA using parts that have the same kind of bonds on all faces. Using these kind of building blocks will inevitably lead to periodic, noncomplex structures.

* Corresponding authors: bjorn.hogberg@miun.se, hakan.olin@miun.se

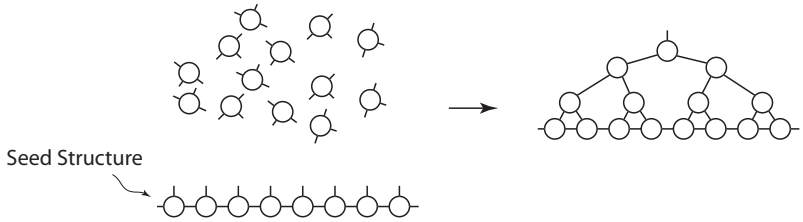


Figure 1 The sudden death of a PSA assembly with three-bond building blocks. As each incoming block needs to interact through bonds with at least two already assembled blocks, it has only one bond left for new building blocks. The structure quickly closes up and a seed with n+1 open bonds is needed to assemble a structure with n building blocks.

Furthermore, the bonds of the building blocks in a PSA system must be unspecific as to what type of *building block* it can make a bond to. (Note, that this is not the same as being unspecific as to what type of *bond* it will bind to.) Complex structures can only be assembled when bonds are unspecific in this way. If all bonds are specific and only fit together with one specific type of building block, the resulting structure will be either periodic or uniquely addressed [9].

Since the bonds are unspecific, the resulting assembly will be random unless another restriction is imposed on the process: *In a PSA process, an incoming building block must interact with at least two other, already assembled blocks, in order to assure correctness of the PSA assembly*. The reason for this is, that one bond alone will not determine the type of the next, to be assembled, building block, because of the ambiguity of bonds discussed above. If an incoming building block is forced to make at least one other bond as it assembles, the choice of blocks would be further limited. Preferably the choice could be limited to one specific block and the process would then be deterministic and programmable. In this report we will not go further into the details of the PSA process itself but concentrate on the production of the PSA building blocks. For general discussions on criteria for PSA see for example [9-11].

One consequence of the above reasoning is that a PSA building block must have at least two bonds, in order to interact with at least two, already assembled blocks. Furthermore, if the assembly process is to continue and not be blocked, it must also have two bonds left for new, incoming, building blocks. Having only one bond left is not enough, because if an incoming block has just one connection left after it has made bonds to two already assembled blocks, it has reduced the number of open bonds in the assembly by one. This leads to the conclusion that to PSA assemble a structure of n one need to start out with a seed with n+1 open bonds (see Figure 1). So in practice a general PSA building block must have at least four bonds. It is possible to build three-dimensional structures using building blocks with only four bonds. However, one would then be required to create a kind of diamond like structure (using tetrahedron building blocks with functionalized corners). A more practical approach is to use space-filling building blocks like triangular prisms or cubes having at least 5, respectively 6 bonds (6, respectively 8 bonds, if the bonds are situated at the corners).

Building blocks that have a size of tens of micrometers can be manufactured and functionalized using conventional micro-technology. Standard techniques has also

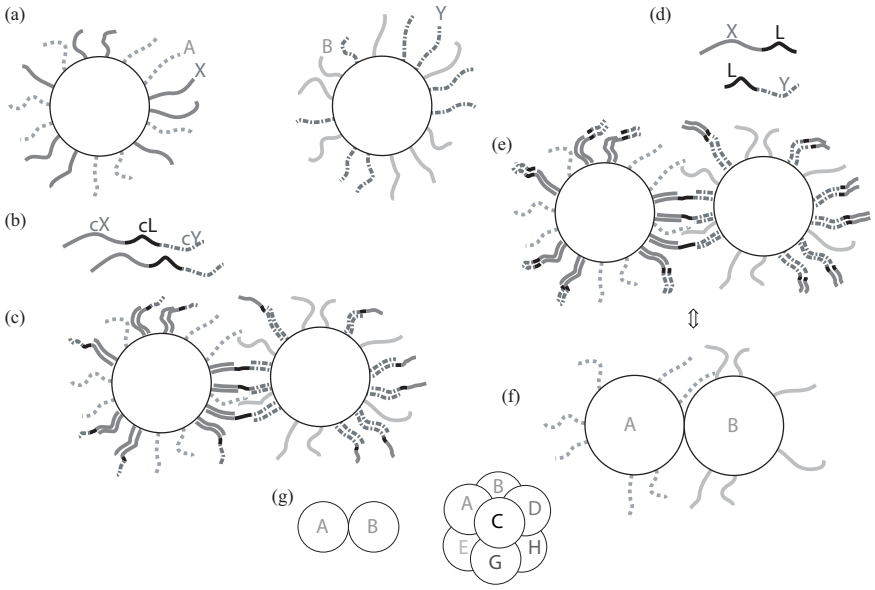


Figure 2 Scheme for nanoparticle PSA building block assembly. (a) Two different sets of nanospheres are each coated with two (or more) single stranded DNA sequences A(green),X(blue) and B(yellow),Y(red) respectively. These nanospheres are mixed in solution and DNA-linker molecules cX-cL-cY (blue-black-red)are added (b). The linker molecules hybridize with the complementary parts on the nanospheres, making the nanospheres bind together (c). Dimers can be purified out of the solution by chromatographic methods. Adding the complementary molecules (X-L and L-Y) (d), passivates the remaining sticky ends of the linker molecules (e). The only single stranded DNA left is of type A on one sphere and of type B on the other so the dimer in (e) is functionally equivalent to the one in (f). Using several iterations of this processes one can, for example, produce eight-mers with eight separate and specific binding sites to be used for PSA.

been tried on colloidal particles. One example of a top-down approach for the functionalization of two distinct areas of 1 μm particles can be found in [12]. However, as the dimensions of the building blocks are reduced, it is increasingly difficult to functionalize different parts of the building blocks with different functions. When dealing with nanoscale objects, a top-down procedure for selective functionalization is no longer feasible. A new approach is needed. We here propose such a new method for the fabrication of nanoscale PSA building blocks using only bottom-up methods.

3. Making PSA-Building Blocks from Nanoparticles

Firstly, nanospheres of a suitable material are functionalized with two different types of single stranded DNA (ssDNA) (called particles of type I below). For example, for

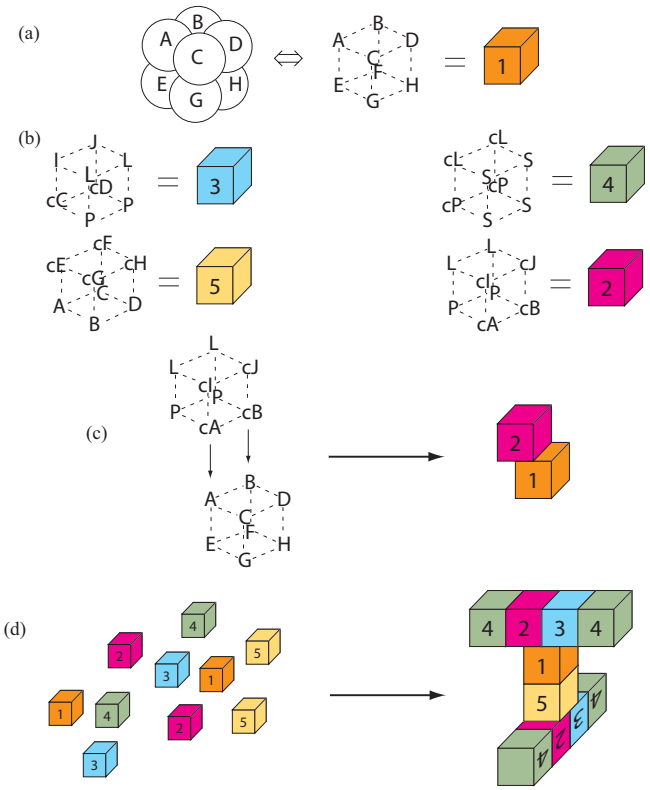


Figure 3 Scheme of a self-assembly process, using DNA nanosphere eight-mers as building blocks. The eight-mers have eight specific ssDNA binding sites. These eight-mers can be thought of as cubes having functionalized surfaces (a). In (b) four more eight-mers are depicted. When mixed in solution the DNA complements will bind to each other. In (c) for example, the cA and cB of block 2 binds to the A and B strands on block 1. Mixing all the building blocks of (a) and (b) will lead to the formation of structures like the one on the right in (d). Note that the "S" DNA of block 4 has no complement on the other blocks so block 4 acts as a stop-block.

gold [13, 14] and SiO_2 [15] nanoparticles, there exists standard protocols for this. Secondly, nanospheres of a second kind are prepared in the same way but using yet another set of different DNA sequences (these are called type II particles). These spheres are then mixed in solution, see Figure 2(a).

Consider the case when nanospheres of type I have ssDNA sequences A and X and the nanospheres of type II have sequences B and Y. Linker molecules, consisting of a chain of complementary sequences to Y (cY) and X (cX) coupled by some sequence of bases that are irrelevant to the assembly process (cL) are added to the solution, see Figure 2(b). As the linker molecules are added the spheres of type I and II will adhere to one another and form aggregates of different sizes. By controlling the temperature of the solution we can promote the formation of dimers. After a while the linker DNA

is rinsed away, stopping any further aggregation of the nanoparticles. The dimers could then be separated from the rest of the aggregates by centrifugation, gel electrophoresis or some other mass separation technique. If necessary, the sticky ends of the linker molecules could be passivated by adding a solution of LY and LX DNA (Figure 2(d)-(e)). The dimers produced in this way constitute simple PSA building blocks with ssDNA of types A and B sticking out at different ends of the dimers. By starting out with particles coated with four specific DNA sequences and using several iterations of this technique one could make four-mers and eight-mers to be used as more advanced PSA bulding blocks, Figure 2(g).

Following the above method one can create a large number of different basic building blocks with specific sticky ssDNA on different faces of the blocks. As the basic building blocks are mixed together with blocks having complementary DNA self-assembly will occur. Figure 3 contains an example of an assembly process using these basic PSA nanosphere building blocks.

As discussed above in section 2, the programmable self-assembly process will require simultaneous binding of two bonds in order to uniquely produce the desired structures. For example, in Figure 3, the blocks 5 and 1 can connect to the 2 block. The right position is determined by matching two bonds. The A DNA's of both block 1 and 5 could, in principle, make a single bond to the cA DNA of block 2 but the alignment of the connected blocks would probably be wrong. This could be avoided by setting the temperature and/or the amount of stirring to values where blocks bonded by one bond are unstable structures whereas blocks bonded by two bonds are stable.

3. Materials and Methods

Gold colloids with mean diameters ranging from 10 nm to 50 nm were purchased from G.Kisker GbR. Typical size distributions were about 15%, concentrations are approximately 0.07 nM for the 50 nm up to about 7 nM for the 10 nm particles. Thiol modified (5' and 3') and unmodified ssDNA were purchased from Cybergene AB. All oligos were HPLC-purified by the manufacturer. The length of the thiolated oligonucleotides were 16 code bases and 10 bases of consecutive T's between the thiol group and the coding sequence. The ten T's act as a separator between the particle and the hybridizing sequence. The coding sequences were optimized to have cross-hybridization probability as small as possible. The sequences were also checked for unwanted hairpin and dimer formation. To this end we used software developed by ourselves and some online tools [16]. We have primarily used a subset of the DNA library for DNA-computing by Penchovsky and Ackermann [17]. This set of oligos is well optimized for uniqueness and the sequences are also optimized to have similar melting temperatures.

The functionalization of the gold particles followed a modified version of the Storhoff protocol [3]. Equal amounts of 4 different thiolated oligos to a total of 1 nmole DNA are mixed separately and then mixed with 1.3 ml of the gold colloid. These samples are stored at 30 ° C overnight followed by the addition of a sodium phosphate buffer to 0.01 M (pH 7) and a NaCl solution up to 0.1 M. The addition of

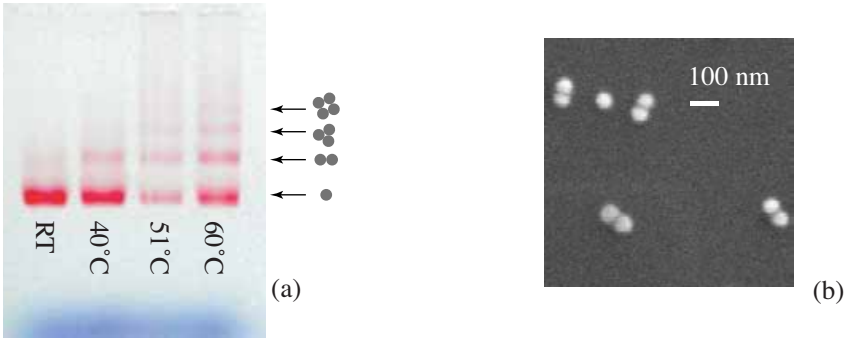


Figure 4 (a) Photograph of the discrete red lines formed by agarose gel electrophoresis of 50 nm DNA coated nanoparticles with linker molecule added. Bands of increasingly bigger aggregates are visible. The temperatures below the bands refer to the incubation temperature. (b) Electron micrograph of electrophoresis-purified nanoparticle dimers. The concentration of single particles is still relatively high in these samples, probably due to to breaking up of some of the dimers when the particles are extracted from the gel by centrifugation.

salt is done drop wise and in two steps separated by roughly two hours. Furthermore, the addition of salt is done at an elevated sample temperature, typically 70-90° C. Without this temperature increase, the gold solution easily aggregates. After another 48 h at 30° C the samples are brought to 0.3 M NaCl, again at 70-90° C. The colloidal solutions are then centrifuged twice at 14000 rpm for 10-60 min (depending on particle size) with an intermediate rinsing. Finally the pellet containing the DNA-modified particles, is redispersed in a 0.01M phosphate, 0.3 M NaCl, 0.01% Azide solution.

Particles of different DNA types were mixed together and the appropriate linker molecules added. After incubation for about 60 min at 20-60° C gel-loading buffer (dextrose) was added and the samples were allowed to cool down during gel loading. Agarose gel (0.8% w/v) electrophoresis was performed to separate the dimers [18] from single particles and from larger aggregates, see fig. 3(a).

The dimer bands were cut out from the gel using a scalpel and diced into smaller pieces. The dimers were retrieved from the gel slices using Nanosep MF centrifugal filters (Pall Corporation).

SEM studies were performed on the 50 nm gold particles using a LEO-1450 EP electron microscope.

4. Preliminary Results and Discussion

The ssDNA successfully attached to the particles. This is clear from the fact that the gold colloids are stable in high salt concentration. The formation of dimers, trimers and larger aggregates has been verified by gel-electrophoresis, see Figure 4(a) and SEM photos, Figure 4(b). The incubation temperature seems to have an important

effect. The only difference between the columns in Figure 4(a) is the incubation temperature. At lower temperatures the incubation time of 1 h seems to be inadequate to form aggregates. The samples heated to 51 ℃ and 60 ℃ show bands that may correspond to gold-particle clusters of single, dimer, trimer and even four-mer as well as a long "tail" of larger aggregates not found in 20 ℃ and 40 ℃ samples. Different incubation times, from 10 minutes to 1 hour, showed no difference in the electrophoresis separation.

We have performed other electrophoresis experiments where gold particles coated with two types of ssDNA also showed strong dependence on temperature. Below 46℃, the column did not separate but above 51 ℃ it showed the same separated bands that presumably depends on the cluster size. 51 ℃ is close to the calculated DNA melting temperature for our oligos. One interesting observation is that gold particles coated with only one type of ssDNA traveled furthest in electrophoresis gel compared to those coated with 4 ssDNA, whereas the gold particles without DNA coating did not move at all in the gel. Thus, the response of coated gold particles in gel to the applied electric voltage (100V) is due to electrostatic force of the coated DNA molecules only. The separated bands are due to the balancing of the electrostatic force that is proportional to the coated DNA density and the drag force that is proportional to the cluster size of gold particles.

The method of separating the dimers from the agarose gel by centrifugal filtration seems to be working. Single particles, dimers, and trimers have been detected by SEM, see Figure 4(b). There is still a relatively large portion of single particles in the extracted dimer and trimer samples. We hypothesise that this could be due to breaking up of the dimers and trimers during the gel-extraction by centrifugation. There are also larger aggregates in SEM photos at the edge of the sample area. This may be due to aggregation during the drying process of the SEM sample droplet.

If the linking DNA strands were ligated before electrophoresis the bond should be stronger and less single particles would be found as the dimers would not break up so easily. This would also make the dimers more resistant to elevated temperatures.

One problem with the current procedure is the low yield of dimers using the gel separation technique, this low yield is also to be expected for the assembled four-mers and eight-mers. Another difficulty is to control the temperature of the gel chamber. An optimum temperature for dimers and other smaller aggregates is probably just below the melting temperature for the oligos. A separation technique with a higher yield and precise thermal control will eventually be needed.

5. Conclusions

We argue that building blocks for programmable self-assembly needs to have at least four distinct, geometrically separated bonds. A principal scheme for the production of building blocks with well-directed bonds for programmable self-assembly using DNA-nanoparticles has been presented. The introduced procedure is a completely bottom-up approach and can be used to produce quite advanced PSA building blocks

like nanoparticle eight-mers with eight bonds. Initial experiments have not given any indications that the devised scheme would be unfeasible. On the contrary we are encouraged by the fact that this method seems to work with simple and proved methods. However, to get a higher yield from the process a new particle separation method might be needed.

References

[1] C. A. Mirkin, R. L. Letsinger, R. C. Mucic, and J. J. Storhoff, Nature **382**, 607 (1996).
[2] A. P. Alivisatos, K. P. Johnsson, X. G. Peng, T. E. Wilson, C. J. Loweth, M. P. Bruchez, and P. G. Schultz, Nature **382**, 609 (1996).
[3] J. J. Storhoff, R. Elghanian, R. C. Mucic, C. A. Mirkin, and R. L. Letsinger, J. Am. Chem. Soc. **120**, 1959 (1998).
[4] T. A. Taton, R. C. Mucic, C. A. Mirkin, and R. L. Letsinger, J. Am. Chem. Soc. **122**, 6305 (2000).
[5] E. Winfree, in *DNA Based Computers*, edited by R. J. Lipton and E. B. Baum (Amer. Math. Soc., Princeton, 1996), Vol. 27, p. 199.
[6] E. Winfree, J. Biomolecular Structure & Dynamics, 263 (2000).
[7] P. W. K. Rothemund and E. Winfree, in *Proc. of 32nd Ann. ACM Symp. Theory Comp*. (ACM, New York, NY and Portland, OR, USA, 2000), p. 459.
[8] Z. L. Zhang and S. C. Glotzer, Nano Letters **4**, 1407 (2004).
[9] B. Högberg and H. Olin, *Parameterization of Programmable Self-Assembly - Unique Structures and Bond Uniqueness*, Manuscript (2005).
[10] E. Klavins, IEEE Proc. Int. Conf. Robotics and Automation **5**, 5293 (2004).
[11] P. W. K. Rothemund, N. Papadakis, and E. Winfree, Plos Biology **2**, 2041 (2004).
[12] Z. Bao, L. Chen, M. Weldon, E. Chandross, O. Cherniavskaya, Y. Dai, and J. B.-H. Tok, Chemistry of Materials **14**, 24 (2002).
[13] R. C. Jin, G. S. Wu, Z. Li, C. A. Mirkin, and G. C. Schatz, J. Am. Chem. Soc. **125**, 1643 (2003).
[14] L. M. Demers, M. Ostblom, H. Zhang, N.-H. Jang, B. Liedberg, and C. A. Mirkin, J. Am. Chem. Soc. **124**, 11248 (2002).
[15] Csogor, M. Nacken, M. Sameti, C.-M. Lehr, and H. Schmidt, Mat. Sci. Eng.: C **23**, 93 (2003).
[16] Online tools for oligo-design. rnature.com/oligonucleotide.html, rna.tbi.univie.ac.at/cgi-bin/alifold.cgi, scitools.idtdna.com/Analyzer/
[17] R. Penchovsky and J. Ackermann, J. Computational Biology **10**, 215 (2003).
[18] D. Zanchet, C. M. Micheel, W. J. Parak, D. Gerion, S. C. Williams, and A. P. Alivisatos, J. Phys. Chem. B **106**, 11758 (2002).

Paper III

Study of DNA coated nanoparticles as possible self-assembly building blocks

Available online at www.sciencedirect.com
SCIENCE DIRECT
Applied Surface Science 252 (2006) 5538–5541
applied surface science
www.elsevier.com/locate/apsusc

Study of DNA coated nanoparticles as possible programmable self-assembly building blocks

Björn Högberg [a,*], Jing Helmersson [a,b], Svante Holm [c], Håkan Olin [a,*]

[a] Department of Engineering Physics, Mid Sweden University, Sweden
[b] Department of Physics and Astronomy, California State University Long Beach, California, USA
[c] Department of Natural Science, Mid Sweden University, Sweden

Available online 3 April 2006

Abstract

Nanoparticles coated with single stranded DNA have been shown to efficiently hybridize to targets of complementary DNA. This property might be used to implement programmable (or algorithmic) self-assembly to build nanoparticle structures. However, we argue that a DNA coated nanoparticle by itself cannot be used as a programmable self-assembly building block since it does not have directed bonds. A general scheme for assembling and purifying nanoparticle eight-mers with eight geometrically well-directed bonds is presented together with some preliminary experimental work.

PACS: 81.07.−b; 81.16.Dn

Keywords: Nanoparticles; Nanostructure; DNA coated nanoparticles; Programmable self-assembly; Gold particles

1. Introduction

DNA functionalized nanoparticles have been a prospect material for the construction of self-assembled structures since first demonstrated [1,2]. So far, the main interest in these particles has been to use them in detection methods for small amounts of specific DNA [3]. The structures formed in these, and other [4] experiments, are mainly periodic, or random agglomerations of particles. To fully exploit the potential of self-assembly of DNA coated nanoparticles one needs to address the problem of assembling non-periodic structures.

To implement programmable self-assembly (PSA) one needs building blocks of a certain complexity [5–9]. In short, the requirements are: (1) unspecific bonds, that can bind to several types of different building blocks, (2) an incoming building block must interact with at least two other, already assembled blocks and (3) at least four bonds on each block to avoid blocking of the assembly. For general discussions on criteria for PSA see for example [9–11]. Using a diamond like structure, it is possible to build three-dimensional structures using building blocks with only four bonds, but a more practical approach is to use triangular prisms or cubes having at least 5, respectively, 6 bonds (6, respectively 8 bonds, if the bonds are situated at the corners).

Building blocks that have a size of micrometers can be manufactured and functionalized using conventional micro-technology, one example of a such a top-down approach for the functionalization of two distinct areas of 1 μm particles can be found in [12]. However, with reducing dimensions, it is increasingly difficult to functionalize different parts of the building blocks with different functions making a top-down procedure no longer feasible. A new approach is needed. We here propose such a new method for the fabrication of nanoscale PSA-building blocks using only bottom-up methods.

2. Making PSA-building blocks from nanoparticles

Two sets of nanospheres of a suitable material are functionalized each with two different types of single stranded DNA (ssDNA). For example, for gold [13,14] and SiO_2 [15] nanoparticles, there are standard protocols for this. These spheres are then mixed in solution together with linker

* Corresponding authors. Fax: +46 60 14 88 75.
E-mail addresses: bjorn.hogberg@miun.se (B. Högberg), hakan.olin@miun.se (H. Olin).

0169-4332/$ – see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.apsusc.2005.12.127

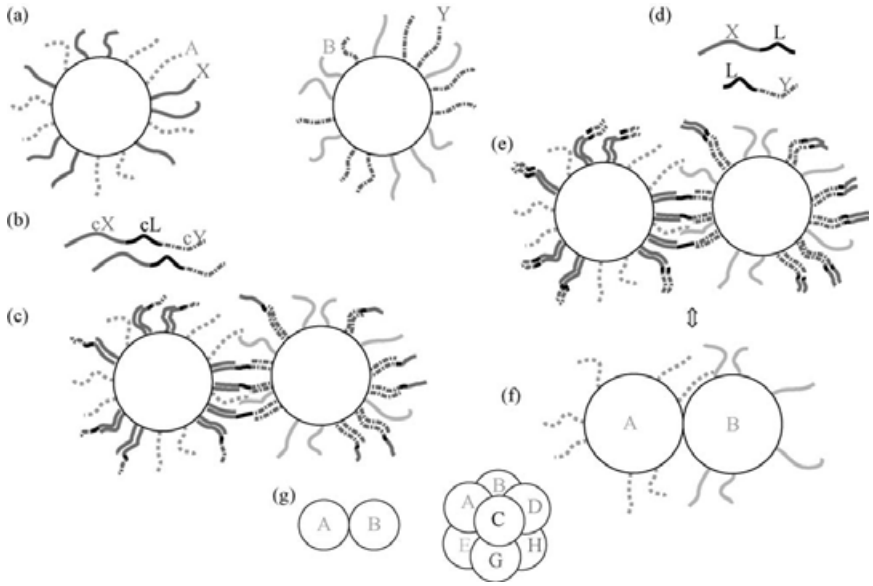


Fig. 1. Scheme for nanoparticle PSA-building block assembly. (a) Two different sets of nanospheres are each coated with two (or more) single stranded DNA sequences (A–X and B–Y). (b) Linker molecules. (c) Dimers can be extracted by mass separation. (d) Passivation of remaining sticky ends by complementary molecules (X–L and L–Y). (e) The only single stranded DNA left is of type A on one sphere and of type B on the other so the dimer in (e) is functionally equivalent to the one in (f). Using several iterations of this process will produce eight-mers (g) with eight separate and specific binding sites to be used for PSA.

molecules (Fig. 1a–e). By controlling the temperature of the solution we can promote the formation of dimers. After a while the linker DNA is rinsed away, stopping any further aggregation of the nanoparticles. The dimers are then separated from the rest of the aggregates by mass separation. The dimers produced in this way constitute simple PSA-building blocks with ssDNA of types A and B sticking out at different ends of the dimers (Fig. 1f and g). By starting out with particles coated with four specific DNA sequences and using several iterations of this technique one could make four-mers and eight-mers to be used as more advanced PSA-building blocks (Fig. 1g).

Following the above method one can create different basic building blocks with specific sticky ssDNA on different faces of the blocks. As the basic building blocks are mixed together with blocks having complementary DNA self-assembly will occur (Fig. 2).

Programmable self-assembly require simultaneous binding of two bonds to uniquely produce the desired structures. For example, in Fig. 2, the blocks 5 and 1 can connect to the 2 block. The right position is determined by matching two bonds. The A DNA's of both blocks 1 and 5 could, in principle, make a single bond to the cA DNA of block 2 but the alignment of the connected blocks would probably be wrong. This could be avoided by setting the temperature and/or the amount of stirring to values where blocks bonded by one bond are unstable structures whereas blocks bonded by two bonds are stable.

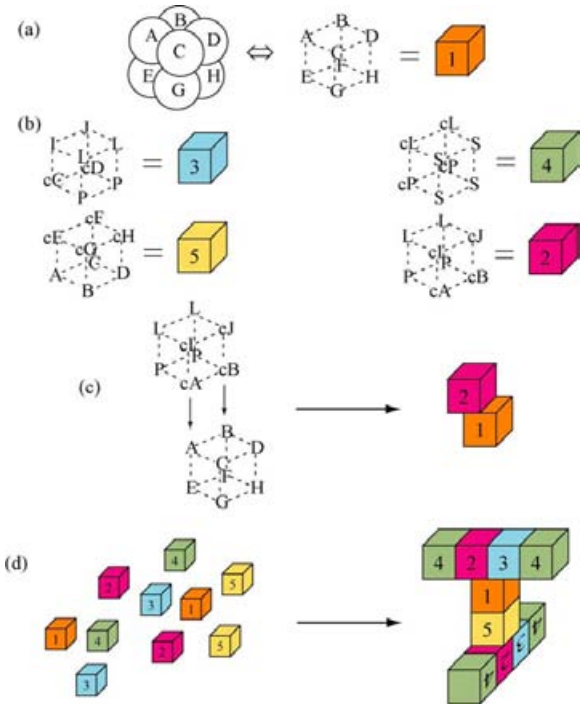


Fig. 2. Scheme of progammable self-assembly. (a) The eight-mers have eight specific ssDNA binding sites. (b) Four more eight-mers are depicted. (c) Example of binding: the cA and cB of block 2 binds to the A and B strands on block 1. (d) Structure formation. The "S" DNA of block 4 has no complement on the other blocks so it acts as a stop-block.

3. Materials and methods

Gold colloids with mean diameters ranging from 10 to 50 nm were purchased from G.Kisker GbR. Typical size

distributions were about 15%, concentrations are approximately 0.07 nM for the 50 nm up to about 7 nM for the 10 nm particles. Thiol modified (5′ and 3′) and unmodified ssDNA were purchased from Cybergene AB. All oligos were HPLC-purified by the manufacturer. The length of the thiolated oligonucleotides were 16 code bases and 10 bases of consecutive T's between the thiol group and the coding sequence. The 10 T's act as a separator between the particle and the hybridizing sequence. The coding sequences were optimized to have cross-hybridization probability as small as possible. The sequences were also checked for unwanted hairpin and dimer formation. To this end we used software developed by ourselves and some online tools [16]. We have primarily used a subset of the DNA library for DNA-computing by Penchovsky and Ackermann [17]. This set of oligos is well-optimized for uniqueness and the sequences are also optimized to have similar melting temperatures. The functionalization of the gold particles followed a modified version of the Storhoff protocol [3]. Equal amounts of four different thiolated oligos to a total of 1 nmole DNA are mixed separately and then mixed with 1.3 ml of the gold colloid. These samples are stored at 30 °C overnight followed by the addition of a sodium phosphate buffer to 0.01 M (pH 7) and a NaCl solution up to 0.1 M. The addition of salt is done drop wise and in two steps separated by roughly 2 h, at an elevated sample temperature, typically 70–90 °C. After another 48 h at 30 °C the samples are brought to 0.3 M NaCl, again at 70–90 °C. The colloidal solutions are then centrifuged twice at 14000 rpm for 10–60 min (depending on particle size) with an intermediate rinsing. Finally, the pellet containing the DNA-modified particles, is redispersed in a 0.01 M phosphate, 0.3 M NaCl, 0.01% Azide solution. Particles of different DNA types were mixed together and the appropriate linker molecules added. After incubation for about 60 min at 20–60 °C gel-loading buffer (dextrose) was added and the samples were allowed to cool down during gel-loading. Agarose gel (0.8%, w/v) electrophoresis was performed to separate the dimers [18] from single particles and from larger aggregates, see Fig. 3a. The dimer bands were cut out from the gel using a scalpel and diced into smaller pieces. The dimers were retrieved from the gel slices using Nanosep MF centrifugal filters (Pall Corporation). SEM imagining were done using a LEO-1450 EP.

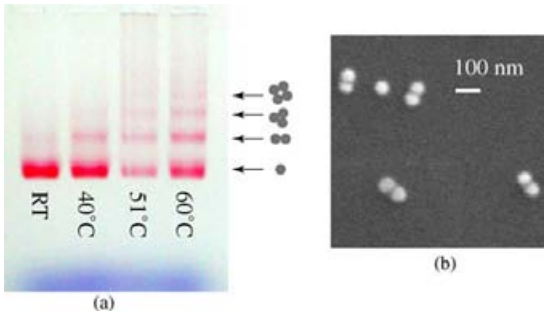


Fig. 3. (a) Photograph of the discrete lines formed by agarose gel electrophoresis of 50 nm DNA coated nanoparticles with linker molecule added. Bands of increasingly bigger aggregates are visible. The temperatures below the bands refer to the incubation temperature. (b) Electron micrograph of electrophoresis-purified nanoparticle dimers.

4. Results and discussion

The ssDNA successfully attached to the particles. This was clear from the stability of the gold colloids in high salt concentration. The formation of dimers, trimers and larger aggregates was verified by gel-electrophoresis and SEM imaging (Fig. 3a and b). The incubation temperature seemed to have an important effect. The only difference between the columns in Fig. 3a was the incubation temperature. At lower temperatures the incubation time of 1 h seemed to be inadequate to form aggregates. The samples heated to 51 and 60 °C showed bands that may correspond to gold particle clusters of single, dimer, trimer and even four-mer as well as a long ''tail'' of larger aggregates not found in 20 and 40 °C samples. Different incubation times, from 10 min to 1 h, showed no difference in the electrophoresis separation.

We have performed other electrophoresis experiments where gold particles coated with two types of ssDNA also showed strong dependence on temperature. Below 46 °C, the column did not separate but above 51 °C it showed the same separated bands that presumably depends on the cluster size. Fifty-one degree Celsius is close to the calculated DNA melting temperature for our oligos. One interesting observation was that gold particles coated with only one type of ssDNA traveled furthest in electrophoresis gel compared to those coated with 4 ssDNA, whereas the gold particles without DNA coating did not move at all in the gel. Thus, the response of coated gold particles in gel to the applied electric voltage (100 V) is due to electrostatic force of the coated DNA molecules only. The separated bands are due to the balancing of the electrostatic force that is proportional to the coated DNA density and the drag force that is proportional to the cluster size of gold particles.

The method of separating the dimers from the agarose gel by centrifugal filtration was seemed to be working as single particles, dimers, and trimers were detected by SEM (Fig. 3b). There was still a relatively large portion of single particles in the extracted dimer and trimer samples. This might be due to breaking up of the dimers and trimers during the gel-extraction by centrifugation. There are also larger aggregates seen in the SEM micrographs at the edge of the sample area. This may be due to aggregation during the drying process of the SEM sample droplet. If the linking DNA strands were ligated before electrophoresis the bond should be stronger and less single particles would be found as the dimers would not break up so easily. This would also make the dimers more resistant to elevated temperatures.

One problem with the current procedure is the low yield of dimers using the gel separation technique, this low yield is also to be expected for the assembled four-mers and eight-mers. Another difficulty is to control the temperature of the gel chamber. An optimum temperature for dimers and other smaller aggregates is probably just below the melting temperature for

the oligos. A separation technique with a higher yield and precise thermal control will eventually be needed.

5. Conclusions

A completely bottom-up scheme for the production of building blocks with well-directed bonds for programmable self-assembly using DNA-nanoparticles has been presented. Initial experiments have not given any indications that the devised scheme would be unfeasible, on the contrary we are encouraged by the fact that this method seems to work with simple and proved methods. However, to get a higher yield from the process a new particle separation method might be needed.

References

[1] C.A. Mirkin, R.L. Letsinger, R.C. Mucic, J.J. Storhoff, Nature 382 (1996) 607.
[2] A.P. Alivisatos, K.P. Johnsson, X.G. Peng, T.E. Wilson, C.J. Loweth, M.P. Bruchez, P.G. Schultz, Nature 382 (1996) 609.
[3] J.J. Storhoff, R. Elghanian, R.C. Mucic, C.A. Mirkin, R.L. Letsinger, J. Am. Chem. Soc. 120 (1998) 1959.
[4] T.A. Taton, R.C. Mucic, C.A. Mirkin, R.L. Letsinger, J. Am. Chem. Soc. 122 (2000) 6305.
[5] E. Winfree, in: R.J. Lipton, E.B. Baum (Eds.), DNA Based Computers, vol. 27, Amer. Math. Soc., Princeton, 1996, p. 199.
[6] E. Winfree, J. Biomol. Struct. Dyn. 263 (2000).
[7] P.W.K. Rothemund, E. Winfree, Proc. of 32nd Ann. ACM Symp. Theory Comp., ACM, New York, NY/Portland, OR, USA, 2000, p. 459.
[8] Z.L. Zhang, S.C. Glotzer, Nano Lett. 4 (2004) 1407.
[9] B. Högberg, H. Olin, Programmable Self-Assembly—Unique Structures and Bond Uniqueness, J. Comput. Theor. Nanosci., in press.
[10] E. Klavins, IEEE Proc. Int. Conf. Robotics and Automation, vol. 5, 2004, p. 5293.
[11] P.W.K. Rothemund, N. Papadakis, E. Winfree, Plos Biol. 2 (2004) 2041.
[12] Z. Bao, L. Chen, M. Weldon, E. Chandross, O. Cherniavskaya, Y. Dai, J.B.-H. Tok, Chem. Mater. 14 (2002) 24.
[13] R.C. Jin, G.S. Wu, Z. Li, C.A. Mirkin, G.C. Schatz, J. Am. Chem. Soc. 125 (2003) 1643.
[14] L.M. Demers, M. Ostblom, H. Zhang, N.-H. Jang, B. Liedberg, C.A. Mirkin, J. Am. Chem. Soc. 124 (2002) 11248.
[15] Csogor, M. Nacken, M. Sameti, C.-M. Lehr, H. Schmidt, Mater. Sci. Eng. C 23 (2003) 93.
[16] Online Tools for Oligo-Design. rnature.com/oligonucleotide.html, rna.t-bi.univie.ac.at/cgi-bin/alifold.cgi, scitools.idtdna.com/Analyzer/.
[17] R. Penchovsky, J. Ackermann, J. Comput. Biol. 10 (2003) 215.
[18] D. Zanchet, C.M. Micheel, W.J. Parak, D. Gerion, S.C. Williams, A.P. Alivisatos, J. Phys. Chem. B 106 (2002) 11758.

Paper IV

Anisotropically DNA-Functionalized Nanoparticle Dimers

EPJ manuscript No.
(will be inserted by the editor)

Anisotropically DNA-Functionalized Nanoparticle Dimers

Björn Högberg and Håkan Olin[a]

Department of Engineering Physics and Mathematics, Mid Sweden University, SE-851 70 Sundsvall, Sweden

Received: date / Revised version: date

Abstract. Self-assembly of complex, non-periodic nanostructures can only be achieved by using anisotropic building-blocks. The building blocks need to have at least four bonds pointing in separate directions [1]. We have previously presented a method for the synthesis of such building-blocks using DNA-functionalized gold nanoparticles [2,3]. Here, we report on the progress in the experimental realization of this scheme. The first goal, in a process to make programmable self-assembly building-blocks using nanoparticles, is the production of dimers with different DNA-functions on the two component particles. We report on the fabrication of anisotropically functionalized dimers of nanoparticles of two different sizes. As a result of their anisotropy, these demonstrator building blocks can be made to assemble into spherical structures.

PACS. 81.16.Dn Self-assembly – 81.16.Rf Nanoscale pattern formation – 82.39.Pj Nucleic acids, DNA and RNA bases

1 Introduction

A technology for the production of self-assembled nanostructures is generally regarded as an important step towards better electronics, sensors and medical technology. [4] As always in these applications, the required geometrical structures are rather complex and often far from periodic in nature. Whereas periodic self assembly in the form of monolayers, particle crystals etc has been thoroughly investigated in the past, the assembly of complex, aperiodic, structures has only recently begun to attract attention. The most complex self-assembled nanostructures produced to date are made from DNA. [5,6] Although impressive in their structural complexity, it is generally accepted that the functionality of structures made from DNA alone is rather limited. To produce functional nanodevices metals and semiconductors will probably be needed. DNA can easily be attached to gold nanoparticles[7,8] and DNA-coated gold particles have been successfully attached to pure DNA-nanostructures. [9] Our aim is to directly assemble structurally complex particle structures, without the need for a pre-fabricated DNA scaffold. In order to produce non-periodic structures from self-assembly of simple building blocks, theese need to be of a certain minimal complexity. In particular, they need to have the possibility to form at least four different bonds and they need to be anisotropic so that the bonds are well directed in separate directions.[1] Some attemps to produce anisotropically functionalized particles have been made, see for example [10]. However these particles are large (1 μm) and it is hard to imagine how similar methods could be used on nanoscale particles.

[a] hakan.olin@miun.se

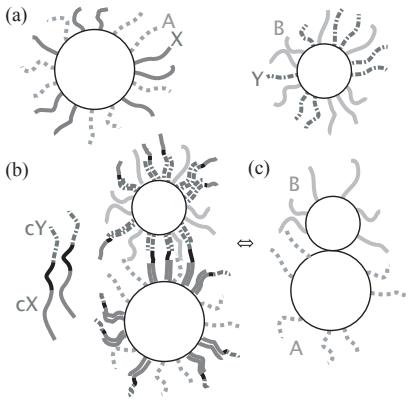


Fig. 1. (a) Two batches of nanoparticles are uniformly coated with a mix of two types of DNA, A, X and B, Y. (b) The two nanoparticle types are mixed with linker DNA, cX-cY, that binds two DNAs together (the X and Y DNAs), forming a nanoparticle dimer. The nanoparticle dimer has different kinds of unused DNA left on the different particles and is thus functionally equivalent to the anisotropic dimer shown in (c).

2 Ideas and experimental design

Instead of trying to anisotropically functionalize single particles, our approach is to make building blocks consisting of several particles. By making a heterodimer, for example, a building block is obtained which provides two different bonding options, one on each particle. The idea is briefly presented in fig. 1. This scheme can in principle be extended to produce more advanced building blocks like tetramers or octamers, see [2,3]. Note that when linker

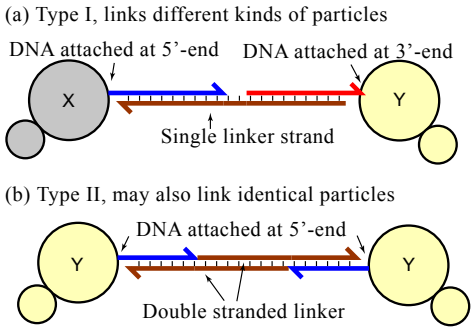


Fig. 2. The two ways of attaching a DNA strand to a particle, either at its 3'-end or its 5'-end, defines to types of possible ways of linking the particles with DNA.

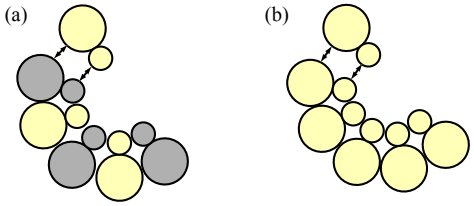


Fig. 3. Self-assembly of anisotropic dimers. If the small particles binds to small particles and large particles binds to large particles, the dimers should self-assemble into curved structures (or spherical 3-dimensional structures). In type I experiments the interacting dimers must be of two different types (a) whereas in type II experiments the dimers can all be identical (b).

DNA is added, the formation of dimers is not the only possible reaction. One would expect a number of larger aggregates to form as well. By adding equimolar amounts of linker, particles of type one, and particles of type two, we can promote dimer formation. However, purification by gel electrophoresis is still necessary to obtain pure heterodimer samples.

After the production and purification of dimers we proceed to self-assembly using the dimers as building blocks. Two types of DNA linkage has been tested:

2.1 Type I

In type I experiments we used the same type of DNA linking during assembly as in the case of the dimer formation. Here, the gold-attached DNA strands that participated in the binding reaction had opposite orientations. One strand was attached to its particle by the DNA 5'-end and the other to its particle by the 3'-end. The linker was single stranded and bound with its 5'-end to the 3'-attached particle and the 3'-end of the linker strand bound to the 5'-attached particle. See fig. 2(a).

2.2 Type II

In type II experiments the particles were linked by a double stranded DNA linker with protruding sticky ends. Here the strands attached to the particles must have the same orientation, *i.e.* both were attached at the 5'-end. See fig. 2(b). In type II linkage, the participating particle types can be identical whereas in type I linkage the particles must be of different types since the strands attached to the particles must have different orientations.

2.3 Proving anisotropy of the dimers

As explained above, the dimers should behave like a particle with two bond types, one on each constituent particle. By mixing the dimers with linkers that make the small particles stick to other small particles and large particles stick to large particles, the dimers should self-asssemble into curved structures like the ones shown in fig. 3. Other structures are also possible by multiple linkages between small or large particles. However, cooperative binding of several bonds, both on the large and the small particles, should promote formation of the curved structures. For discussions on cooperative bonding in programmable self-assembly, see [5,1] and references therein.

3 Materials and Methods

3.1 Nanoparticles

Gold nanoparticles of size 10 and 20 nm were bought from G.Kisker GbR, Germany. The red gold suspensions (100 ml) were mixed with 4,4'-(Phenylphosphinidene)bis-(benzenesulfonic acid) dipotassium salt (about 10 mg), and stirred overnight. This step created a ligand shell around the particles [11], preventing them to precipitate when exposed to salt. It also facilitated the electrophoresis by giving the particles a negative charge. [12]

3.2 DNA-samples

Synthetic oligonucleotides were bought from Cybergene AB, Sweden. The strands for attachment on gold particles were equipped with a thiol modification at the 3' or 5' end.

3.3 DNA-Gold Conjugates

The ligand protected nanoparticles were concentrated by salt precipitation followed by centrifugation. An excess (compared to reported maximum surface coverage [13]), about 50 DNA strands/particle, of thiolated DNA were added to the particle suspension. The solution was brought to 0.1 M NaCl and 10 mM Phosphate buffer, pH 7, and left at room temperature for two days. After conjugation, excess DNA was washed away by repeated centrifugation and re-suspension of the pellet in 0.3 M NaCl, 10 mM phos. buffer. The amount of particles was assesed by absorbance spectroscopy at λ=520 nm on a Varian Cary 50 UV/Vis spectrophotometer.

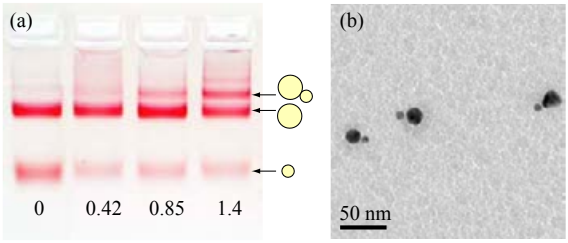


Fig. 4. Electrophoretic separation of 10 nm and 20 nm-dimers. The numbers below the lanes in (a) indicate how many linker strands per small-large particle pairs that are added to the solution. The dimer band is strongest when about 1.4 linkers per pair are added. (b) A TEM micrograph from a dimer sample extracted from the gel.

3.4 Dimer Formation and Purification

To prepare 10 nm and 20 nm dimers, an equimolar mix (in terms of number of particles) of the two size DNA-conjugated particles, were mixed with linker DNA. The formation of dimers was verified by agarose gel-electrophoresis in TBE buffer and the dimer bands were electroeluted to a small piece of glass-fiber filter paper. The filter paper was brought to a centrifugal filter and the dimer solution collected by centrifugation.

3.5 Self-Assembly

NaCl (aq., 5 M) was added to the suspensions of dimers (two kinds of dimers in type I experiments, and one kind in type II) to give a 0.3 M NaCl concentration. The resulting suspensions were mixed with appropriate DNA-linker strands, heated to around 45°C, and allowed to cool slowly under 2 h.

3.6 TEM-sample preparation

A drop of gold suspension was placed on a silicon monoxide / formvar coated copper grid (Ted Pella Inc.), allowed to adsorb to the surface for about 10 minutes and then dried by wicking from the side with a piece of filter paper.

4 Results and Discussion

The formation of dimers of 10 and 20 nm gold particles was verified by gel electrophoresis. As shown in fig. 4(a) the samples containing linker-DNA showed additional bands corresponding to dimers and larger aggregates. The most efficient way to produce dimers was to add slightly more than one linker par pair of 10 nm and 20 nm particles. We normally used 1.4 strands per pair. The dimers were always made using type I linkage.

We tried a few methods for retrieval of the dimers from agarose gel. When we inserted a glass fiber filter backed by a piece of dialysis membrane in front of the desired

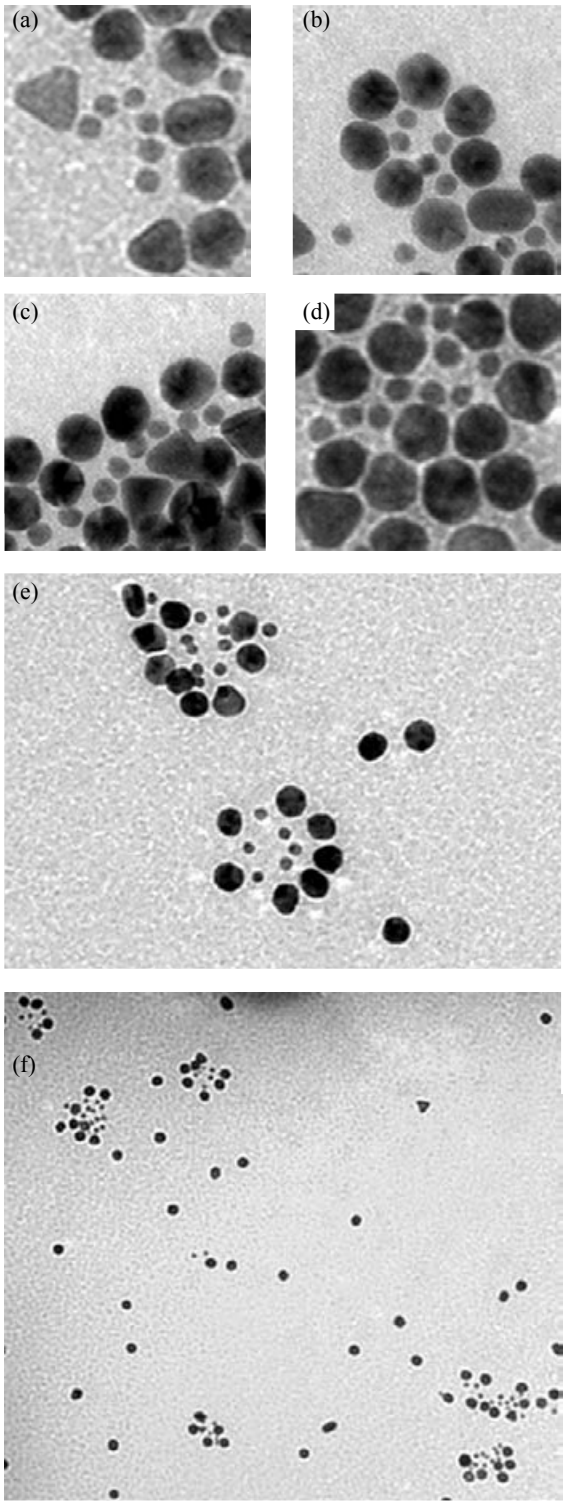


Fig. 5. TEM images of samples resulting from dimer self-assembly. (a)-(d) Type I linkage, two types of dimers. (e)-(f) Type II linkage, one type of dimers.

band and then electroeluted the sample onto the filter, the largest amount of dimers was collected. We estimate that only about 50% of all particles form dimers and that about 50% of the dimers are lost during gel extraction, leading to a total yeild of about 25%. Fig. 4(b) shows a TEM image from an extracted dimer sample. Most of the particles were found in dimers, however, a substantial amount (about 5-10%) of the particles were singles. We hypothesize that many of the singlets are formed from dimers breaking up during gel-extraction.

Initially, this work was performed using only type I linking, both for dimer formation and assembly of the dimers. The results are shown in figs. 5(a)-(d). The yield of structures were quite low, many particles were found in larger aggregates without ordered structure. Structures, like the ones found in 5(c)-(d) were unexpected. Maybe an excess of single particles could explain this sort of linear structures.

Two problems were identified in these type I experiments: stochiometry and temperature sensitivity. Getting the right stochiometry between the two types of dimers used, was very difficult. Because the amount of extracted dimer sample was so low, the quantization readings by spectroscopy were of poor quality. Thus, the preferred (1:1) stochiometry of the two dimer types could not be ensured. Furthermore, to get good self-assembly we wanted to keep the sample at a temperature where the dimer-dimer bonds formed. In the initial experiments, figs. 5(a)-(d), the intra-dimer bonds were as strong as the bonds between dimers, *i.e.* the sticky ends of the DNAs that formed the bonds were of the same length. The bonds keeping the dimers together, thus had the same melting temperature as the bonds making up the dimer-dimer structures. In order to remedy this, we designed a type II experiment where only one type of dimer where involved, thus avoiding the problem of stochiometry. Furthermore the type II experiment was designed so that the DNA sticky ends involved in the dimer-dimer assembly were much shorter than the ones keeping the dimers together. This way we could stay at the melting temperature of structure formation without risking dimer break-up.

The results of the type II experiment is shown in figs. 5(e)-(f). Here, a majority of all particles were found in curved structures. The particles not in curved structures were almost all single particles, so one can conclude that almost all of the correctly preserved dimers were successfully assembled into curved structures. This in all gives a clear indication that the observed structures are indeed results of anisotropic building-blocks.

To test if the structures could be tuned by changing the type of linkers added, we made a sample where no large particle-large particle linker was added, only the small-small linker. Since the large particles do not participate in the self-assembly, only as luggage for the small particles, the structures formed should be more irregular. The result is shown in fig. 6. As expected, this sample showed structures much more disordered than the previous ones but still with a clear effect of aggregation of small particles, surrounded by larger ones.

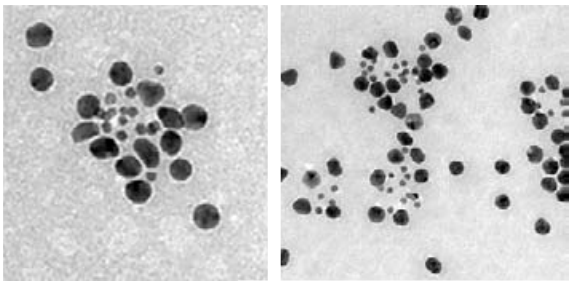


Fig. 6. TEM images of a sample where the large-large linkers have been excluded.

5 Conclusions

A method for the production of nanoparticle building blocks for DNA-mediated self-assembly is presented. As anisotropic building blocks are crucial for the production of complex patterns by self-assembly we have focused on the production such building blocks: anisotropic dimers. The fact that the dimers self-assemble into small curved aggregates demonstrates that the dimers are in fact anisotropic building blocks.

References

1. B. Högberg and H. Olin. *J. Comput. Theor. Nanosci.*, 3(3):391–397, 2006.
2. B. Högberg and H. Olin. In *Proc. 2nd Conf. Found. Nanosci.*, pages 219–226, Snowbird, Utah, 2005. ScienceTechnica.
3. B. Högberg, J. Liu-Helmersson, S. Holm, and H. Olin. *Applied Surface Science, in press (doi:10.1016/j.apsusc.2005.12.127)*, 2006.
4. R. Compañó, editor. *Technology Roadmap for Nanoelectronics*. European Commission. Office for Official Publications of the European Communities, Luxemburg, 2001.
5. P. W. K. Rothemund, N. Papadakis, and E. Winfree. *Plos Biology*, 2(12):2041–2053, 2004.
6. P. W. K. Rothemund. *Nature*, 440(7082):297–302, 2006.
7. A. P. Alivisatos, K. P. Johnsson, X. G. Peng, T. E. Wilson, C. J. Loweth, M. P. Bruchez, and P. G. Schultz. *Nature*, 382(6592):609–611, 1996.
8. C. A. Mirkin, R. L. Letsinger, R. C. Mucic, and J. J. Storhoff. *Nature*, 382(6592):607–609, 1996.
9. John D. Le, Yariv Pinto, Nadrian C. Seeman, Karin Musier-Forsyth, T. Andrew Taton, and Richard A. Kiehl. *Nano Letters*, 4(12):2343–2347, 2004.
10. Z. Bao, L. Chen, M. Weldon, E. Chandross, O. Cherniavskaya, Y. Dai, and J.B.-H. Tok. *Chemistry of Materials*, 14(1):24, 2002.
11. G. Schmid and A. Lehnert. *Angew. Chemie - I. E.*, 28(6):780–81, 1989.
12. D. Zanchet, C. M. Micheel, W. J. Parak, D. Gerion, S. C. Williams, and A. P. Alivisatos. *J. Phys. Chem. B*, 106(45):11758–11763, 2002.
13. L. M. Demers, C. A. Mirkin, R. C. Mucic, R. A. Reynolds, R. L. Letsinger, R. Elghanian, and G. Viswanadham. *Analytical Chemistry*, 72(22):5535–5541, 2000.

Paper V

DNA-scaffolded nanoparticle structures

DNA-scaffolded nanoparticle structures

Björn Högberg and Håkan Olin

Department of Engineering Physics and Mathematics, Mid Sweden University, SE-851 70 Sundsvall, Sweden

E-mail: bjorn.hogberg@miun.se

Abstract. DNA self-assembly is a powerful route to the production of very small, complex structures. When used in combination with nanoparticles it is likely to become a key technology in the production of nanoelectronics in the future. Previously, demonstrated nanoparticle assemblies have mainly been periodic and highly symmetric arrays, unsuited as building blocks for any complex circuits. With the invention of DNA-scaffolded origami reported earlier this year [1], a new route to complex nanostructures using DNA has been opened. Here, we give a short review of the field and present the current status of our experiments were DNA origami is used in conjunction with nanoparticles. Gold nanoparticles are functionalized with thiolated single stranded DNA. Strands that are complementary to the gold particle strands can be positioned on the self-assembled DNA-structure in arbitrary patterns. This property should allow an accurate positioning of the particles by letting them hybridize on the lattice. We report on our recent experiments on this system and discuss open problems and future applications.

1. Introduction

DNA nanotechnology [2] is the art of building nanoscale objects using DNA. This field has evolved rapidly in recent years, from periodic DNA-lattices [3] via algorithmic assemblies [4] to recent experiments where each strand has its pre-determined position in a lattice (*uniquely addressed* assemblies) [1, 5]. DNA nanotechnology has also been used to produce three dimensional objects like a 22 nm truncated octahedron [6]. To produce useful objects for electronics however, this range of complicated nanoscale objects will need integration with metals and semiconductors. DNA can easily be attached to gold nanoparticles[7, 8]. And a few groups have now started to focus on the attachment of functionalized nanoparticles on the pure DNA structures [9, 10]. The work presented in [9] and [10] deals with nanoparticles on DNA *tilings*, where each tile is about 15-20 nm. This distance is the same as the spatial resolution for patterns on these type of assemblies. The newly invented DNA-origami [1] has a much smaller spatial resolution on the order of 5 nm. Our work focus on the attachment of nanoparticles on DNA origami.

2. DNA origami

We have used a "tall rectangle" origami, designed by Rothemund [1] as a test lattice for the attachment of nanoparticles and proteins. The origami is self-assembled by letting a long circular DNA strand hybridize with 225 shorter *staple* strands, the principle is described in fig. 1. The long circular strand was DNA from a M13mp18 virus and was bought from Sigma Aldrich, Germany. The short staple strands were synthetic, cartridge-purified, oligos, bought from DNA

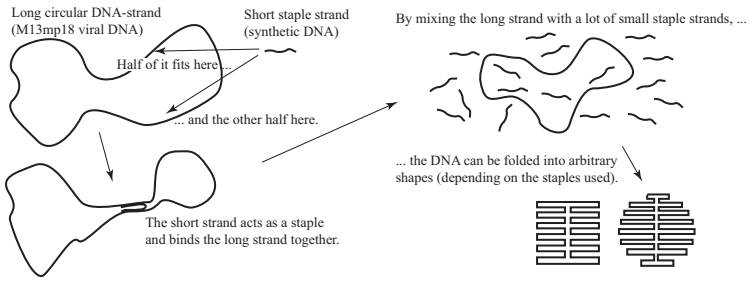


Figure 1. The principle behind DNA-scaffolded origami [1]. Since each staple has its predetermined position in the finished assembly, the staples can be used to address specific locations in the lattice (like in fig. 3(a)).

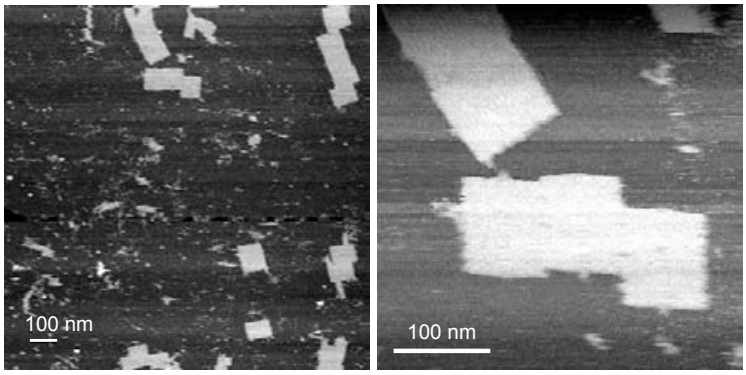


Figure 2. AFM images of the square, $\sim 70 \times 90$ nm, DNA origami's used in our experiments. AFM performed on mica under liquid.

Technology, Denmark. The origami's were self-assembled by mixing 0.05 pmole of the M13mp18 scaffold DNA with 5 pmole of each of the 225 staple strands in a 50 μl volume of Tris-HCl buffer, pH 8, 10 mM, with 1 mM EDTA and 12.5 mM $MgCl_2$. Final concentration of viral scaffold was 0.001 pmol/μl. The solution was heated and allowed to cool slowly (about 1h) in a PCR-thermocycler from 95 to 20°C. Samples for atomic force microscopy (AFM) were produced by placing a drop of the solution on a piece of freshly cleaved mica. Liquid AFM was performed directly under buffer or under isopropanol after rinsing with distilled water, the result is shown in fig. 2.

3. Attaching proteins to the origami

To test the addressability of the DNA origami we exchanged some of the staples with oligos modified with a biotin molecule at the 5'-end. These oligos were purchased from Cybergene, Sweden. The biotin-oligos were exchanged for some of the staples to produce a pattern of a walking man, see fig. 3(a). After assembly of the origami's, streptavidin protein (Sigma Aldrich), was added to the solution and incubated overnight at 4°C. Streptavidin binds to biotin and creates a contrast in the AFM because of the $\sim$2 nm diameter of the streptavidin protein. Since the biotin staples were present in a large excess over the origamis, streptavidin must also be added in a large excess, this results in the high streptavidin background that can be seen in the AFM micrograph (fig. 3(b))

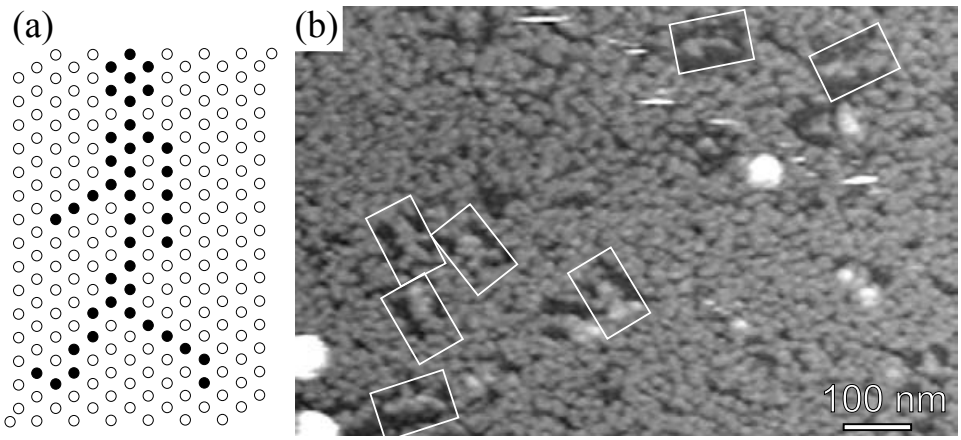


Figure 3. (a) The pattern of staples in the rectangle origami. The black staples were exchanged with biotin-modified DNA. (b) AFM image of biotin modified origami's where streptavidin proteins have been attached to produce the walking man pattern. White rectangles have been drawn around some of the walking men as a guide to the eye. AFM performed in isopropanol.

4. DNA functionalized gold nanoparticles

In order to attach nanoparticles to the DNA lattice, some of the staples are exchanged for staples that have prolonged sequences at the 5'-end. The extended part of the staple sequence is CTCTCCTTCCCTTT, and the nanoparticles are functionalized with the complementary strand AAAGGGAAGGAGAG. So instead of a biotin group in the case of the biotin-streptavidin experiment above, the origami's now contain some staples with sticky ends of DNA protruding from the lattice for attachment of functionalized nanoparticles.

Three types of DNA-functionalized gold nanoparticles were used, schematically depicted in fig. 4. We used particles that either had only one coding DNA strand per particle, like type I(a) and I(b) in fig. 4. With coding strand, we mean that the sequence of that strand is the complement to the sticky ends protruding from the origami lattice. We hypothesized that the advantage of type I particles would be that they could be mixed directly with the scaffold- and staple-DNA in a one-pot experiment. Whereas the particles of type II, with a coverage of many coding DNA strands, a one-pot experiment would probably fail, since each of the DNA strands of the particle would bind a staple and thus prohibit correct assembly of the origami. The type II particles with many coding strands were used exclusively in two-pot experiments where the origami first is produced and the particles added subsequently. The addition of a shell of non-coding DNAs as in type I(b) has the effect of stabilizing the particles in Mg^{2+} solutions, see below.

4.1. Nanoparticles

Gold nanoparticles of size 5 nm were bought from G.Kisker GbR, Germany. The red gold suspensions (100 ml) were mixed with 4,4'-(Phenylphosphinidene)bis(benzenesulfonic acid) dipotassium salt (about 10 mg), and stirred overnight. This step is created a ligand shell around the particles [11], preventing them to precipitate when exposed to salt. [12]

4.2. Thiolated DNA strands

Synthetic, modified oligonucleotides were bought from Cybergene AB, Sweden. The strands for attachment on gold particles were equipped with a thiol modification at the 3' end.

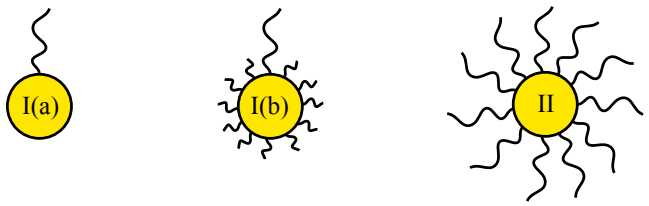


Figure 4. The three types of DNA-functionalized nanoparticles used. I(a) a single DNA strand attached to each particle. I(b) a single long, coding, DNA strand attached and many small, non-coding AAAAA sequences to prohibit particle aggregation. Type II nanoparticles uses a homogenous coverage of coding DNA strands.

4.3. DNA-Gold Conjugates

The ligand protected nanoparticles were concentrated by salt precipitation followed by centrifugation. In the production of type I(a) particles, a one-to-one ratio of particles and thiolated DNA was mixed, promoting the formation of particles conjugated to exactly one DNA strand. To produce type I(b) particles we took type I(a) particles and added a large excess of 3'-thiolated AAAAA strands. In the case of the type II particles, a large excess (about 50 DNA strands/particle) of thiolated DNA were added to the original ligand-particle suspension. In all cases the solution was brought to 0.1 M NaCl and 10 mM Phosphate buffer, pH 7, and left at room temperature for two days. After conjugation, excess DNA was washed away by repeated centrifugation and re-suspension of the pellet in 0.3 M NaCl, 10 mM phos. buffer. The amount of particles was assessed by spectroscopy at 520 nm.

4.4. One-pot experiments

In one-pot experiments, particles of type I(a) or (b) were mixed with staples and the viral, scaffold DNA in the following fashion: Twice as many functionalized particles were added as "pattern"-staples, *i.e.* the staples that were prolonged with a coding sequence at the 5'-end for attachment of nanoparticles. And the amount of each of the staples were about 20-50 times the amount of M13mp18 viral DNA. The final concentration of scaffold DNA was generally around $3{\cdot}10^{-4}$ pmol/μl, about 3 times lower than in the pure DNA experiment.

We found that the type I(a) particles were stable in NaCl but unstable in $MgCl_2$ solutions. To try the ability of forming origami's in NaCl we made a few runs with pure DNA origami's like in sect. 2 but with the Mg-ions substituted for a Na-ions at different concentrations (0.1-0.5 M). Non of the experiments yielded as clear, well-formed, origami's as with Mg-ions, (fig. 2). We therefore abandoned the experiments with particles of type I(a) in favor of particles of type I(b) where a shell of non-coding DNAs keeps the nanoparticles from precipitating in $MgCl_2$ [13]. We found that the type I(b) particles were stable up to about 4.5 mM $MgCl_2$ and 0.5 M NaCl. To try the efficiency of origami formation, DNA origami's were made in a solution of 4.5 mM $MgCl_2$, 0.5 M NaCl, 10 mM Tris-HCl (pH7) and 1 mM EDTA. These origami's were as reproducible and as well formed as the ones made in 12.5 mM $MgCl_2$, so a one-pot experiment with type I(b)-particles was expected to produce the desired nanoparticle patterned origami's. However, when type I(b) particles were added to the mixture, no traces of patterns or origami's could be found by AFM or TEM.

4.5. Two-pot experiments

The two-pot experiments were conducted by first assembling the DNA-origami's, like described in sect. 2, fixing them on a piece of freshly cleaved mica, then rinsing the mica with distilled

water, and subsequently adding a few drops of a suspension of particles of type II. As with the one-pot experiments, initial results are discouraging. We observed no clear patterns of particles.

5. Conclusions and Open Problems

As shown in the original experiment by Rothemund [1], and in our trials with streptavidin proteins it is straightforward to address different locations on a DNA-origami and make complex nanoscale patterns. However, judging from our initial experiences, it seems more difficult to attach non-biological components with the same accuracy. One reason why our experiments with nanoparticles fail to produce the desired patterns might be that the effective diameter of the nanoparticles is much greater than 5 nm. In reports dealing with periodic patterns of particles on DNA lattices [9, 10, 13], the smallest distance between two 5 nm particles is normally about 20 nm, even if the underlying lattice has more closely spaced hybridization points as in [9] and [10]. Furthermore, the apparent width of a 5 nm particle as measured in AFM is also around 20 nm. If the effective diameter of a functionalized 5 nm particle is in fact around 20 nm, then it is clear that it would be hard to prove the assembly of any complex patterns on a $\sim 70 \times 90$ nm large rectangle. We have tried to assemble nanoparticles in sparse patterns on the origami rectangles, using only a few particles one each origami. Since the rectangles end up in a non-periodic way on the mica, and the DNA lattice becomes undetectable in AFM when particles are numerous on the surface, we have not been able to prove that the particles seen in AFM really rests on an origami. The use of smaller nanoparticles needs to be tested.

Things are further complicated by the fact that the bond that attaches the particle to the origami is a rather floppy chain of DNA, about 5 nm in length (the coding sequence is 14 base-pairs long). This makes a correctly hybridized particle mobile within a 5 nm radius and could produce distortions in the desired pattern. One could try a shorter coding strand but this would make the bond more unstable. One could also try a periodic coding sequence, like a sequence of T's alone as is done in [10], this would give the particles a bit more freedom to move along the coding strand and may make it easier to closely pack particles. However, a periodic nucleotide sequence would make it harder to extend the technology to involve different coding sequences for different types of particles.

References

[1] P. W. K. Rothemund. *Nature*, 440(7082):297–302, 2006.

[2] N. C. Seeman. *Materials Today*, 6(1):24–29, 2003.

[3] E. Winfree, F. R. Liu, L. A. Wenzler, and N. C. Seeman. *Nature*, 394(6693):539–544, 1998.

[4] P. W. K. Rothemund, N. Papadakis, and E. Winfree. *Plos Biology*, 2(12):2041–2053, 2004.

[5] S. H. Park, C. Pistol, S. J. Ahn, J. H. Reif, A. R. Lebeck, C. Dwyer, and T. H. LaBean. *Angew. Chemie Int. Ed.*, 45(5):735–739, 2006.

[6] M.M. Shih, J.D. Quispe, and G.F. Joyce. *Nature*, 427(6975):618, 2004.

[7] A. P. Alivisatos, K. P. Johnsson, X. G. Peng, T. E. Wilson, C. J. Loweth, M. P. Bruchez, and P. G. Schultz. *Nature*, 382(6592):609–611, 1996.

[8] C. A. Mirkin, R. L. Letsinger, R. C. Mucic, and J. J. Storhoff. *Nature*, 382(6592):607–609, 1996.

[9] J. P. Zhang, Y. Liu, Y. G. Ke, and H. Yan. *Nano Lett.*, 6(2):248–251, 2006.

[10] J. D. Le, Y. Pinto, N. C. Seeman, K. Musier-Forsyth, T. A. Taton, and R. A. Kiehl. *Nano Letters*, 4(12):2343–2347, 2004.

[11] G. Schmid and A. Lehnert. *Angew. Chemie - I. E.*, 28(6):780–81, 1989.

[12] D. Zanchet, C. M. Micheel, W. J. Parak, D. Gerion, S. C. Williams, and A. P. Alivisatos. *J. Phys. Chem. B*, 106(45):11758–11763, 2002.

[13] J. Sharma, R. Chhabra, Y. Liu, Y. G. Ke, and H. Yan. *Angew. Chem. Int. Ed.*, 45(5):730–735, 2006.