Self-Assembly of Complex DNA Nanostructures and Reconfigurable DNA Devices

by

Fei Zhang

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved April 2015 by the
Graduate Supervisory Committee:

Hao Yan, Co-Chair
Yan Liu, Co-Chair
Ian Gould
Peiming Zhang

ARIZONA STATE UNIVERSITY

May 2015

ABSTRACT

Deoxyribonucleic acid (DNA) has emerged as an excellent molecular building block for nanoconstruction in addition to its biological role of preserving genetic information. Its unique features such as predictable conformation and programmable intra- and inter-molecular Watson-Crick base pairing interactions make it a remarkable engineering material. A variety of convenient design rules and reliable assembly methods have been developed to engineer DNA nanostructures. The ability to create designer DNA architectures with accurate spatial control has allowed researchers to explore novel applications in directed material assembly, structural biology, biocatalysis, DNA computing, nano-robotics, disease diagnosis, and drug delivery.

This dissertation focuses on developing the structural design rules for "static" DNA nano-architectures with increasing complexity. By using a modular self-assembly method, Archimedean tilings were achieved by association of different DNA motifs with designed arm lengths and inter-tile sticky end interactions. By employing DNA origami method, a new set of design rules was created to allow the scaffolds to travel in arbitrary directions in a designed geometry without local symmetry restrictions. Sophisticated wireframe structures of higher-order complexity were designed and constructed successfully. This dissertation also presents the use of "dynamic" DNA nanotechnology to construct DNA origami nanostructures with programmed reconfigurations.

ACKNOWLEDGMENTS

I wish to thank, first and foremost, my research advisors Dr. Hao Yan and Dr. Yan Liu for being such extraordinary mentors. They showed me the path to this wonderful DNA nanotechnology world. Their knowledge, attitude and enthusiasm always inspire me throughout my graduate study. They also provided me numerous professional guidance and personal support. I also wish to thank Dr. Ian Gould and Dr. Peiming Zhang, my supervisory committee members, who offered me invaluable advices.

I am deeply grateful to all the past and present Yan lab members, who have created a supportive and dynamic academic environment that made my study at ASU a very pleasant experience. I would like to thank Dr. Xiaowei Liu who taught me all the basic lab techniques. Dr. Jeanette Nangreave helped my spoken English and proofread my manuscripts. I discussed many ideas and experimental details with Dr. Yang Yang, Dr. Wei Li and Dr. Xiaodong Qi. I also thank Shuoxing Jiang, Chad Simmons, and Guoliang Ke for being great collaborators and team players.

This thesis would not have been possible without the support of my family. My husband Dian Xu always encourages me and my parents have been very supportive since I wanted to pursue academic career. My son Weiteng Xu brings me joy and happiness to my life and makes me strong.

TABLE OF CONTENTS

CHAPTER Page

LIST OF FIGURES

Figure Page

CHAPTER 1

INTRODUCTION

Adapted with permission from Zhang, F.; Nangreave, J.; Liu, Y.; Yan, H. Structural DNA Nanotechnology: State of the Art and Future Perspective, *J. Am. Chem. Soc.*, **2014**, 136, 11198–11211. Copyright 2014 American Chemical Society.

1.1 Introduction of DNA nanotechnology

Self-assembly is a remarkable process by which chemical systems composed of non-living components are organized into living, biological systems. Nature accomplishes this incredible feat by adding information to matter and by guiding the self-assembly process to create functional structures. As the basic units of life, living cells are the epitome of molecular sophistication, with a multitude of components and systems that interact to perform thousands of routine functions like creating and using energy, synthesizing nucleic acids and proteins, and responding to environmental cues. The components of a cell range in size from individual atoms and molecules, to globular nanoscale objects, to even larger highly ordered structures such as microtubules. Nanotechnology is a field in which researchers manipulate matter on the same scale, operating in the window between molecular and macroscopic worlds, a window that is ideally suited for information storage, processing, and transmission.

Humanity has a long history of looking to nature for inspiration, from prehistoric times when early humans mimicked the hunting techniques of other animals, to Leonardo daVinci's 15th century sketches of flying machines based on bird flight, to today's scientists and engineers who are attempting to engineer cell like structures and machines.

With the exponential advances in science and technology that have occurred over the past few decades, scientists are getting closer to engineering bio-inspired components that can communicate, regulate and actuate in artificial molecular networks. Toward this goal, information coding polymers such as DNA, RNA and proteins are ideal building blocks in the assembly of designer nano-architectures. This introduction will concentrate on the most recent and inspiring advances in structural DNA nanotechnology. An outlook of the future of this rapidly expanding field is presented in Chapter 6. More comprehensive reviews that provide very detailed descriptions of the state of the art in this field can be found elsewhere in the literature[1-5].

1.2 Structural DNA construction

DNA, nature's molecule of choice for storing and transmitting genetic information, is an excellent nanoscale building block because of its specific three-dimensional (3D) conformation, chemical addressability, and predictable Watson-Crick base pairing. Structural DNA nanotechnology, derived from Seeman's innovative proposal that DNA could be used as a physical material for the self-assembly of nanoscale structures[6] (Figure 1.1), has developed with astounding speed over the past 30 years. The most significant underlying concept is the application of immobile, branched DNA junctions, together with sequence specific sticky end associations, to create self-assembling arrays, objects and devices (Figure 1.1A).

Over the past several decades, researchers have established a collection of convenient methods to construct DNA nanostructures that exhibit significant geometric and topological complexity. Designing and predicting the three dimensional (3D)

conformation of these nanostructures is now routine thanks to several user-friendly software interfaces that have been developed[7-12]. A number of two-dimensional and three-dimensional lattices assembled from small, repeating DNA nanostructure motifs were produced[13-20] (Figure 1.1C), and several discrete polyhedral objects were constructed from fixed numbers of DNA junction motifs[21-27] (Figure 1.1D). In 2009, Seeman's group was the first to assemble 3D DNA crystals from deliberately designed sticky-end connections (Figure 1.1C, right), rather than through simple, non-specific base stacking[16]. They used self-assembling tensegrity triangle motifs to create 3D crystals with various unit dimensions. This work represents a milestone in fulfilling Seeman's initial vision of using 3D DNA lattices as hosts to organize guest protein molecules and facilitate protein crystallography[6] (Figure 1.1B). Several researchers encoded algorithms into DNA nanostructure components to direct the assembly of particular 2D lattice arrays and had some initial success[28-29] (Figure 1.1E); however, scaling up algorithmic assembly to realize more complex patterns remains a challenge, mainly because of the errors that accumulate during assembly. If error correction mechanisms[30-31] could be implemented, it would represent a ground-breaking advance in this field.

In 2006, the emergence of DNA-origami[32] transformed the landscape of DNA nanotechnology. The DNA origami method uses a number of short single stranded DNA (ssDNA) oligonucleotides to direct the folding path of a long ssDNA 'scaffold' strand. Rothemund used genomic ssDNA from the M13 phage as the scaffold strand (7249 nucleotides), and designed a set of short "staple" strands to selectively bind to distant regions of the scaffold and fold it into a pre-designed shape. This assembly method results in near-quantitative yield for most 2D designs (Figure 1.1F), even with unpurified

staples. Several groups successfully extended DNA origami fabrication to 3D[33-36], and to the assembly of twisted[37] and curved[38] 3D objects (Figure 1.1F). Other research groups have focused their attention on scaling up DNA origami using the following methods: edge-to-edge base stacking interactions between individual origami units[39], sequence specific sticky end cohesion between individual units[40], "super origami"[41] and use of longer scaffold for origami construction[42-43].

More recently, Yin and coworkers synthesized a variety of 1D[51], 2D[45] and 3D[46] DNA nanostructures from single stranded DNA tiles (SSTs). The platform that they developed is based on a series of interlocking local connections between SSTs. Collections of SSTs form 2D sheet or 3D block canvases that can be selectively engraved to create different shapes and patterns, by simply including or omitting specific SSTs (Figure 1.1G).

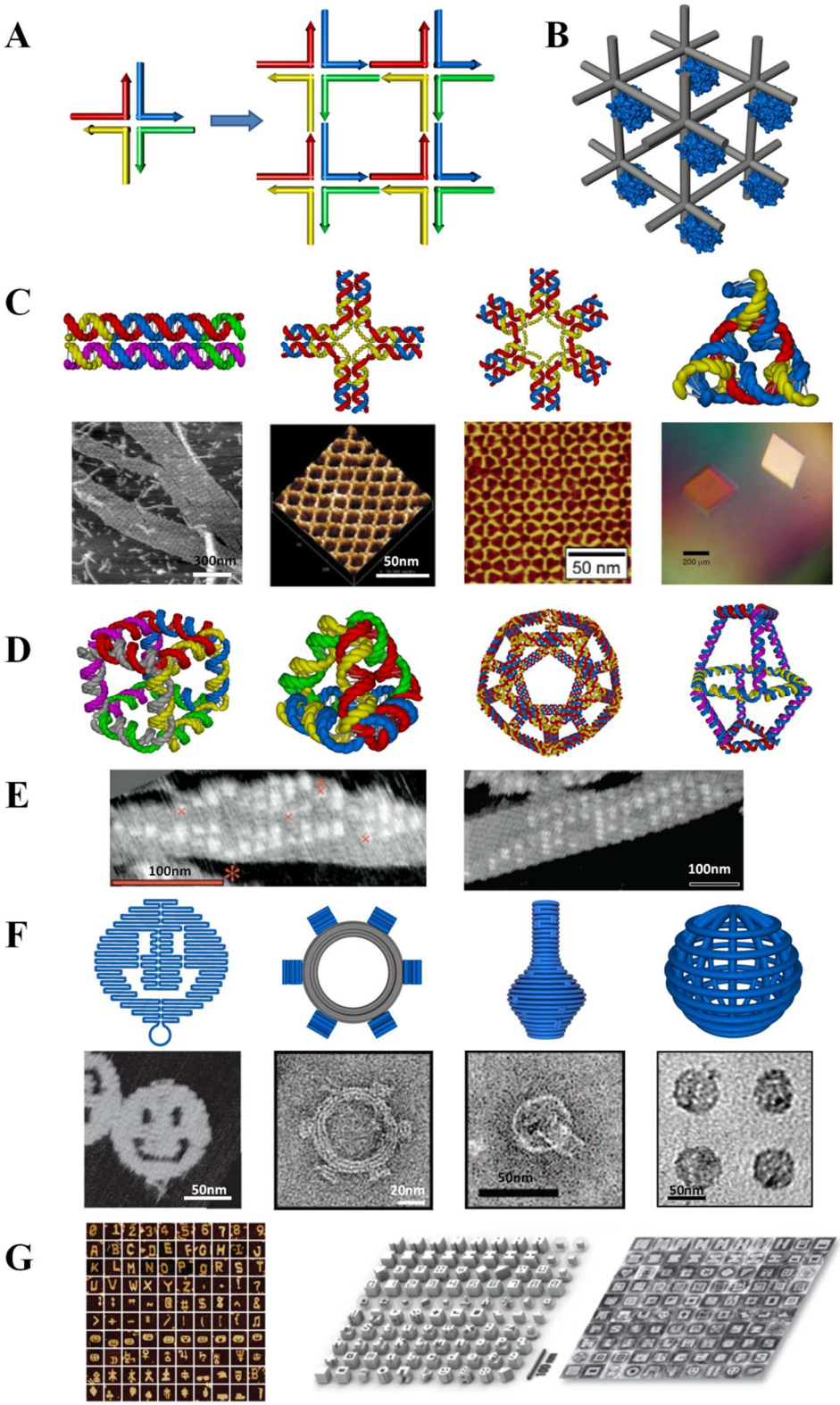

A
B
C
300nm
50nm
50 nm
200 µm
D
E
100nm
100nm
F
50nm
20nm
50nm
50nm
G

Figure 1.1 Structural foundations of structural DNA nanotechnology. (A & B) Seeman's original proposals to use immobile DNA junctions to create self-assembling arrays[6], and self-assembled 3D DNA lattices as scaffolds to organize macromolecules into crystalline lattices[6]. (C) Representative examples of DNA nanostructure motifs used to create periodic 2D arrays and 3D crystal (top: helical structure of the motif; bottom: AFM images of the assembled 2D arrays and optical image of the 3D crystal). From left to right: double-crossover DNA tile[13], 4 by 4 DNA tile[14] and 6 by 4 DNA tile[15]. And 3D crystal[16]. (D) Representative examples of polyhedral DNA nanostructures. From left to right: molecular models of a DNA cube[21], DNA tetrahedron[22], DNA dodecahedron[23] and DNA biprism[24]. (E) Representative examples of algorithmic self-assembly based on double-crossover tiles: Sierpinski triangles[28] (left) and binary counter[29] (right). (F) Representative examples of DNA origami nanostructures (top: schematic drawings of the structures; bottom: corresponding AFM or TEM images). From left to right: a 2D DNA origami smiley face[32], a 3D DNA origami in the shape of a gear[37], a curved single layer 3D origami in the shape of a vase[38] and a DNA origami gridiron[44]. (G) Complex nanostructures produced using the single stranded DNA tile strategy[45,46]. Figures reproduced with permission from: (C) ref. 13, 1998 NPG, ref. 14, 2003 AAAS, ref. 15, 2006 ACS, ref. 16, 2009 NPG; (E) ref. 28, courtesy of P. Rothemund, ref. 29, 2005 ACS; (F) ref. 32, 2006 NPG, ref. 37, 2009 AAAS, ref. 38, 2011 AAAS, ref. 113, 2013 AAAS; (G) ref. 45, 2012 NPG, ref. 46, 2012 AAAS.

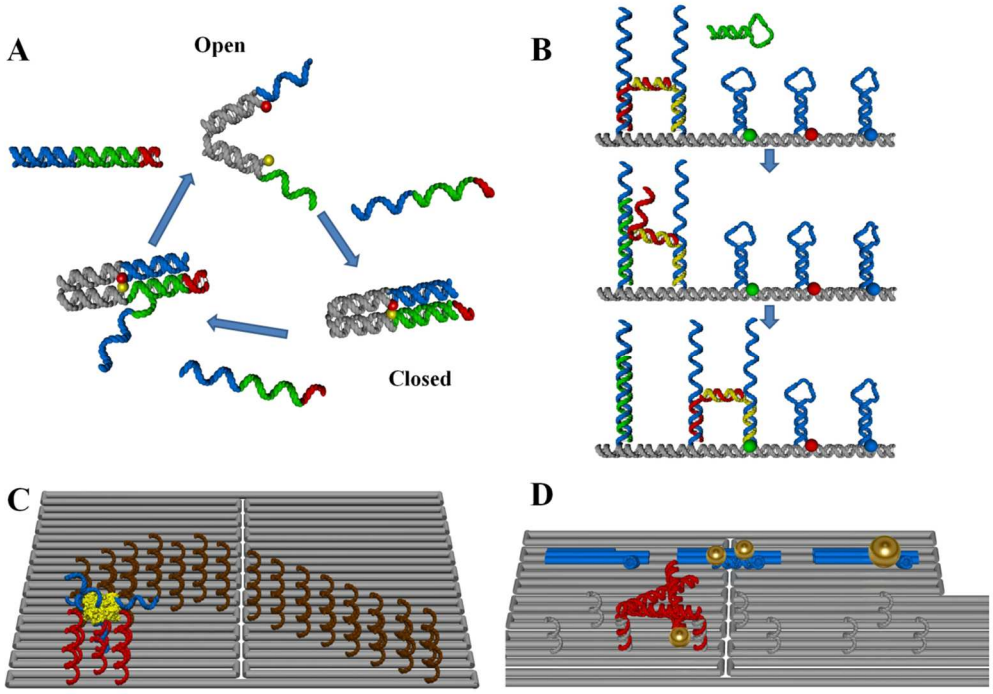


Figure 1.2 Representative dynamic DNA nanostructures. (A) A DNA tweezer based on DNA strand displacement technique[47]. (B) A autonomous DNA walker catalyzed by meta-stable DNA hairpin fuel[48]. (C) Movement of DNA spider on a prescribed

landscape[49]. (D) A DNA assembly line: DNA walker will transport gold nanoparticles to different product formation station with instructions from DNA strand displacement[50].

1.3 Dynamic DNA nanodevices

Natural biological devices are designed to operate in dynamic conditions, responding to subtle biological cues to realize their functions. The structural properties of DNA that allow it to serve as a versatile construction material have been exploited to create dynamic nanodevices (Figure 1.2A) ranging from small switchable structures[47, 52-56] and reconfigurable systems[57-62], to structures that display complex movements such as rolling[63], rotating[64] and walking[48-49, 65-66].

Protein molecular motors transform chemical energy into mechanical energy to facilitate a variety of biological functions from cell division, transport, and motility, to enzymatic activity. DNA nanotechnologists have long envisioned programming DNA walker molecules to mimic the ability of natural motor proteins to walk along intracellular tracks and achieve controlled motion. Imparting directionality to DNA walkers could be realized by means of successive addition of DNA fuels, by coordinating conformational changes between different components of the walker, by leading the walker through selective track modifications, or by pairing their motion to unidirectional reaction cycles. Researchers have already demonstrated unidirectional motion by DNA walkers through prescribed tracks[48, 67] and landscapes[49] (Figure 1.2B&C). Based on this technology, it is possible to develop walkers that are programmed to travel a certain path by encoding the directions into the nucleotide sequences of the walker itself, and into the corresponding landscape. For example, Seeman's group reported a DNA-based robot that

manufactured structures on a nanoscale assembly line[50] (Figure 1.2D). Their DNA walker traveled through three fixed modules that were individually programmed to selectively incorporate a gold nanoparticle into the final product, resulting in eight possible outcomes. Recently, researchers reported those DNA walkers were also used to mediate multistep organic synthesis[68].

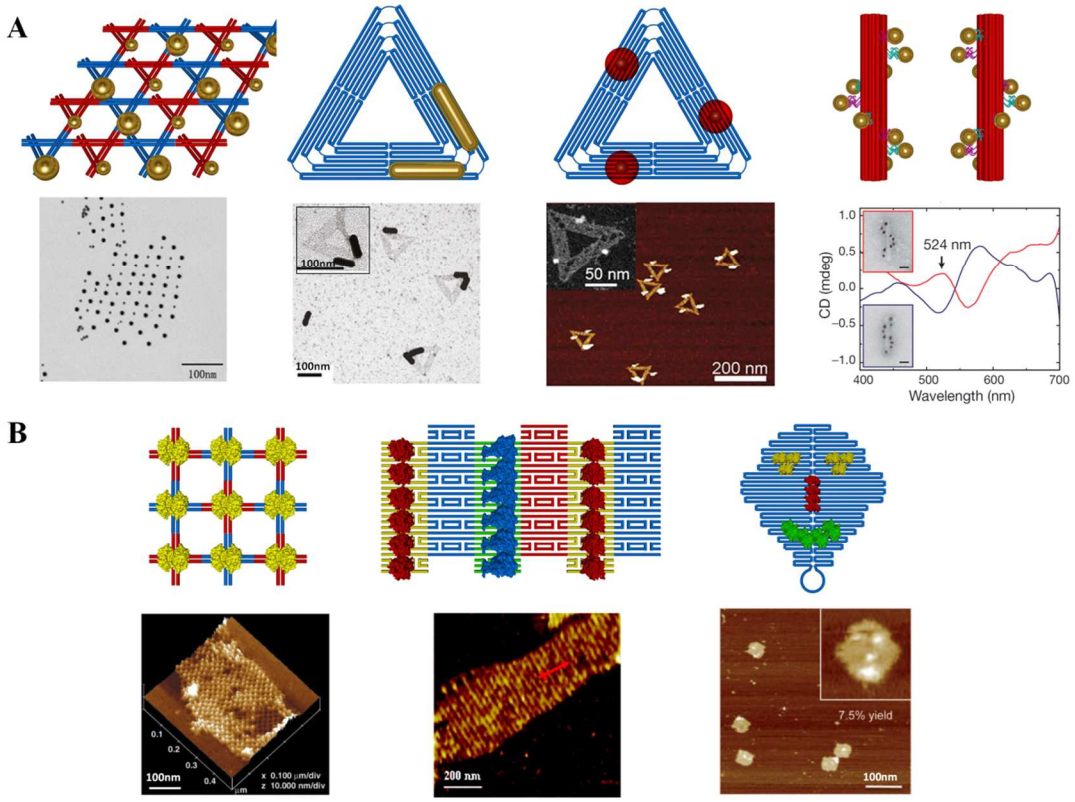


Figure 1.3 Representative examples of DNA nanostructure directed assembly of inorganic and protein molecules (top: schematics; bottom: corresponding TEM or AFM images). (A) From left to right: gold nanoparticles organized by a 2D DNA tile array[69], gold nanorod dimers with controlled angles between the nanorods organized by DNA origami,[70] DNA origami directed quantum dot architectures[71], and DNA origami directed gold nanoparticles into chiral arrangement and the induced circular dichromic effect[72]. (B) Organization of streptavidin proteins by a 2D DNA nanoarray[14], protein arrays templated by a 2D DNA nanostructure through aptamer-protein interactions[73], orthogonal Snap-tag and His-tag mediated decoration of DNA origami[74] . Figures reproduced with permission from: (A) ref. 69, 2006 ACS, ref. 70, 2011 ACS, ref. 71, 2012 ACS, ref. 72, 2012 NPG; (B) ref. 14, 2003 AAAS, ref. 161, 2007 ACS, ref. 162, 2010 Wiley.

1.4 Applications of DNA nanotechnology

As structural DNA nanotechnology transitions from adolescence into adulthood, the need to demonstrate potential applications is of the utmost importance. We must improve our ability to engineer and program complex molecular systems and prove that designer DNA nanostructures can be employed to real world applications. If we continue to exploit the programmability of DNA nanostructures to accurately template functional molecules, materials and probes, we will be able to organize these external elements into practical devices and engineer molecular sensors, circuits, and actuators.

Inorganic nanomaterials such as quantum dots, nanowires and rods, and metal nanoparticles have attracted attention because of their unique optical and electronic properties that can be used in solar cells, phototransistors, laser diodes, light-emitting diodes, and other optoelectronic devices[75]. However, a better understanding of the photo-physical behavior of these materials is necessary to use them in such devices. Researchers have successfully used DNA nanoscaffolds to organize metallic nanoparticles, semiconductor nanocrystals[69-72, 76-77] (Figure 1.3A), organic chromophores[78], and cholesterol moieties[79] into well-defined architectures. These inorganic particle-DNA nanostructure complexes have enabled systematic investigation of distance dependent interactions between photonic elements[80-82]. In one example, Liedl and co-workers constructed a spiral, nanoscale staircase on which gold nanoparticles were arranged at regular intervals and with chiral geometries[72] (Figure 1.3A). This work demonstrates how DNA scaffolding can be used to control the precise structural arrangement of metal nanoparticles, enabling researchers to tailor surface plasmon resonance and the

interaction with visible light. In another example, DNA nanostructures were used to organize various organic chromophores into artificial light harvesting complexes with control over cascading, unidirectional energy transfer[78].

As we previously mentioned, one of the initial goals of structural DNA nanotechnology was to use 3D DNA lattices as hosts to organize guest protein molecules and facilitate protein crystallography. Although this vision has yet to be realized, people have already begun to use DNA nanostructures as chaperones to align and organize protein molecules using different strategies, including ligand-protein (such as Biotin-streptavidin) interactions, aptamer-target interactions, and ligand-engineered (tagged) protein interactions (Figure 1.3B). Shih and co-workers recently designed DNA origami nanotube liquid crystals to provide the appropriate "alignment environment" for determining the previously unknown structure of a membrane protein by nuclear magnetic resonance (NMR)[83]. Turberfield and co-workers used periodic 2D DNA tile arrays as templates to arrange proteins and subsequently used cryo-EM to solve their structures[84].

Some of nature's most powerful agents, proteins, are large macromolecules that perform a wide assortment of functions required to sustain life, including metabolic catalysis, DNA replication, and molecular transport. In order to better understand the governing dynamics in complex protein systems we need control over the number, orientation, and arrangement of the constituents. Nucleic acid scaffolds afford this level of control and researchers have already used RNA and DNA platforms to engineer a number of enzyme cascades[85-88] (Figure 1.4A). For example, Silver and co-workers used a bacterial host to transcribe RNA and assemble intracellular RNA nanoscaffolds for

spatial organization of metabolic elements for hydrogen production[85]. Willner and co-workers organized glucose oxidase and horseradish peroxidase enzyme cascade by 2D DNA lattices[87]. More recently, Fu conducted substrate channeling in a multi-enzyme cascade by an artificial DNA swinging arm[89].

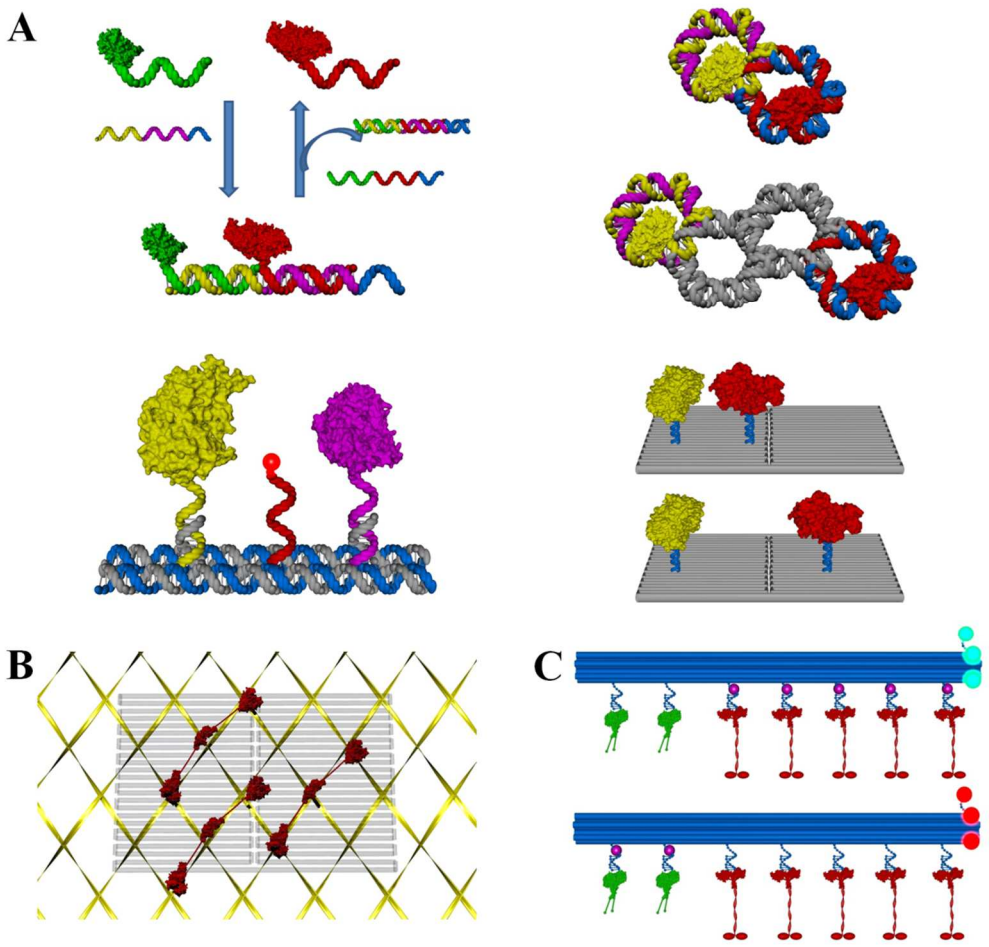


Figure 1.4 Representative examples of DNA nanostructure directed assembly protein molecules for functional structures. (A) From left to right: (upper panel) assembly and disassembly of holoenzymes mediated by DNA strand displacement[86], glucose oxidase (yellow) and horseradish peroxidase (red) enzyme cascade organized by 2D DNA lattices[87], (lower panel) substrate channeling in a multi-enzyme cascade by an artificial DNA swinging arm[89], glucose oxidase (yellow) and horseradish peroxidase (red) enzyme cascade organized on DNA origami with distance control[88]. (B) Rectangular DNA origami travels on a cellular actin network through the binding and action of myosin lever

arms[90]. (C) Molecular tug of war between two motor proteins displayed from a 12-helix DNA bundle[91].

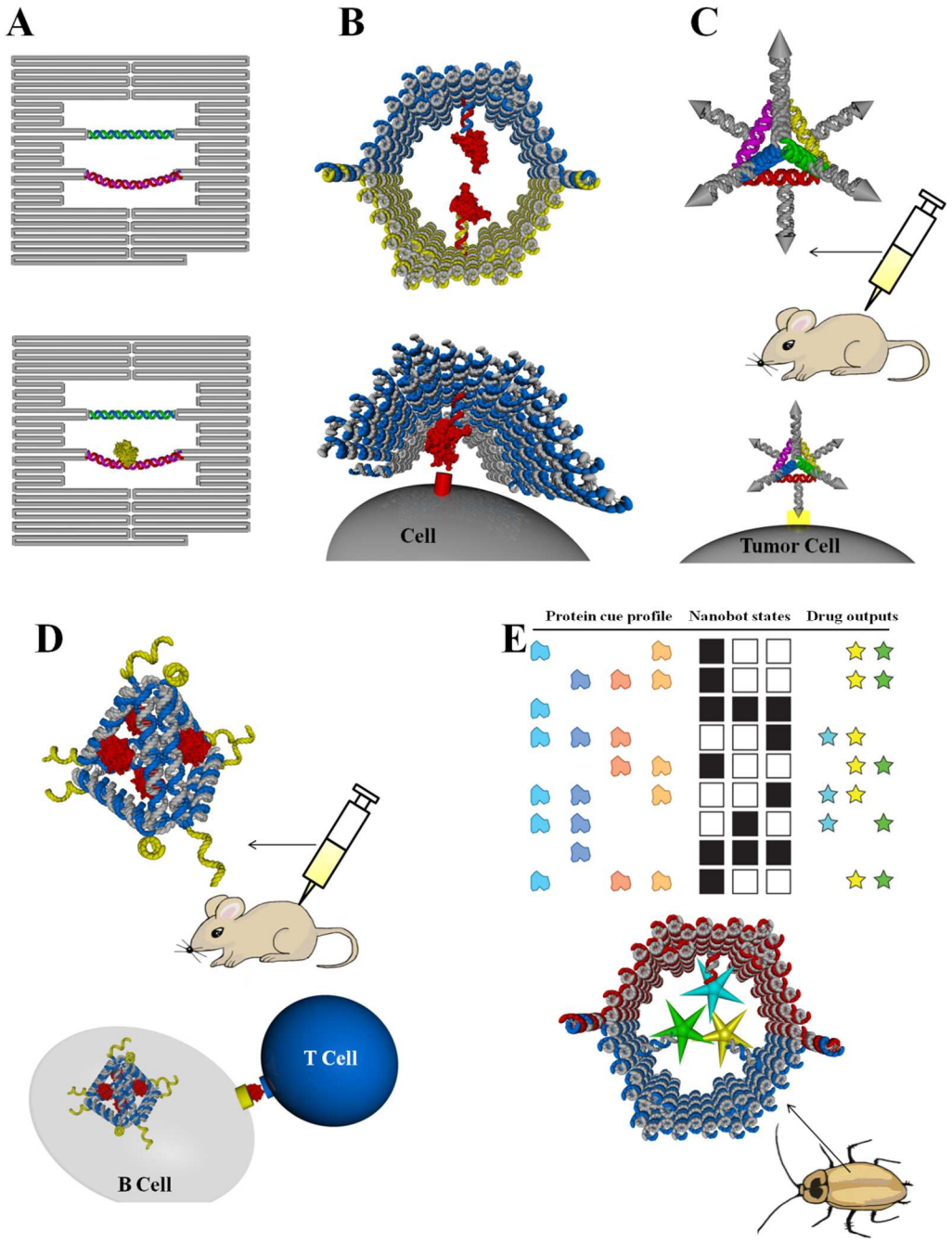


Figure 1.5 Biological applications of DNA directed assembly. (A) A DNA origami frames to investigate protein-DNA binding events in real time at the single molecule level[92]. (B) A barrel-like DNA nanorobot programmed to be open in the presence of target cells and expose Fab antibody fragment cargo[93]. (C) Six siRNA duplexes and folic acid tags (grey) chaperoned by a DNA tetrahedron are injected into mice; the tetrahedra

bind to tumor cells by targeting folate receptors expressed on the tumor cell surface[94]. (D) A DNA tetrahedron adjuvant-antigen vaccine complex; CpG ODN adjuvant molecules (curved yellow ribbons) and the model streptavidin antigen (red) bind specifically to B cells and are subsequently presented to T cells to activate B cell response and antibody production[95]. (E) Three different drugs carried by a DNA nanorobot can be released in a programmed fashion by undergoing complex DNA computation in a living cockroach[96]. Figures reproduced with permission from: (E) ref. 98, 2014 NPG.

DNA origami scaffolds have also been used to organize motor proteins and study their spatially dependent motility[90] (Figure 1.4B). Understanding how motors cooperate productively, and compete antagonistically, is important for understanding how intracellular transport is regulated. Researchers recently demonstrated this "molecular tug-of-war" by displaying different numbers of dynein and kinesin motor proteins from a DNA-origami structure[91]. By controlling the number, distance and orientations of the two types of biological motors they were able to systematically study coordinated motor behavior (Figure 1.4C).

Structural DNA nanotechnology has also emerged as a useful tool for biological and medicinal applications (Figure 1.5). The intrinsic biocompatibility, nanoscale dimensions, programmability, and ability for functionalization of DNA nanostructures are virtually unrivaled by existing techniques. In particular, the addressable configuration of DNA origami lends itself to detection of gene expression[97] and single nucleotide polymorphism[98]. The Sugiyama group developed DNA origami frames and rulers to investigate biomolecular interactions such as protein-DNA binding events and homologous recombination processes in real time at the single molecule level[92, 99-100] (Figure 1.5A). Further, the spatial addressability and multivalent properties of DNA nanostructures make them promising vehicles for targeted drug delivery. For example,

Douglas and co-workers demonstrated a barrel-shaped nanorobot that releases Fab antibody fragments in the presence of target cells[93]. In their system two single stranded DNA aptamer locks are opened by specific markers present on the surface of cells (Figure 1.5B). After opening, the payload molecules inside the barrel are exposed, inducing a particular cellular signaling pathway. Anderson and co-workers used a DNA tetrahedron to deliver small interfering RNA *in vivo* to target and suppress gene expression in a mouse model[94] (Figure 1.5C). Programmable DNA nanostructures have also been used as synthetic vaccine platforms[95, 101]. The Yan group used a DNA tetrahedron to co-assemble model antigens and CpG adjuvants into nanoscale complexes with precise control of the valency and spatial arrangement of each component[95] (Figure 1.5D). Tests on immunized mice demonstrated that antigen-adjuvant-DNA complexes induced stronger and longer-lasting antibody responses against the antigen, without stimulating a reaction to the DNA nanostructure itself, as compared to an unstructured mixture of antigen and CpG molecules. More recently, Amir *et al.* showed that DNA origami robots can dynamically interact with each other and perform logic computations in a living animal, opening up opportunities to develop smart theranostic nanodevices[96] (Figure 1.5E).

1.5 References

(1) Aldaye, F. A.; Palmer, A. L.; Sleiman, H. F., *Science* **2008,** 321, 1795-1799.

(2) Li, H. Y.; Carter, J. D.; LaBean, T. H., *Mater Today* **2009,** 12, 24-32.

(3) Seeman, N. C., *Annu Rev Biochem* **2010,** 79, 65-87.

(4) Service, R. F., *Science* **2011,** 332, 1140-1142.

(5) Zadegan, R. M.; Norton, M. L., *Int J Mol Sci* **2012,** 13, 7149-7162.

(6) Seeman, N. C., *J Theor Biol* **1982,** 99, 237-247.

(7) Seeman, N. C., *Journal of Biomolecular Structure and Dynamics* **1990,** 8, 573-581.

(8) Andersen, E. S.; Dong, M. D.; Nielsen, M. M.; Jahn, K.; Lind-Thomsen, A.; Mamdouh, W.; Gothelf, K. V.; Besenbacher, F.; Kjems, J., *Acs Nano* **2008,** 2, 1213-1218.

(9) Douglas, S. M.; Marblestone, A. H.; Teerapittayanon, S.; Vazquez, A.; Church, G. M.; Shih, W. M., *Nucleic Acids Res* **2009,** 37, 5001-5006.

(10) Williams, S.; Lund, K.; Lin, C.; Wonka, P.; Lindsay, S.; Yan, H., **2009,** 5347, 90-101.

(11) Zadeh, J. N.; Steenberg, C. D.; Bois, J. S.; Wolfe, B. R.; Pierce, M. B.; Khan, A. R.; Dirks, R. M.; Pierce, N. A., *J Comput Chem* **2011,** 32, 170-173.

(12) Kim, D. N.; Kilchherr, F.; Dietz, H.; Bathe, M., *Nucleic Acids Res* **2012,** 40, 2862-2868.

(13) Winfree, E.; Liu, F. R.; Wenzler, L. A.; Seeman, N. C., *Nature* **1998,** 394, 539-544.

(14) Yan, H.; Park, S. H.; Finkelstein, G.; Reif, J. H.; LaBean, T. H., *Science* **2003,** 301, 1882-1884.

(15) He, Y.; Tian, Y.; Ribbe, A. E.; Mao, C. D., *J Am Chem Soc* **2006,** 128, 15978-15979.

(16) Zheng, J. P.; Birktoft, J. J.; Chen, Y.; Wang, T.; Sha, R. J.; Constantinou, P. E.; Ginell, S. L.; Mao, C. D.; Seeman, N. C., *Nature* **2009,** 461, 74-77.

(17) LaBean, T. H.; Yan, H.; Kopatsch, J.; Liu, F.; Winfree, E.; Reif, J. H.; Seeman, N. C., *J Am Chem Soc* **2000,** 122, 1848-1860.

(18) He, Y.; Chen, Y.; Liu, H. P.; Ribbe, A. E.; Mao, C. D., *J Am Chem Soc* **2005,** 127, 12202-12203.

(19) Malo, J.; Mitchell, J. C.; Venien-Bryan, C.; Harris, J. R.; Wille, H.; Sherratt, D. J.; Turberfield, A. J., *Angew Chem Int Edit* **2005,** 44, 3057-3061.

(20) Majumder, U.; Rangnekar, A.; Gothelf, K. V.; Reif, J. H.; LaBean, T. H., *J Am Chem Soc* **2011,** 133, 3843-3845.

(21) Chen, J. H.; Seeman, N. C., *Nature* **1991,** 350, 631-633.

(22) Goodman, R. P.; Schaap, I. A. T.; Tardin, C. F.; Erben, C. M.; Berry, R. M.; Schmidt, C. F.; Turberfield, A. J., *Science* **2005,** 310, 1661-1665.

(23) He, Y.; Ye, T.; Su, M.; Zhang, C.; Ribbe, A. E.; Jiang, W.; Mao, C. D., *Nature* **2008,** 452, 198-201.

(24) Aldaye, F. A.; Sleiman, H. F., *J Am Chem Soc* **2007,** 129, 13376-13377.

(25) Zhang, Y.; Seeman, N. C., *J Am Chem Soc* **1994,** 116, 1661-1669.

(26) Shih, W. M.; Quispe, J. D.; Joyce, G. F., *Nature* **2004,** 427, 618-621.

(27) Erben, C. M.; Goodman, R. P.; Turberfield, A. J., *J Am Chem Soc* **2007,** 129, 6992-6993.

(28) Rothemund, P. W. K.; Papadakis, N.; Winfree, E., *Plos Biol* **2004,** 2, 2041-2053.

(29) Barish, R. D.; Rothemund, P. W. K.; Winfree, E., *Nano Lett* **2005,** 5, 2586-2592.

(30) Winfree, E.; Bekbolatov, R., *Lect Notes Comput Sc* **2004,** 2943, 126-144.

(31) Sahu, S.; Reif, J. H., *Algorithmica* **2010,** 56, 480-504.

(32) Rothemund, P. W. K., *Nature* **2006,** 440, 297-302.

(33) Ke, Y. G.; Sharma, J.; Liu, M. H.; Jahn, K.; Liu, Y.; Yan, H., *Nano Lett* **2009,** 9, 2445-2447.

(34) Andersen, E. S.; Dong, M.; Nielsen, M. M.; Jahn, K.; Subramani, R.; Mamdouh, W.; Golas, M. M.; Sander, B.; Stark, H.; Oliveira, C. L. P.; Pedersen, J. S.; Birkedal, V.; Besenbacher, F.; Gothelf, K. V.; Kjems, J., *Nature* **2009,** 459, 73-75.

(35) Douglas, S. M.; Dietz, H.; Liedl, T.; Hogberg, B.; Graf, F.; Shih, W. M., *Nature* **2009,** 459, 414-418.

(36) Endo, M.; Hidaka, K.; Kato, T.; Namba, K.; Sugiyama, H., *J Am Chem Soc* **2009,** 131, 15570-15571.

(37) Dietz, H.; Douglas, S. M.; Shih, W. M., *Science* **2009,** 325, 725-730.

(38) Han, D. R.; Pal, S.; Nangreave, J.; Deng, Z. T.; Liu, Y.; Yan, H., *Science* **2011,** 332, 342-346.

(39) Woo, S.; Rothemund, P. W. K., *Nat Chem* **2011,** 3, 620-627.

(40) Liu, W. Y.; Zhong, H.; Wang, R. S.; Seeman, N. C., *Angew Chem Int Edit* **2011,** 50, 264-267.

(41) Zhao, Z.; Liu, Y.; Yan, H., *Nano Lett* **2011,** 11, 2997-3002.

(42) Zhang, H. L.; Chao, J.; Pan, D.; Liu, H. J.; Huang, Q.; Fan, C. H., *Chem Commun* **2012,** 48, 6405-6407.

(43) Pound, E.; Ashton, J. R.; Becerril, H. c. A.; Woolley, A. T., *Nano Lett* **2009,** 9, 4302-4305.

(44) Han, D. R.; Pal, S.; Yang, Y.; Jiang, S. X.; Nangreave, J.; Liu, Y.; Yan, H., *Science* **2013,** 339, 1412-1415.

(45) Wei, B.; Dai, M. J.; Yin, P., *Nature* **2012,** 485, 623-626.

(46) Ke, Y. G.; Ong, L. L.; Shih, W. M.; Yin, P., *Science* **2012,** 338, 1177-1183.

(47) Yurke, B.; Turberfield, A. J.; Mills, A. P.; Simmel, F. C.; Neumann, J. L., *Nature* **2000,** 406, 605-608.

(48) Yin, P.; Choi, H. M. T.; Calvert, C. R.; Pierce, N. A., *Nature* **2008,** 451, 318-322.

(49) Lund, K.; Manzo, A. J.; Dabby, N.; Michelotti, N.; Johnson-Buck, A.; Nangreave, J.; Taylor, S.; Pei, R. J.; Stojanovic, M. N.; Walter, N. G.; Winfree, E.; Yan, H., *Nature* **2010,** 465, 206-210.

(50) Gu, H. Z.; Chao, J.; Xiao, S. J.; Seeman, N. C., *Nature* **2010,** 465, 202-205.

(51) Yin, P.; Hariadi, R. F.; Sahu, S.; Choi, H. M. T.; Park, S. H.; LaBean, T. H.; Reif, J. H., *Science* **2008,** 321, 824-826.

(52) Mao, C. D.; Sun, W. Q.; Shen, Z. Y.; Seeman, N. C., *Nature* **1999,** 397, 144-146.

(53) Yan, H.; Zhang, X. P.; Shen, Z. Y.; Seeman, N. C., *Nature* **2002,** 415, 62-65.

(54) Li, J. W. J.; Tan, W. H., *Nano Lett* **2002,** 2, 315-318.

(55) Liu, D. S.; Balasubramanian, S., *Angew Chem Int Edit* **2003,** 42, 5734-5736.

(56) Li, Y. M.; Zhang, C.; Tian, C.; Mao, C. D., *Org Biomol Chem* **2014,** 12, 2543-2546.

(57) Feng, L. P.; Park, S. H.; Reif, J. H.; Yan, H., *Angew Chem Int Edit* **2003,** 42, 4342-4346.

(58) Goodman, R. P.; Heilemann, M.; Doose, S.; Erben, C. M.; Kapanidis, A. N.; Turberfield, A. J., *Nat Nanotechnol* **2008,** 3, 93-96.

(59) Maye, M. M.; Kumara, M. T.; Nykypanchuk, D.; Sherman, W. B.; Gang, O., *Nat Nanotechnol* **2010,** 5, 116-120.

(60) Han, D. R.; Pal, S.; Liu, Y.; Yan, H., *Nat Nanotechnol* **2010,** 5, 712-717.

(61) Zhang, F.; Nangreave, J.; Liu, Y.; Yan, H., *Nano Lett* **2012,** 12, 3290-3295.

(62) Yang, Y. Y.; Endo, M.; Hidaka, K.; Sugiyama, H., *J Am Chem Soc* **2012,** 134, 20645-20653.

(63) Tian, Y.; Mao, C. D., *J Am Chem Soc* **2004,** 126, 11410-11411.

(64) Rajendran, A.; Endo, M.; Hidaka, K.; Sugiyama, H., *J Am Chem Soc* **2013,** 135, 1117-1123.

(65) Omabegho, T.; Sha, R.; Seeman, N. C., *Science* **2009,** 324, 67-71.

(66) Wickham, S. F. J.; Endo, M.; Katsuda, Y.; Hidaka, K.; Bath, J.; Sugiyama, H.; Turberfield, A. J., *Nat Nanotechnol* **2011,** 6, 166-169.

(67) Wickham, S. F. J.; Bath, J.; Katsuda, Y.; Endo, M.; Hidaka, K.; Sugiyama, H.; Turberfield, A. J., *Nat Nanotechnol* **2012,** 7, 169-173.

(68) He, Y.; Liu, D. R., *Nat Nanotechnol* **2010,** 5, 778-782.

(69) Zheng, J. W.; Constantinou, P. E.; Micheel, C.; Alivisatos, A. P.; Kiehl, R. A.; Seeman, N. C., *Nano Lett* **2006,** 6, 1502-1504.

(70) Pal, S.; Deng, Z. T.; Wang, H. N.; Zou, S. L.; Liu, Y.; Yan, H., *J Am Chem Soc* **2011,** 133, 17606-17609.

(71) Deng, Z. T.; Samanta, A.; Nangreave, J.; Yan, H.; Liu, Y., *J Am Chem Soc* **2012,** 134, 17424-17427.

(72) Kuzyk, A.; Schreiber, R.; Fan, Z. Y.; Pardatscher, G.; Roller, E. M.; Hogele, A.; Simmel, F. C.; Govorov, A. O.; Liedl, T., *Nature* **2012,** 483, 311-314.

(73) Chhabra, R.; Sharma, J.; Ke, Y. G.; Liu, Y.; Rinker, S.; Lindsay, S.; Yan, H., *J Am Chem Soc* **2007,** 129, 10304-10305.

(74) Sacca, B.; Meyer, R.; Erkelenz, M.; Kiko, K.; Arndt, A.; Schroeder, H.; Rabe, K. S.; Niemeyer, C. M., *Angew Chem Int Edit* **2010,** 49, 9378-9383.

(75) Tan, S. J.; Campolongo, M. J.; Luo, D.; Cheng, W. L., *Nat Nanotechnol* **2011,** 6, 268-276.

(76) Ding, B.; Deng, Z.; Yan, H.; Cabrini, S.; Zuckermann, R. N.; Bokor, J., *J Am Chem Soc* **2010,** 132, 3248-3249.

(77) Klein, W. P.; Schmidt, C. N.; Rapp, B.; Takabayashi, S.; Knowlton, W. B.; Lee, J.; Yurke, B.; Hughes, W. L.; Graugnard, E.; Kuang, W., *Nano Lett* **2013,** 13, 3850-3856.

(78) Dutta, P. K.; Varghese, R.; Nangreave, J.; Lin, S.; Yan, H.; Liu, Y., *J Am Chem Soc* **2011,** 133, 11985-11993.

(79) Langecker, M.; Arnaut, V.; Martin, T. G.; List, J.; Renner, S.; Mayer, M.; Dietz, H.; Simmel, F. C., *Science* **2012,** 338, 932-936.

(80) Chhabra, R.; Sharma, J.; Wang, H.; Zou, S.; Lin, S.; Yan, H.; Lindsay, S.; Liu, Y., *Nanotechnology* **2009,** 20, 485201.

(81) Pal, S.; Dutta, P.; Wang, H.; Deng, Z.; Zou, S.; Yan, H.; Liu, Y., *The Journal of Physical Chemistry C* **2013,** 117, 12735-12744.

(82) Acuna, G. P.; Moller, F. M.; Holzmeister, P.; Beater, S.; Lalkens, B.; Tinnefeld, P., *Science* **2012,** 338, 506-510.

(83) Berardi, M. J.; Shih, W. M.; Harrison, S. C.; Chou, J. J., *Nature* **2011,** 476, 109-113.

(84) Selmi, D. N.; Adamson, R. J.; Attrill, H.; Goddard, A. D.; Gilbert, R. J. C.; Watts, A.; Turberfield, A. J., *Nano Lett* **2011,** 11, 657-660.

(85) Delebecque, C. J.; Lindner, A. B.; Silver, P. A.; Aldaye, F. A., *Science* **2011,** 333, 470-474.

(86) Erkelenz, M.; Kuo, C. H.; Niemeyer, C. M., *J Am Chem Soc* **2011,** 133, 16111-16118.

(87) Wilner, O. I.; Weizmann, Y.; Gill, R.; Lioubashevski, O.; Freeman, R.; Willner, I., *Nat Nanotechnol* **2009,** 4, 249-254.

(88) Fu, J. L.; Liu, M. H.; Liu, Y.; Woodbury, N. W.; Yan, H., *J Am Chem Soc* **2012,** 134, 5516-5519.

(89) Fu, J.; Yang, Y.; Buck, A. J.; Liu, M.; Liu, Y.; Walter, N. G.; Woodbury, N. W.; Yan, H., *Nat Nanotechnol* **2014,** 9, 531-536.

(90) Hariadi, R. F.; Cale, M.; Sivaramakrishnan, S., *P Natl Acad Sci USA* **2014,** 111, 4091-4096.

(91) Derr, N. D.; Goodman, B. S.; Jungmann, R.; Leschziner, A. E.; Shih, W. M.; Reck-Peterson, S. L., *Science* **2012,** 338, 662-665.

(92) Endo, M.; Katsuda, Y.; Hidaka, K.; Sugiyama, H., *J Am Chem Soc* **2010,** 132, 1592-1597.

(93) Douglas, S. M.; Bachelet, I.; Church, G. M., *Science* **2012,** 335, 831-834.

(94) Lee, H.; Lytton-Jean, A. K. R.; Chen, Y.; Love, K. T.; Park, A. I.; Karagiannis, E. D.; Sehgal, A.; Querbes, W.; Zurenko, C. S.; Jayaraman, M.; Peng, C. G.; Charisse, K.; Borodovsky, A.; Manoharan, M.; Donahoe, J. S.; Truelove, J.; Nahrendorf, M.; Langer, R.; Anderson, D. G., *Nat Nanotechnol* **2012,** 7, 389-393.

(95) Liu, X. W.; Xu, Y.; Yu, T.; Clifford, C.; Liu, Y.; Yan, H.; Chang, Y., *Nano Lett* **2012,** 12, 4254-4259.

(96) Amir, Y.; Ben-Ishay, E.; Levner, D.; Ittah, S.; Abu-Horowitz, A.; Bachelet, I., *Nat Nanotechnol* **2014,** 9, 353-357.

(97) Ke, Y.; Lindsay, S.; Chang, Y.; Liu, Y.; Yan, H., *Science* **2008,** 319, 180-183.

(98) Subramanian, H. K. K.; Chakraborty, B.; Sha, R.; Seeman, N. C., *Nano Lett* **2011,** 11, 910-913.

(99) Endo, M.; Tatsumi, K.; Terushima, K.; Katsuda, Y.; Hidaka, K.; Harada, Y.; Sugiyama, H., *Angew Chem Int Edit* **2012,** 51, 8778-8782.

(100) Endo, M.; Yang, Y. Y.; Suzuki, Y.; Hidaka, K.; Sugiyama, H., *Angew Chem Int Edit* **2012,** 51, 10518-10522.

(101) Li, J.; Pei, H.; Zhu, B.; Liang, L.; Wei, M.; He, Y.; Chen, N.; Li, D.; Huang, Q.; Fan, C. H., *Acs Nano* **2011,** 5, 8783-8789.

CHAPTER 2

COMPLEX ARCHIMEADIAN TILING SELF-ASSEMBLED FROM DNA NANOSTRUCTURES

Adapted with permission from Zhang, F.; Liu, Y.; Yan, H. Complex Archimedean Tiling Self-assembled from DNA Nanostructures, *J. Am. Chem. Soc.* **2013**, 135, 7458–7461.

2.1 Abstract

Archimedean tilings are periodic polygonal tessellations that are created by placing regular polygons edge-to-edge around a vertex to fill the plane. Archimedean tilings were first classified by Johannes Kepler in 1619[1] and are still of great interest today due to the unique and interesting properties of the resulting patterns. For example, Archimedean tilings can be used to generate photonic crystals, which are periodic optical nanostructures that affect the propagation of electromagnetic waves in much the same way that semiconductors affect electrons[2]. Another report describes a specific Archimedean tiling ($3^3.4^2$) that forms a "wetting layer" between periodic and quasi-crystalline phases in a binary colloidal system[3]. Here we explored two different design methods to form Archimedean tilings: one method is employing three and four arm DNA junction tiles, with specifically designed arm lengths and inter-tile sticky end interactions, which can be used to form sophisticated two- and three-dimensional (2D and 3D) tessellation patterns; the other method is creating individual asymmetric three or four arms motifs with certain angles to form large 2D and 3D tessellations. We in total demonstrate four different complex Archimedean patterns, ($3^3.4^2$), ($3^2.4.3.4$), (4.8^2),

(3.6.3.6), and the formation of 2D lattices, 3D tubes, or sealed polygon shaped pockets from the tessellations. The successful growth of a hybrid DNA tile motif arrays and the formation of Archimedean tiling from asymmetric building blocks suggests that it may be possible to generate 2D quasi-crystals from DNA building blocks.

2.2 Introduction

Synthetic DNA molecules are powerful and effective materials for the construction of addressable 2D and 3D nanostructures[4-6], and have demonstrated their potential use in nanoelectronic, biosensing, and computational applications[7-10]. Multi-arm junctions in particular have been widely used for the assembly of various 2D and 3D structures[11-15]. These structural DNA motifs can be efficiently assembled from a small pool of short single stranded oligonucleotides (20-100 nts) and connected by programming unique inter-tile sticky end interactions to create more complex higher order structures. However, previous reports of their assembly are mostly based on repeating patterns of uniform geometric building blocks (regular tiling), which can be treated as a special case of Archimedean tiling with homogeneous vertices, tiles and edges. In this section we demonstrated two strategies to fabricate Archimedean tilings: first design approach is creating the hybridization between three and four arm DNA junction tiles with specifically designed arm lengths (geometric rules) and inter-tile sticky end interactions (matching rules); second approach is design individual asymmetric three or four arm tiles with controllable angle between adjacent arms by adjusting the poly T loops in the center of each junction.

2.3 Materials and methods

See Appendix A.

2.4 Results and Discussion

Here we utilized combinations of precisely designed three- and four-arm junction DNA tiles to generate rationally designed sophisticated 2D and 3D tessellation nanostructures. In contrast to regular tilings semi-regular Archimedean tilings are composed of more than one type of regular polygon[16] and thus require at least two unique building blocks. As shown in Figure 2.1a, b we demonstrate that two Archimedean tilings, (3^2.4.3.4) and (3^3.4^2), can be created through the self-assembly of three- and four-arm junction DNA tiles. Our results show that both tilings assemble into 2D lattice arrays with dimensions in the micrometer scale. By changing the design of the junctions (lengths of the arms and complementarity of the sticky ends) and tuning the annealing conditions it is possible to form tubes and pockets displaying the same Archimedean pattern. The successful formation of these hybrid DNA junction patterns establishes a foundation for the construction of more complex higher order DNA nanostructures or even DNA quasi-crystals.

The first step in the design process is to select desirable lengths for each of the arms of the DNA motifs' so that they are spatially compatible and facilitate connection between building blocks. Here, there are two important factors to consider: the geometry/dimensions of the desired Archimedean tiling and the 3D helical structure of double-stranded (ds) DNA. For semi-regular Archimedean patterns formed from equilateral triangle and square motifs (as shown in Figure 2.1a,b), both polygons have the

same edge lengths. The two motifs can be represented by 3-arm and 4-arm junctions and transformed into Cairo pentagonal and Prismatic pentagonal tilings by connecting the junctions. The geometric constraints of the tiling parameters require a 1.732:1 ($3^{1/2}$:1) ratio between arm lengths (4-arm:3-arm). This ratio allows the tiles to be assembled such that the arms do not overlap or have gaps between, ensuring that any compression-stretching or bending of the DNA double helices is minimized. It is also important to consider that B-form ds-DNAs are 3D molecules that exhibit 10.5 base pairs per helical turn in solution. For quasi-2D and 2D structures the distance between the crossovers in adjacent tile arms is restricted to even (all tiles face up) or odd (alternating face up and face down) numbers of half turns, respectively. Adhering to these restrictions will avoid deviation of the self-assembled structures from the desired quasi-2D and 2D patterns, and minimize over- or under-twisting of the DNA strands.

With these two considerations in mind, three different combinations of arm lengths (defined in number of helical turns) corresponding to the 4-arm and 3-arm junction tiles were evaluated: 4.5:2.5 (=1.80), 3.5:2.0 (=1.75) and 4.0:2.25 (=1.78). The ratio of arm lengths (4-arm:3-arm) ranges from 1.75 to 1.80. The corresponding tiling patterns are shown in Figure 2.1a-e. These length ratios satisfy the structural requirements of DNA double helices such that the edge-to-edge distances between the junction points in the assemblies are either whole or half helical turns and therefore, adjacent DNA tiles are either facing the same direction or opposite faces of the same 2D plane. In addition, these combinations of arm lengths are relatively close to the ideal geometric ratio of 1.732. Due to the inherent flexibility and soft materials properties of DNA, this small discrepancy can apparently be accommodated. We restricted the arm

lengths (maximum 9 full turns or 31 nm between the vertices) in order to maintain the rigidity of DNA.

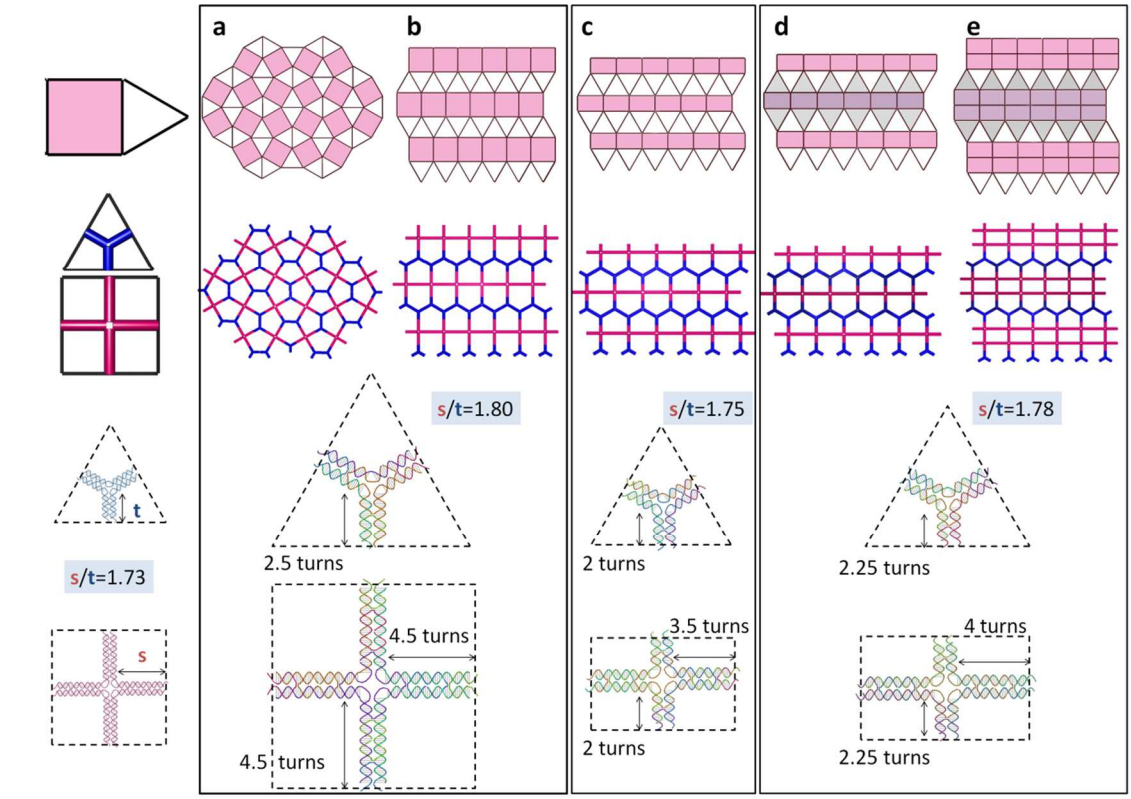


Figure 2.1 Designing the length of the arms in DNA junction tiles to create different Archimedean tiling patterns. (a) Upper panels: Archimedean tiling pattern ($3^2.4.3.4$) and the corresponding transformed Cairo pentagonal tiling that can be represented by 3-arm and 4-arm junction motif DNA tiles with 2.5 and 4.5 turn arm lengths (lower panels), respectively, at a ratio of 1:1.80. (b) Upper panels: Archimedean tiling pattern ($3^3.4^2$) and the corresponding transformed Prismatic pentagonal tiling that can be represented by 3-arm and 4-arm junction motifs with the same arm lengths (lower panels). (c) Upper panels: shortened Archimedean tiling ($3^3.4^2$) and transformed shortened Prismatic pentagonal tiling, in which the multi-arm junction motifs have 2 and 3.5 turn arm lengths (lower panels), respectively, at a ratio of 1:1.75. Note that the length of the arms in the 4-arm junction tile is shorter (2 turns) in the vertical direction. (d) Upper panels: shortened Archimedean tiling with a corrugated design and shortened Prismatic pentagonal tiling with neighboring layers of unit cells (grey indicates facing down) facing in opposite directions, respectively. Here the multi-arm junction motifs have arm lengths of 2.25 and 4 turns (lower panels), respectively, at a ratio of 1:1.78. (e) Upper panels: the more complex 3-isogonal tiling when extra layer of rectangles tiles are included and the corresponding transformed 2-uniform tiling, respectively. Lower panels: the same multi-arm junction motifs as those in d are used to create this pattern.

In the non-isotropic tilling pattern shown in Figure 2.1b-e the four sides of the squares do not have to be equal length; the squares can be replaced by rectangles, or multiple layers of rectangles can be introduced without interrupting the overall periodic lattice growth. Here, the only geometric requirement is that the side of the rectangle that is in contact with the triangle motif must be equal to the length of the equilateral triangle. Meanwhile, the sides that are in contact with other squares or rectangles can be any length that satisfies the requirements imposed by the properties of the DNA double helices and crossover patterns.

The second step in the design process is to identify the matching rules corresponding to the desired patterns and encode the specific inter-unit interactions within the sticky-ends of the tiles. Archimedean tiling has translational periodicity based on “unit cells”. We can determine the “unit cell” within each pattern and use the DNA multi-arm junction motifs to physically construct these unit cells. By considering the symmetry of the unit cells we can specify the unique sticky ends and minimize the number of different DNA building blocks required. For example, the unit cell of the pattern shown in Figure 2.2b (also called Prismatic pentagonal tiling with prismatic pentagon shaped cavities) is an elongated hexagon which can be constructed from two 3-arm motifs and one 4-arm motif. Here, we designed a single 3-arm motif (instead of two) that contains one arm (1*) that interacts with arm 1 from the 4-arm motif, and two arms that are self-complementary (2/2*). Meanwhile, the 4-arm motif contains two opposite arms (1) that can be connected with arm 1* of the 3-arm motif, and two opposite arms that are self-complementary (3/3*). We anticipated that mixing these two tiles in a 2:1

molar ratio would result in the self-assembly of the units into the pattern depicted in Figure 2.2b.

However, symmetry within the DNA motifs can increase the possibility of mismatched interactions. It is important to carefully balance the simplicity of the building blocks with the ability to form a unique pattern. For example, when we attempted to construct the tiling pattern shown in Figure 2.2a, also called a Cairo pentagonal tiling, from a single 4-arm motif (instead of two) with the same sticky ends (1=2 and 1*=2*) we did not obtain the expected pattern and only small mismatched pieces were observed (see images in supplementary Figure S1). It is likely that the symmetry of the 4-arm tile motif reduced the probability of forming a unique structure. Here, the unit cell of the desired pattern includes cyclization of two 4-arm junctions and three 3-arm junctions in a 3-3-4-3-4 pattern to form an asymmetric pentagon. The symmetry of the sticky end sequences in the 4-arm tile may promote the exclusion of one of the 3-arm motifs such that the remaining 4 tiles connect in a 3-4-3-4 pattern to form a square (or rhombus) unit. Deformation of the junction angles is possible as the arms of these 4-arm and 3-arm DNA junction motifs are relatively long and less rigid than shorter ones, and the angles at the branch points of each individual tile may be flexible enough to deviate from the expected 90 and 120 degree angles, respectively. The formation of a 4-member ring is kinetically more favorable than a 5-member ring; however, the 4-member unit cells do not grow into large 2D arrays due to excess structural strain. Even when the correct numbers of junction motifs combine to form the expected unit cells, they will not be able to assemble with perfect edge-to-edge tiling in the presence of incorrectly formed 4-member rings. Thus, designing building blocks with precisely encoded sticky-ends is essential for the

successful formation of a desired pattern. In the case described above we were required to use two unique 4-arm tiles and one unique 3-arm tile to realize the design. By mixing these three unique tiles in a 1:1:4 molar ratio we facilitated the self-assembly of the unit motifs into the expected pattern shown in Figure 2.2a.

The final step in the design process is to assign sequences to the ssDNA that comprise each structural motif and the corresponding sticky ends. We found that asymmetric sequences are required for the arms, even for cases when the sticky-ends are the same. For example, in the Cairo pentagonal tiling (Figure 2.2a) each 4-arm tile has four identical sticky-ends but still requires four different sequences for the branches to avoid aggregation at the individual tile level (see Supplementary Figure S10b). This is likely because the length of the arms used here are much longer than the ones described previously[12-13, 15, 17] such that the strands with repeating sequences have a greater chance to be linked to other tiles. Obviously the sequences of the sticky ends should be distinct to avoid mismatches between building blocks. It is also important to note that the sequences of the sticky-ends should not be rich in G or C residues, otherwise, undesired oligomerization of the individual tile motifs may occur (Supplementary Figure S10c). For example, we observed that the 4-arm junction tiles with four identical sticky-end sequences (GCAG) self-associate in an end-to-end manner to form linear oligomers ranging from dimers to tetramers, with the final assemblies resembling rhombus-like ribbon structures where each tile exhibits a twisted junction (supplementary Figure S10d). Thus, we avoided GC rich sequences in the sticky end design. In the work reported here we utilized 4-bp sticky ends throughout. 4^4 (= 256) total possibilities provide adequate sequence space for the selection of unique inter-unit complementarity.

Experimentally, each structure shown in Figure 2.2 was assembled by mixing all the strands needed in the correct stoichiometric ratio for a final unit-cell concentration of 0.6 μM, in tris-acetate–EDTA (TAE) buffer (pH 8.0, containing 12.5 mM Mg^{2+}) and annealing the mixture from 90 to 4 °C over 12 hours (see Supplementary Information for detailed experimental methods).

Figure 2.2a demonstrates the formation of the Cairo pentagonal tiling corresponding to the Archimedean pattern (3^2.4.3.4) in which the 3-arm and 4-arm building blocks are combined in a 4:1:1 ratio. AFM characterization of the products confirms that the tiles self-assembled into micrometer-sized 2D arrays. The AFM images also reveal that the 2D arrays often exhibit curved edges, indicating that they are not perfectly planar in solution before deposition onto mica (see additional images in Figure S2a). This curvature is likely an intrinsic consequence of the design, as the tiles are all facing the same direction and thus any curvature in the individual tiles may be accumulated in the 2D array. However, the relatively large size of the arrays (with dimensions of several micron meters) indicates that the curvature of the unit tiles, if any, is small (< 1 degree per tile).

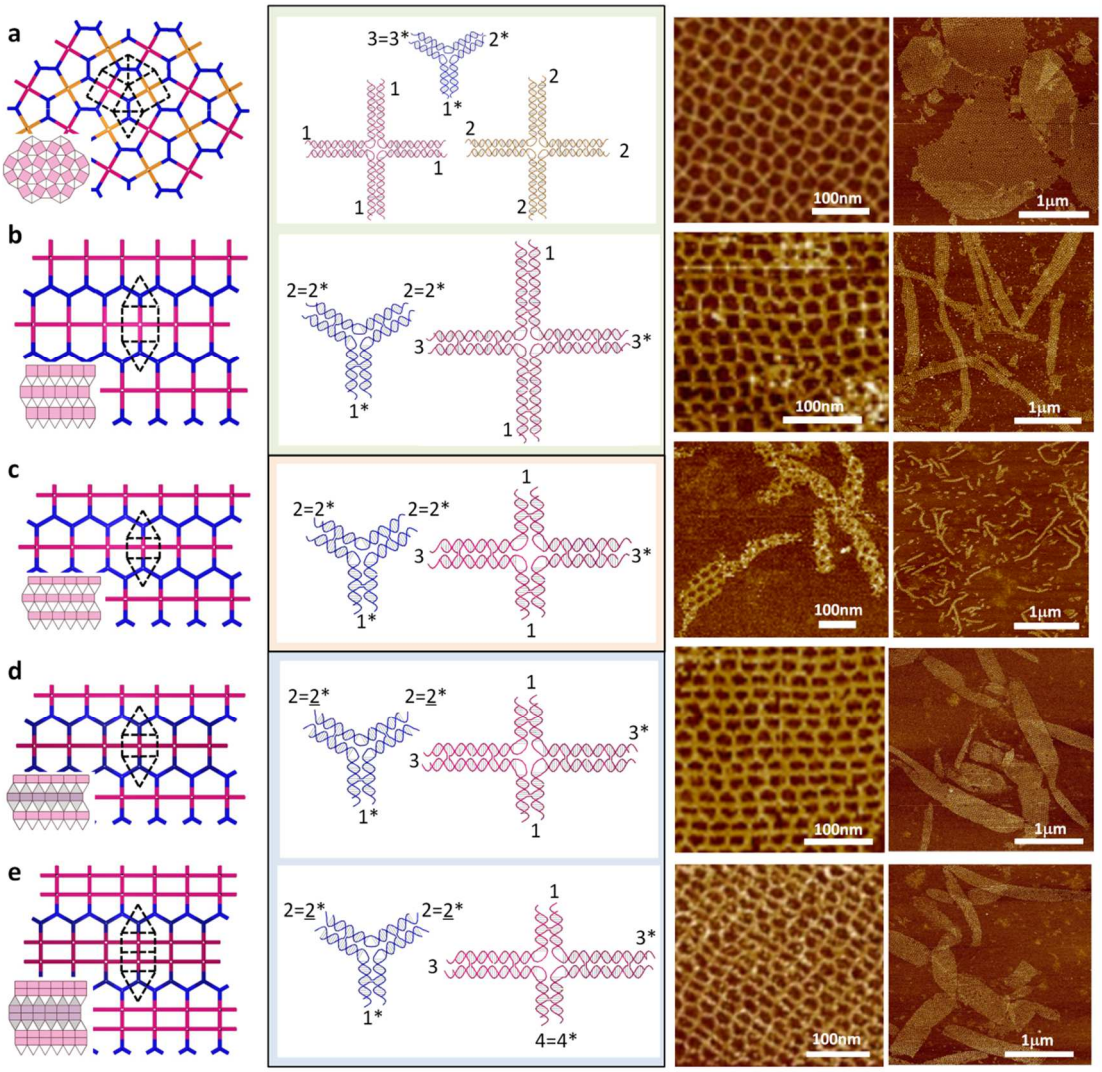


Figure 2.2 Sticky end matching rules and corresponding AFM images of each Archimedean tiling design. (a-e) The left panel illustrates the unit cell represented by dashed lines. The middle panel depicts the sticky ends matching rules: n sticky end interacts with n*, the underlined numbers represent tiles that are connected to other tiles that face the opposite direction in the array (half turn between vertices in the DNA arms). The right panels contain zoom in and zoom out AFM images, respectively, with the scale bars marked. (a) Cairo pentagonal tiling corresponding to Archimedean tiling (3^2.4.3.4). (b) Prismatic pentagonal tiling corresponding to Archimedean tiling (3^3.4^2). (c) Shortened Prismatic pentagonal tiling. (d) Shortened Prismatic pentagonal tiling with corrugated design. (e) 2-uniform tiling with extra layer of rectangular tiles and corrugated design.

Figure 2.2b illustrates the formation of the Prismatic pentagonal tiling corresponding to the Archimedean pattern (3^3. 4^2). Its two building blocks have the same

arm lengths as those in the Cairo pentagonal tiling (Figure 2.2a), but with different sticky end sequences as determined by its own matching rules. Similarly, these tiles are all facing in the same directions in the array. When mixed in a molar ratio of 2:1 they self-assemble into large 2D sheets that curl up into tubes with diameters in the range of 80-250 nm (see schematics in supplementary Figure S3c and additional AFM images in Figure S3d).

Tube formation from DNA tile arrays has been discussed previously[18-20]. This process is thermodynamically allowed as long as the enthalpy gained from DNA hybridization at the edges of the tile arrays is sufficient to compensate for the lost in entropy and the energetic cost of bending the helices within the tile arrays. Based on the designed connection pattern there are four possible ways to fold the 2D array into a tube (schematics shown in Figure S3a). We carefully analyzed AFM images of 88 tubes and found that ~ 57% of the tubes have a long axis that is parallel to the connections between the 4-arm junction tiles (Figure S3e). ~ 43% of the tubes adopted a spiral arrangement, with the long axis of the tube at a < 45 degree angle with the connections between the 4-arm junction tiles. There is no evidence of the formation of structures in which the axis of the tube is perpendicular to the connections between the 4-arm junctions tiles, nor are there any tubes that adopt >45 degree angles with respect to the inter-tile connections.

This observation can be explained in terms of the anisotropic growth dynamic of the tile arrays. The growth rate parallel to the direction of the 4-arm tile-tile association is expected to be much faster than in the perpendicular direction. This is because the rate of growth in the array per building block is faster in parallel than that in the perpendicular direction, as the 4-arm junction tiles are larger in size and require a single pair of sticky

end interactions per tile to secure the subsequent layer of tiles. In contrast, the growth rate in the perpendicular direction per unit tile is smaller, not only because each 3-arm tile is shorter, but also because growth in that direction requires at least two successful pairs of sticky end connections per tile to secure the next layer of tiles. This growth dynamics causes the anisotropic elongation of the tile array that we observed. Given enough time during the annealing process, the 2D tile arrays will reach a point at which it is more difficult and energetically unfavorable for the tiles along the edges to encounter complementary tiles in solution than it is to interact with other edges of the array and form tubes. It is also possible for ring structures to form during the nucleation step; these structures can serve as templates that can be elongated from both ends resulting in the formation of tubes. Shorter tubes may also be connected end-to-end during the last stage of the annealing process to form longer tubes.

Figure 2.2c shows a shortened Prismatic pentagonal tiling using the same matching rules as shown in Figure 2.2b, but with shorter arms in both of the component tiles (less an integer numbers of half turns in each arm). Therefore, the tiles are all facing in the same direction but the anisotropy in the dimensions is higher. The building blocks were found to connect and curl into tubes with relatively small diameters, ~ 43 nm (supplementary Figure S4e). The narrow tubes form within 2 hours and grow longer with extended annealing times (see Supplementary Information for detailed experimental methods and supplementary Figure S4b). This observation supports a nucleation and growth mechanism. The tube folding direction is similar to the Prismatic pentagonal design and can be explained by the same reasons as discussed above (supplementary Figure S4).

Figure 2.2d is another shortened Prismatic pentagonal tiling, but with a corrugated design in which the lengths of the arms are adjusted according to Figure 2.1d so that the neighboring layers of unit cells are alternatively facing up and down. This design balances the natural curvature within the building blocks and leads to formation of large 2D arrays. Interestingly, wide tubes (additional AFM images shown in supplementary Figure S5) with diameters (or half perimeters) ranging from 100-400 nm (supplementary Figure S5c) were observed. It appears that the ability to form tubes cannot be prevented with a corrugated design. This result indicates that the 2D arrays are flexible enough that bending them incurs a smaller energetic penalty than the energy released from base-pairing.

As shown in Figure 2.2e we further modified the 4-arm building block so that only one of the sticky ends was self-complementary (4=4*) (compared with the design in Figure 2.2d), and upon mixing the 3- and 4-arm tiles in a 1:1 ratio they self-assemble into a complex 2D tiling. This is also a corrugated design with neighboring layers of unit cells alternatively facing opposite directions. The arrays formed from this design also curled into tubes with similar diameters 100-450 nm (supplementary Figure S6c) as those in Figure 2.2d. The cavities in this pattern exhibit both rectangular and prismatic pentagon shapes.

We further investigated the parameters that influence the formation of large patterns, including the molar ratio of the building blocks, annealing program and concentration. For the shortened Prismatic pentagonal tiling with a corrugated design and the following assembly conditions: short annealing time (2 h instead of 12 h), low concentration of building blocks (0.2 μM instead of 0.6 μM), and deviation from the

desired stoichiometric ratio of the building blocks (1:0, 0:1, 1:1 or 1:1.9 instead of 1:2), only small fragments were observed (supplementary Figure S7-8). However, for the Cairo pentagonal tiling when the ratio of building blocks was varied from the designed 4:1:1 (supplementary Figure S9) to 8:1:1 or 2:1:1, we observed the formation of small, 2-layer structures with sharp edges and dimensions in the range of 200-500 nm. These pocket-like structures were also observed in the background of the large 2D array samples when a 4:1:1 ratio was used (supplementary Figure S2b), possibly due to the imperfect stoichiometric ratio. The pockets are likely to adopt the observed shapes with certain preferred angles. Figure 2.3 illustrates the possible mechanism of folding when two complementary edges with 90, 180 or 270 degree angles come together to form the pocket-like structures; after they are deposited on 2D substrates for AFM imaging they form two layer structures with sharp 45, 90 or 135 degree angles.

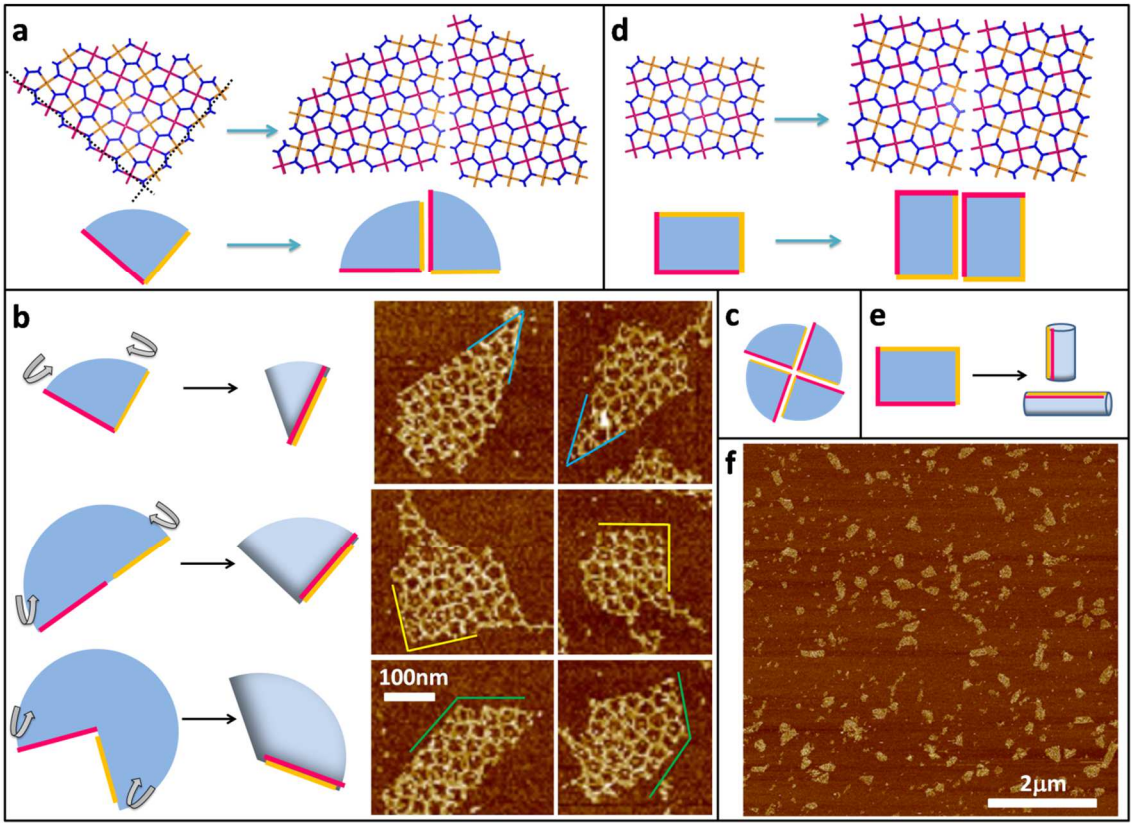


Figure 2.3 Possible mechanisms of the formation of 2-layer pocket-like structures. (a) In the Cairo pentagonal tiling two edges are arranged at a 90 degree angle and are complementary to each other. (b) Three possible folding interactions with 90, 180 and 270 degree inter-edge angles, respectively, that are transformed into 45, 90 and 135 degree angles. The corresponding structures are shown in the AFM images on the right. (c) Based on a, four sets of matching edge interactions point to a center. (d) Two parallel edges match each other to bring two smaller pieces together to form a larger piece. (e) Parallel edges allow a 2D array to fold into a tube. (f) AFM image of the sample obtained from a 0.5:0.5:4 ratio of building blocks (the ideal ratio for the large 2D array is 1:1:4).

2.5 Conclusion

In 2008[21] researchers determined that the intermediate between a crystal and a quasi-crystal is a ($3^3.4^2$) Archimedean-like tiling structure, which was observed in a colloidal monolayer interacting with a quasi-crystalline substrate. Later, a link between Archimedean tilings and quasi-crystals was established when the self-assembly of binary nanoparticles resulted in the formation of quasi-crystalline super-lattices with a ($3^3.4^2$)

Archimedean structure interface between the quasi-crystalline and crystalline phases[3]. It is foreseeable that the successful hybridization of DNA motifs to form Archimedean tiling structures will further increase the complexity of DNA nanostructures and provide the ability to form quasi-crystals based on DNA tiling. Furthermore, DNA directed assembly of quasi-crystalline arrays may produce unique nanostructures with novel properties by functionalizing the DNA tile motifs with other nano-materials.

2.6 References

(1) Grünbaum, B.; Shephard, G. C., *Tilings and Patterns*. W.H. Freeman: 1987.

(2) David, S.; Chelnokov, A.; Lourtioz, J. M., *Ieee J Quantum Elect* **2001,** 37, 1427-1434.

(3) Talapin, D. V.; Shevchenko, E. V.; Bodnarchuk, M. I.; Ye, X. C.; Chen, J.; Murray, C. B., *Nature* **2009,** 461, 964-967.

(4) Seeman, N. C., *J Theor Biol* **1982,** 99, 237-247.

(5) Rothemund, P. W. K., *Nature* **2006,** 440, 297-302.

(6) Winfree, E.; Liu, F. R.; Wenzler, L. A.; Seeman, N. C., *Nature* **1998,** 394, 539-544.

(7) Seeman, N. C., *Nature* **2003,** 421, 427-431.

(8) Gothelf, K. V.; LaBean, T. H., *Org Biomol Chem* **2005,** 3, 4023-4037.

(9) Voigt, N. V.; Torring, T.; Rotaru, A.; Jacobsen, M. F.; Ravnsbaek, J. B.; Subramani, R.; Mamdouh, W.; Kjems, J.; Mokhir, A.; Besenbacher, F.; Gothelf, K. V., *Nat Nanotechnol* **2010,** 5, 200-203.

(10) Deng, Z. T.; Samanta, A.; Nangreave, J.; Yan, H.; Liu, Y., *J Am Chem Soc* **2012,** 134, 17424-17427.

(11) Yan, H.; Park, S. H.; Finkelstein, G.; Reif, J. H.; LaBean, T. H., *Science* **2003,** 301, 1882-1884.

(12) He, Y.; Chen, Y.; Liu, H. P.; Ribbe, A. E.; Mao, C. D., *J Am Chem Soc* **2005,** 127, 12202-12203.

(13) He, Y.; Tian, Y.; Ribbe, A. E.; Mao, C. D., *J Am Chem Soc* **2006,** 128, 15978-15979.

(14) He, Y.; Tian, Y.; Chen, Y.; Ribbe, A. E.; Mao, C. D., *Chem Commun* **2007**, 165-167.

(15) Zhang, C.; Su, M.; He, Y.; Zhao, X.; Fang, P. A.; Ribbe, A. E.; Jiang, W.; Mao, C. D., *P Natl Acad Sci USA* **2008,** 105, 10665-10669.

(16) Chavey, D., *Comput Math Appl* **1989,** 17, 147-165.

(17) He, Y.; Ye, T.; Su, M.; Zhang, C.; Ribbe, A. E.; Jiang, W.; Mao, C. D., *Nature* **2008,** 452, 198-201.

(18) Ke, Y. G.; Liu, Y.; Zhang, J. P.; Yan, H., *J Am Chem Soc* **2006,** 128, 4414-4421.

(19) Maune, H. T.; Han, S. P.; Barish, R. D.; Bockrath, M.; Goddard, W. A.; Rothemund, P. W. K.; Winfree, E., *Nat Nanotechnol* **2010,** 5, 61-66.

(20) Mitchell, J. C.; Harris, J. R.; Malo, J.; Bath, J.; Turberfield, A. J., *J Am Chem Soc* **2004,** 126, 16342-16343.

(21) Mikhael, J.; Roth, J.; Helden, L.; Bechinger, C., *Nature* **2008,** 454, 501-504.

CHAPTER 3

COMPLEX WIREFRAME DNA ORIGAMI NANOSTRUCTURES WITH MULTI-ARM JUNCTION VERTICES

3.1 Abstract

Engineering complex wireframe architectures with arbitrarily designed connections between selected vertices in three-dimensional (3D) space is an important challenge in nanotechnology. Here we present a new design strategy to realize the formation of finite-size wireframe DNA nanostructures with high complexity and programmability. In this design, the vertices are represented by n × 4 multiarm junctions (n = 2-10) with controlled angles, and the lines are represented by antiparallel DNA crossover tiles of variable lengths. Scaffold strand(s) were used to integrate the vertices and lines into fully assembled structures displaying intricate architectures. A series of two-dimensional (2D) patterns ranging from symmetrical lattice arrays, quasicrystalline structures, curvilinear arrays of variable curvatures to a flower-and-bird artistic structure, a complex 3D snub cube, and a 3D Archimedean solid with 2D to 3D transitions were constructed to demonstrate the versatility of the design strategy.

3.2 Introduction

Programmable DNA self-assembly has shown its true merit in constructing designer nanoarchitectures of increasing complexity [1-4]. Over the years, various design strategies have been devised to create DNA nanostructures with well-defined geometrical features[5-11]. In particular, DNA origami[6] has opened up exciting opportunities to create molecular scaffolds for the templated assembly of functional molecular devices and

sensors[4]. DNA origami is a method that employs hundreds of short, synthetic DNA strands as staples to fold a long single-stranded scaffold into spatially addressable two- and three-dimensional (2D and 3D) DNA architectures (*6*). Most of the current methods in DNA origami rely on restricting the scaffold strand to discrete domains of adjacent helices arranged in a parallel fashion[6, 8, 11-13]. A recent departure from these methods is the creation of DNA Gridiron nanostructures, in which an unusual set of four-arm Holiday junctions connects double helical domains into grid-like structures that maintain local regularity and translational symmetry[14]. Nevertheless, achieving wireframe structures of higher-order arbitrariness and complexity with broken local periodicity and symmetry requires a new set of design rules to allow the scaffolds to travel in arbitrary directions through chosen points in the designed geometry.

Here we present systematic design rules to construct sophisticated wireframe structures without local symmetry restrictions where each vertex and line segment is individually designed and controlled. The length and curvature of the line segments that make the connections between any neighboring vertices can be modified. The number of arms that stretch out from each vertex can be anywhere from 2 to 10, and the angles between any two adjacent arms can be varied. We have designed and constructed a set of complex 2D patterns including symmetrical lattice arrays, quasicrystalline structures, curvilinear arrays, and a simple wire art sketch in the 100-nm scale, as well as 3D objects including a snub cube with 60 edges and 24 vertices and a reconfigurable Archimedean solid that can be controlled to make the unfolding and refolding transitions between 3D and 2D.

3.3 Materials and Methods

3.3.1 Materials

All DNA staple strands were purchased in 96-well plates from Integrated DNA Technologies, Inc. (www.IDTDNA.com) at 25 nmol synthesis scales with concentrations normalized to 200 μM. Staple strands were used directly from plates without further purification. Scaffold single-stranded M13mp18 viral DNA and phiX174 DNA were purchased from New England Biolabs, Inc. (NEB, catalog numbers N4040S and N3023S).

3.3.2 Individual Wireframe DNA Origami Nanostructure Assembly

The structures with a single scaffold were assembled by mixing 5 nM single-stranded M13mp18 DNA (7,249 nucleotides) with a 10-fold molar excess of staple strands in 1× TAE Mg^{2+} buffer (40 mM Tris, 20 mM acetic acid, 2 mM EDTA, and 12.5 mM Magnesium acetate [pH 8.0]). The frame origami structures with two scaffolds were annealed by mixing 5 nM M13mp18 and 5 nM phiX174 with the staple strands in a 1:1:20 molar ratio in 1× TAE Mg^{2+} buffer. The final volume of the mixture was 100 μL. The design details and sequences of the DNA oligos used to form each structure are listed in Section 5. The resulting solutions were annealed in a PCR thermocycler from 90 ºC to 4 ºC in about 12 hours: 90 ºC to 86 ºC at a rate of 4 ºC per 5 minutes; 85 ºC to 70 ºC at a rate of 1 ºC per 5 minutes; 70 ºC to 40 ºC at a rate of 1 ºC per 15 minutes; 40 ºC to 25 ºC at 1 ºC per 10 minutes; and hold at 4 ºC at the end of the cycle.

3.3.3 Hierarchical Assembly of 3×3 Square Lattice Origami Array

The three-square lattice origami with different extended sticky-end staple strands were individually annealed by following the protocol described above. The excess staple

strands were washed away using a 100-kD MWCO Amicon centrifugal filter (Sigma-Aldrich). A 100-μL annealed sample and 400 μL 1× TAE-Mg^{2+} buffer was added into the filter. After spinning for 3 minutes at 5500 rpm, another 400 μL 1× TAE-Mg^{2+} buffer was added for the second wash. After three washes, the three structures with the appropriate stoichiometry (1:4:4) were mixed together and left at room temperature (25 ºC) for 12 hours before characterization.

3.3.4 Transformation of the Cuboctahedron between the 2D Open Net and 3D Shape

The starting structure either in 2D or 3D was annealed using the protocol described above. Without further purification, 2-fold molar excesses of fuel strands were added to the sample and incubated at 45 ºC for 6 hours. Then, 4 times molar excess of set strands were added to the sample and cooled from 45 ºC to 4 ºC at a rate of 1 ºC per 10 minutes.

3.3.5 AFM Imaging

AFM imaging was conducted in "ScanAsyst mode in fluid" (Dimension FastScan, Bruker Corp.) with Scanasyst-Fluid+ tips (Bruker Corp.). The sample preparation for AFM imaging was as follows: a 2 μL annealed sample was deposited onto a freshly cleaved mica surface (Ted Pella, Inc.), and 3 μL $NiCl_2$ (25 mM) was immediately added to the samples (the effect of adding $NiCl_2$ is discussed in supplemental section 3). After waiting ~30 seconds for adsorption to the mica surface, 80 μL 1× TAE-Mg^{2+} buffer was added to the sample, and an extra 40 μL of the same buffer was deposited onto the AFM tip.

3.3.6 Cryo-EM Imaging

Snub cube concentration was 5nM after annealing. The sample was then concentrated with 100K membrane tubes (Millipore Amicon Ultra, Ultracel 100K Membrane) from 1700 μL to 45 μL. The expected final concentration was about 180 nM. 3 μL of the concentrated sample solution were spread onto negative glow discharged Quantifoil grid, then plunge-frozen with vitrobot. Data were recorded using a Gatan 4,096 × 4,096 pixel charge-coupled device (CCD) camera with 29k Magnification, 8μm defocus, 25.2e/A^2 dose, in a Titan Krios transmission electron microscope with field-emission gun operating at 300 kV accelerating voltage.

3.3.7 Single-particle Reconstruction

3D reconstructions study of the DNA cage was carried out with the single-particle image processing software EMAN. OCT symmetry was applied during the reconstruction. 3D maps were visualized using UCSF Chimera software.

3.3.8 Native Agarose Gel Electrophoresis

Native gel electrophoresis was conducted on 0.5% agarose gel in 1× TAE-Mg^{2+} buffer at 80 V sitting in an ice-water bath. Next, 5 μL annealed sample was mixed with 1 μL 100× SYBR Gold nucleic acid gel stain solution (Life Technologies), and the mixture was loaded into the casted gel well. After running for 2–3 hours, the gel was visualized under UV light (Gel Doc XR+ system gel imager, Bio-Rad).

3.4 Results and Discussion

To construct an arbitrarily shaped wireframe architecture, the first step is to convert all the connections between vertices into double lines (Figure 3.1A). The second step is to “loop” and “bridge” all the lines into a single continuity along which one single-

stranded scaffold strand can travel through all the vertices. Proper crossovers between the lines are placed to make the scaffold strand go through all the lines unidirectionally, once and only once, so that in any individual line segment the two lines are antiparallel (Figure 3.1A). Next, the complementary staple strands in the line segments are added following the rules for generating double crossover (DX) DNA tiles[5], where antiparallel DX junctions between the lines are inserted to bridge the two DNA helices separated by a full number of DNA helical turns.

To adjust the angles between arms and assign the sequences, a T_n loop of appropriate length is inserted into the staple strands surrounding the vertex, and a certain number of nucleotides in the scaffold strand are left unpaired opposite to the T_n loop. The added unpaired nucleotides at the vertex provide a degree of structural flexibility and allow bending at the corners to generate the desired angles between the arms at the vertex (Figure 3.1B). Additionally, the design of the staple strands requires consideration of the length and curvature of each of the line segments. The lengths of the line segments between any two neighboring vertices need to be an integer number of DNA helical turns to minimize strain and allow the scaffold DNA to follow the designed folding path. Figure 3.1C shows two possible scenarios when adding staple strands: with or without a scaffold crossover in the line segment, which requires a shift of the crossover positions of the staple strands to avoid juxtaposing two crossovers. This distributes the positions of crossovers evenly along the line segments, which determines the integrity and rigidity of each part of the structure.

To demonstrate the design principles, we first construct three Platonic tessellation patterns with homogenous vertices and edges of regular polygonal patterns in the form of

hexagonal, square, or triangle geometries. The honeycomb pattern containing 16 hexagons is used as an example to illustrate the detailed design process. In this pattern, each vertex is joined by three arms with equal 120° angles. First, we layout the target pattern and convert each arm into two parallel lines (figure S1A). Then we connect each pair of the lines that meet at the vertex and loop them all at the edges. This results in a number of individual hexagonal loops in the center part and a continuous loop along the perimeter of the whole pattern (figure S1B). Next, we insert DXs between the neighboring hexagon loops at selected positions so that the individual loops are all connected to form a single loop. A final bridge is added between the internal and the edge loops (figure S2); therefore, one single-stranded circular DNA acting as a scaffold can go through the entire structure and visit each vertex and line three and two times, respectively.

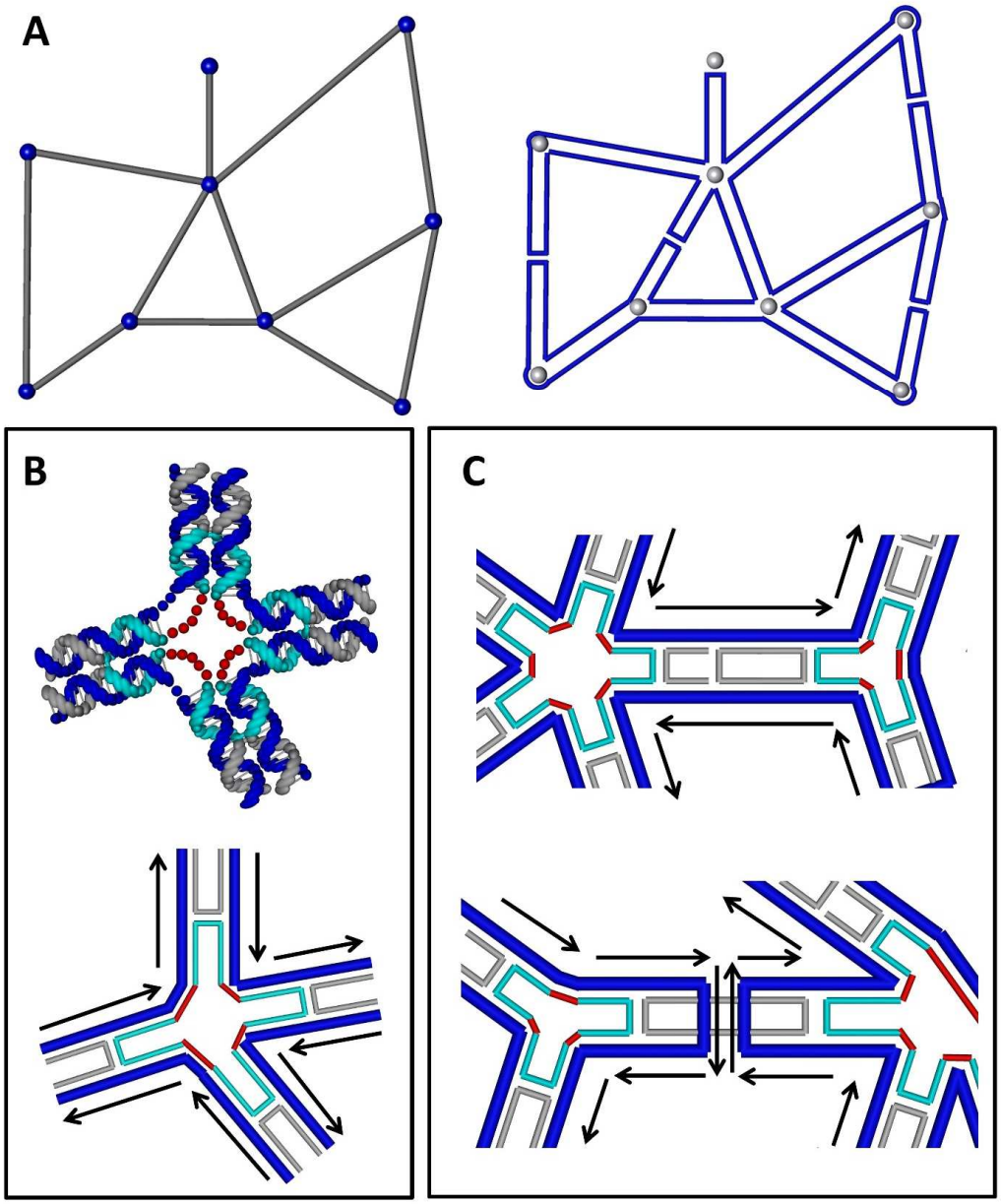


Figure 3.1 Design principles. (A) Left: An arbitrary wire-frame pattern composed of line segments (gray) and vertices (blue). Right: Steps to route a scaffold: first, double all the line segments from the original pattern; second, connect the lines that meet at each vertex; third, "loop" and "bridge" all the lines into one continuous scaffold. (B) A DNA helical model of a 4 × 4 junction (top) and a line model of the 4 × 4 junction (bottom). Each vertex is designed as an n × 4 junction. The angle of adjacent arms in one junction can be adjusted by inserting poly T loops (red dots) and leaving unpaired nucleotides in the scaffold strand opposite to the poly T loop (dark blue dots). (C) Adding staple strands on two different types of edges (5-turn long edges are utilized here for illustration): the edge with two antiparallel scaffolds (top) and the edge with a scaffold bridge (Holliday

junction) in the middle. The arrows point to the direction from 5' to 3' end of the DNA. In B and C, The dark blue strands represent the scaffold strand, and the gray and cyan strands are the staple strands.

The next step is to add the complementary staple strands in the line segments according to the design rules of forming double-crossover tiles. Here we used five full turns of B-form DNA as the length of each edge. Unpaired nucleotides around the vertices are manually assigned to satisfy the angle requirement. Because the honeycomb array has a homogenous 120° intersection angle, a T_3 loop in the staple strands at each corner is suitable to maintain this angle [15], and all the bases in the scaffold strand remain base paired (figure S3B). The edges are treated differently depending on whether or not they contain a scaffold crossover (see figure S4 for design details). Finally, nick points are added at carefully selected positions to obtain appropriate-length staple strands.

To generate the honeycomb structure (Figure 3.2A), 7100 of 7249 nt of the M13mp18 single-stranded DNA (ssDNA) were assigned, and 223 staple strands with lengths ranging from 20 to 52 bases were used. The extra 138 nt of the scaffold is left as an unpaired loop on the outer edge. All the staple strands on the outermost edges were modified with poly T tails (figure S4D) to prevent the potential π-π stacking along blunt-ended helices between individual structures.

Similar design principles were applied to create the other two Platonic patterns. The square tiling has 90° intersection angles, and four T_4 loops were utilized in the center of each vertex to create the 90° angles [16] (figure S11B). The triangular tiling has 60° intersection angles, and six T_5 loops were used at each vertex [17] (figure S11C). The dimensions of the final pattern depend on the length of the scaffold strand available and the length of the edges (minimal three helical turns). By design, the dimensions of the 4 ×

4 hexagonal pattern is 150 nm × 150 nm (considering that the width of a DX tile is ~4.5 nm), the 5 × 5 square pattern is 108 nm × 108 nm, and the triangular pattern is 136 nm × 136 nm in the 2D plane. Atomic force microscope (AFM) images confirmed the successful formation of the designed structures (Figure 3.2A-C) with high yield and structural integrity.

Our strategy can be easily adapted to design larger and more complex structures based on multiple scaffolds. For example, we designed a square tiling pattern using both M13mp18 (7249 nt) and PhiX174 (5386 nt) as the scaffold strands to create a 2D grid containing 8 × 9 cross-shaped vertices (or 7 × 8 square cavities). To ensure structural integrity, the contacts between the two scaffolds are maximized via staple strands. The final structure has four helical turns in each edge and provides a fully addressable square lattice of 148 nm × 167 nm (Figure 3.2D). To further scale up the square lattice pattern, we took inspiration from our previously reported symmetric tile hierarchical assembly method [18] to assemble a square lattice containing 17 × 17 square cavities of ~300 nm in both dimensions (Figure 3.2E). To our knowledge, this is the largest finite size square DNA lattice reported in the literature.

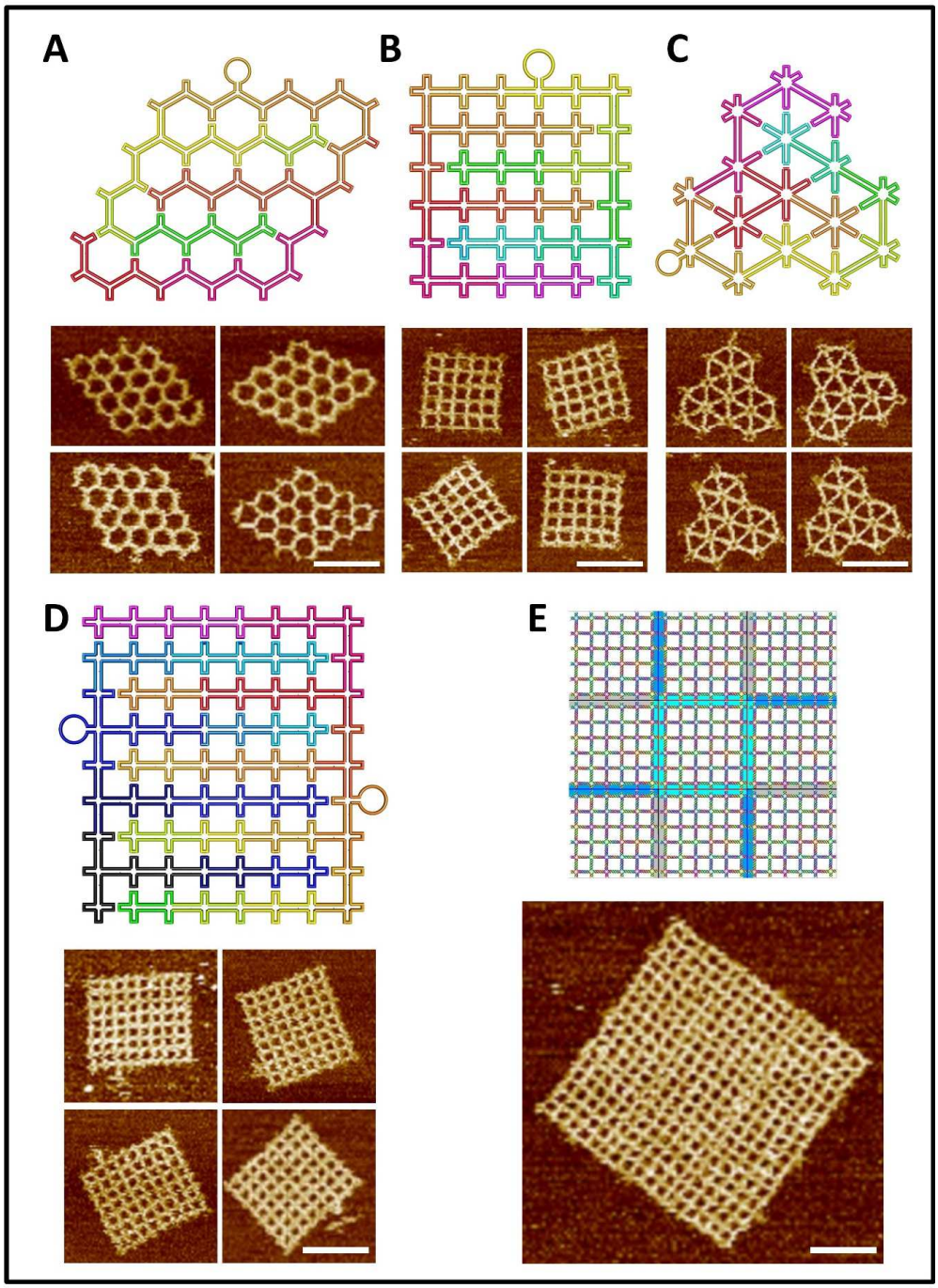


Figure 3.2 The scaffold folding paths and representative AFM images for the simple Platonic tiling. (A-C) Platonic tiling based on hexagon, square, and triangle geometries. (D) A 7 × 8 square Platonic tiling composed of two scaffolds (The black-blue loop on the left is a PhiX174 scaffold, and the colorful loop on the right is an M13mp18 scaffold). (E) A lattice with 17 × 19 square cavities. All scale bars are 100 nm.

Using this design strategy, we were able to create more complex wireframe structures without local translational symmetry (as shown in Figure 3.1A) using quasicrystalline structures as target patterns. We demonstrated three patterns without translational symmetry. These include a star shape, a 5-fold Penrose tiling, and an 8-fold quasicrystalline pattern (Figure 3.3A-C). In contrast to the homogeneous tiling patterns, these three structures contain: 1) a diverse number of arms that meet at each vertex, ranging from three to eight arms; 2) various intersection angles between any neighboring arms, ranging from 30° to 150°; and 3) variable arm lengths, from three to six turns. Successful formation of these structures demonstrates the high programmability of our methods.

Changes about the numbers of arms and intersection angles are achieved by the insertion of T_n loops in the staple strands around each vertex and leaving unpaired bases on the scaffold strand between arms as necessary (see figures S22-S24 for design details). For example, in the Penrose tiling (Figure 3.3B), surrounding the central 5-arm vertex there are five 3-arm vertices that have different angles between the three arms: 108°, 108°, and 144° (figure S23). T_3, T_3, and T_4 are inserted in the staple strand surrounding the vertex, and one unpaired nucleotide was left on the scaffold strand to achieve the 144° angle (figure S23-2). The next group of vertices connected to the central one all contain 5-arms that have angles of two 36°, two 72°, and one 108°. These angles are achieved by inserting T_5 or T_4 loops in the staple strands and a T_5 loop opposite to an unpaired nucleotide on the scaffold (figure S23-4).

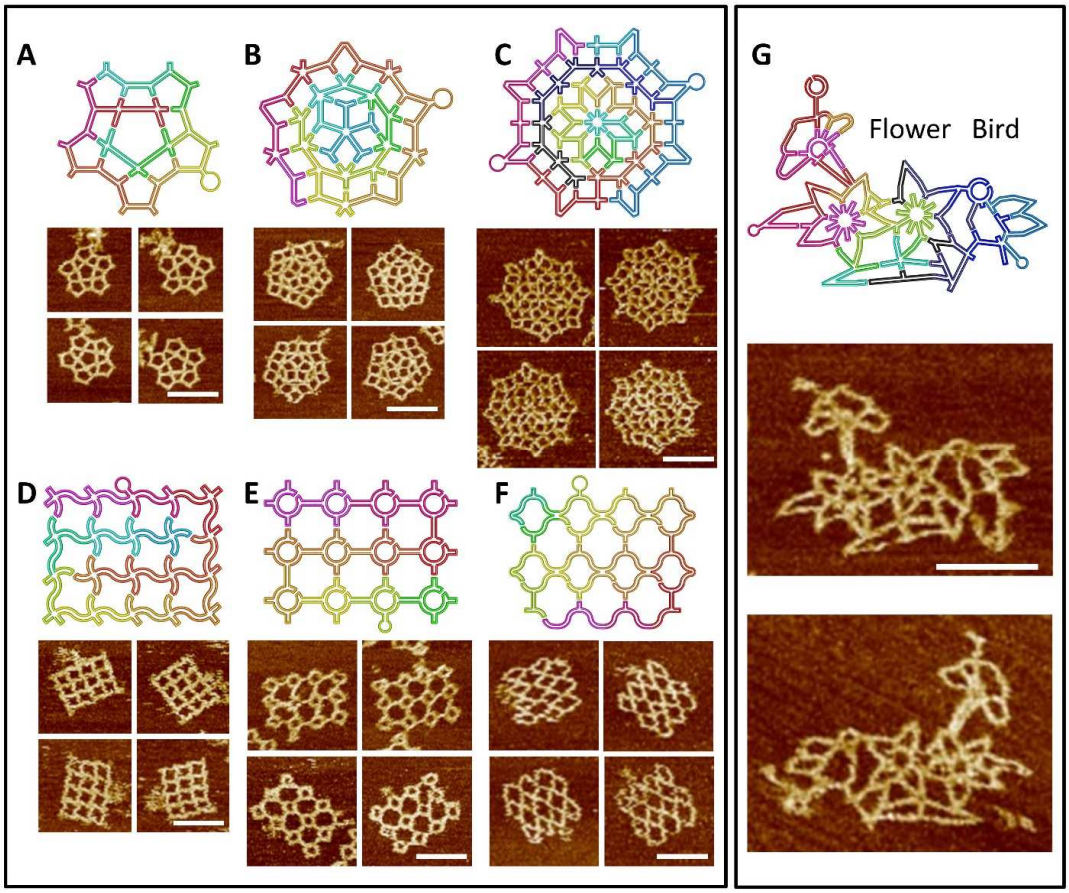


Figure 3.3 The scaffold folding path and representative AFM images for intricate 2D patterns. (A) A star-shape pattern without translational symmetry. (B) A Penrose tiling. (C) An 8-fold quasicrystalline pattern. (D-F) Three curved structures. (D) A waving grid. (E) A sphere array. (F) A fishnet. (G) A flower-and-bird pattern. All scale bars are 100 nm. Two scaffolds (one colorful and one black-blue) are used in C and G.

In another example, the quasicrystalline pattern shown in Figure 3.3C contains six different types of vertices: one symmetric 8-arm with all 45° angles; 24 3-arms with 90°, 135°, 135° angles; 16 4-arms with 45°, 90°, 135°, 90° angles; 8 5-arms with 90°, 45°, 90°, 90°, 45° angles; 8 4-arms with all 90° angles; and 8 2-arms with 45° and 315° angles (figure S24). Each of the unique vertice types requires an individualized design. Generally, T_5, T_4, and T_2 loops are used to create 45°, 90°, and 135° angles, respectively. Adding unpaired nucleotide(s) in the scaffold strand opposite to the loop on the staple

strand produces larger angles. The rules for creating different angles are summarized in the supporting information (figure S38).

For the two quasicrystalline patterns, all the edges are of the same length. However, for the star-shape pattern shown in Figure 3.3A, the edges are three different lengths that have a ratio of 6:5:2.9, which can be achieved by using 6, 5, and 3 full helical turns in the arms, respectively. Therefore, various target structures with reduced local symmetry can be successfully synthesized by adjusting the angles and arm lengths. Structural characterization by native gel electrophoresis (figure S46) and AFM imaging confirmed the successful assembly of the structures with high yield and excellent structural integrity (Figure 3.3A-C).

In our design, every line segment of the patterns can be considered an analog of a DX DNA tile (two double-helical DNA linked by two or three DXs). This feature enables us to introduce more intricate curvatures at each line segment to generate waving or circular patterns (Figure 3.3D-F), which can be achieved using a previously reported targeted insertion and deletion method[12]. Taking the waving grid structure as an example (Figure 3.3D), we laid down the scaffold strand on each line segment as two concentric arcs instead of two parallel lines. In each unit, a ~72° arc is created when a 31-bp and a 21-bp double-helical DNA are laid side-by-side and connected by two crossovers at both ends (see figures S25 and S26 for details). By rotating and attaching such segments together, waving patterns of arbitrary curvatures can be generated (figures S25-S34).

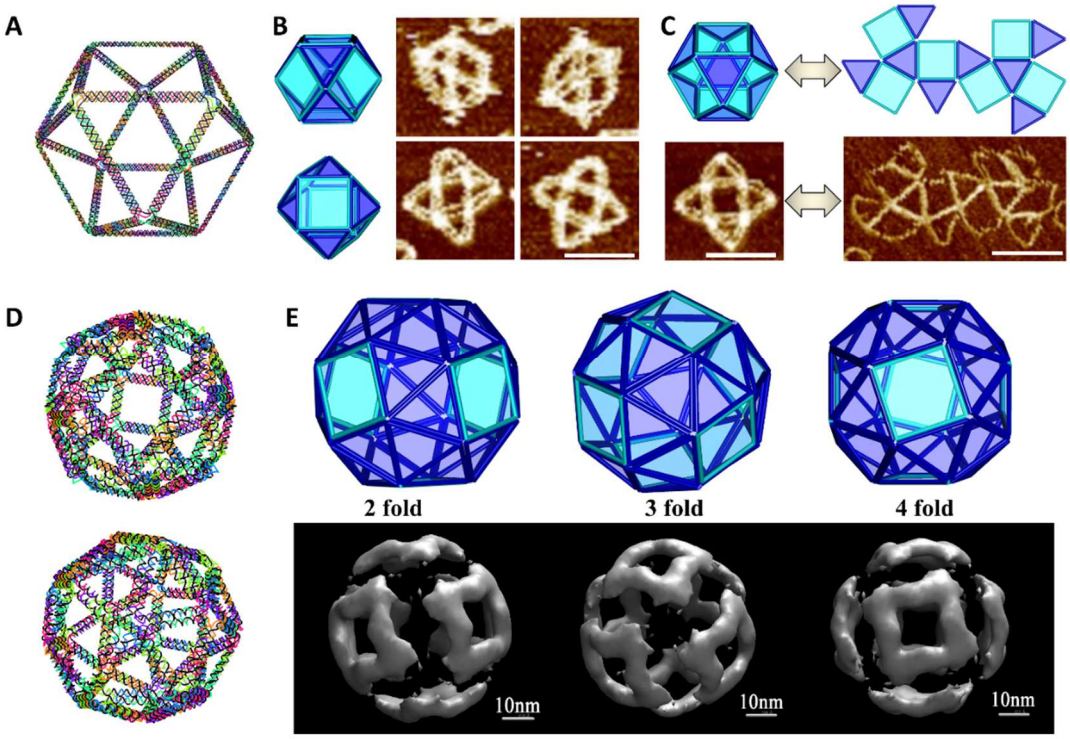


Figure 3.4 3D wire-frame Archimedean solid structures. (A) A 3D model of an Archimedean solid cuboctahedron with 12 vertices and 24 edges. Each vertex is a 4 × 4 junction, and each edge is a 14-turn long double DNA duplex. (B) Left: Models showing possible conformations of the structure when deposited on mica surface; Right: the corresponding AFM images. (C) The reconfiguration between 3D and 2D can be realized by strand displacement by adding fuel and set strands. Top: reconfiguration schematics, Bottom: AFM images showing the transition. All scale bars in AFM images are 100 nm. (D) A 3D model of another Archimedean solid snub cube with 24 vertices and 60 edges. Each vertex is 5 × 4 junction, and each edge is a 5-turn double DNA duplex. (E) Three views reflected the 2-, 3-, and 4-fold symmetry of the DNA snub cube from models (top) and reconstructions from cryo-EM images.

To test our design methods for producing patterns of extreme complexity, a more intricate and arbitrary shape in the form of three flowers and a bird was constructed through the angle variation and curvature creation as discussed above (Figure 3.3C). This structure consists of vertices that range from 2- to 10-arms, a wide range of different angles between the arms, and various arm lengths and curvatures. We folded two scaffold strand loops and individually modified each of the vertices and lines to create the desired

angles and local curvatures (see figures S35-S37 for details). AFM imaging clearly shows the correct formation of the structures resembling the designed patterns, demonstrating the versatility of our strategy to create very sophisticated wire-frame DNA nanostructures that have been difficult to achieve using previous methods.

The design principle can be easily adapted to generate 3D architectures that have vertices and edges arranged in 3D space with or without structural symmetry. This can be achieved by following the same design steps from turning single line edges into two lines, "looping" and "bridging," adjusting angles, and filling in staples. One important step for 3D construction is identifying appropriate folding paths and bridge points to fold the scaffold into a single loop that winds through all parts of the structure. A simple solution is to use the topological equivalent 2D polygon net for the 3D structures to allow the use of the "loop" and "bridge" strategy demonstrated in 2D to generate the looping path of the scaffold (figures S42 and S43). Angles in each vertex are adjusted using the same method by inserting T_n loops in the staples around the vertices and leaving unpaired nucleotides in the scaffold at the vertices (figure S41 and S44).

As shown in Figure 3.4, we designed and constructed a simple Archimedean solid cuboctahedron that contains 12 equivalent 4-arm vertices (with 60°, 90°, 60°, and 90° angles) and equal length 2-helix DNA as edges that are each 14 full DNA helical turns long. The structure can be easily observed under AFM (Figure 3.4, B and C) in a flattened form when it is deposited on the mica surface. The majority of the objects observed are centered with one of the square surfaces, indicating that the square surface has a higher tendency (stronger binding affinity) to contact the mica surface than the triangular surfaces or any vertex. Flattening of the 3D shape on the surface in this

preferred way involved only angle changes at the vertices and minimal stretching or compression of any edges, which is understandable as the angle change at the single-stranded loops around the vertices should involve much smaller free energy costs than stretching or compressing the DNA helices.

Reconfigurable transitions between the 2D and 3D conformations were achieved using the strand displacement technique[19]. Upon addition of corresponding fuel and set strands, the staples on the outer edges of the 2D net were released and replaced by new staples that joined the specific edges together to form the 3D conformation. The new staples on the 3D polyhedron were also designed with a new set of toehold strands to allow the unfolding transformation from 3D to 2D (see supporting information for details). The unfolding-refolding reconfigurations were clearly verified by AFM imaging (Figure 4.4 and figures S63-S64), providing further proof for correct 3D structure formation.

We further constructed a more complex 3D object a snub cube, which is also an Archimedean solid with 60 edges, 24 vertices and 38 faces including 6 squares and 32 equilateral triangles. Each edge were designed as 5-turn long DNA double helix. The 5-arm vertices were assigned as T_5, T_5, T_5, T_5 and T_4 loops between adjacent arms corresponding to 60°, 60°, 60°, 60°, and 90° angles. Cryogenic transmission electronic microscopy (cryo-EM) provided direct evidence for the formation of snub cube (Figure 4.4E and S65). The cryoEM study confirmed the overall geometry and all key elements shown up in the reconstructed model. The observed edges are about 20 nm long and 6 nm thickness, which matches the designed structures (22 nm long and 5 nm thickness). The 2-, 3-, and 4-fold symmetry in the snub cube were proofed by cryo-EM (Figure 4.4E).

3.5 Conclusion

In summary, we have demonstrated the utility of a set of new design rules for engineering wire-frame DNA nanostructures with unprecedented complexity. In principle, this strategy can be applied to design and construct any imaginable wire-frame nanostructure. In the examples demonstrated here, we utilized antiparallel double-crossover junctions to form "bridges" between two neighboring helices. Other DNA or RNA junction motifs such as paranemic crossovers (PX)[20] and RNA kissing loops[21] can also be exploited to enrich the design and produce interesting topological structures that could potentially be replicated in vivo through biological machineries. The structural platform reported here enables the construction of nanostructures with a new level of structural complexity. The fabrication of quasicrystalline materials with rational design and programmability has been a challenging task. The method described here offers a new way to program the formation of quasicrystalline patterns for templated assembly of functional nanomaterials with emerging properties.

3.6 References

(1) Aldaye, F. A.; Palmer, A. L.; Sleiman, H. F., *Science* **2008,** 321, 1795-1799.

(2) Seeman, N. C., *Annu Rev Biochem* **2010,** 79, 65-87.

(3) Pinheiro, A. V.; Han, D. R.; Shih, W. M.; Yan, H., *Nat Nanotechnol* **2011,** 6, 763-772.

(4) Zhang, F.; Nangreave, J.; Liu, Y.; Yan, H., *J Am Chem Soc* **2014,** 136, 11198-11211.

(5) Winfree, E.; Liu, F. R.; Wenzler, L. A.; Seeman, N. C., *Nature* **1998,** 394, 539-544.

(6) Rothemund, P. W. K., *Nature* **2006,** 440, 297-302.

(7) He, Y.; Ye, T.; Su, M.; Zhang, C.; Ribbe, A. E.; Jiang, W.; Mao, C. D., *Nature* **2008,** 452, 198-U41.

(8) Douglas, S. M.; Dietz, H.; Liedl, T.; Hogberg, B.; Graf, F.; Shih, W. M., *Nature* **2009,** 459, 414-418.

(9) Wei, B.; Dai, M. J.; Yin, P., *Nature* **2012,** 485, 623.

(10) Ke, Y. G.; Ong, L. L.; Shih, W. M.; Yin, P., *Science* **2012,** 338, 1177-1183.

(11) Andersen, E. S.; Dong, M.; Nielsen, M. M.; Jahn, K.; Subramani, R.; Mamdouh, W.; Golas, M. M.; Sander, B.; Stark, H.; Oliveira, C. L. P.; Pedersen, J. S.; Birkedal, V.; Besenbacher, F.; Gothelf, K. V.; Kjems, J., *Nature* **2009,** 459, 73-U75.

(12) Dietz, H.; Douglas, S. M.; Shih, W. M., *Science* **2009,** 325, 725-730.

(13) Han, D. R.; Pal, S.; Nangreave, J.; Deng, Z. T.; Liu, Y.; Yan, H., *Science* **2011,** 332, 342-346.

(14) Han, D. R.; Pal, S.; Yang, Y.; Jiang, S. X.; Nangreave, J.; Liu, Y.; Yan, H., *Science* **2013,** 339, 1412-1415.

(15) He, Y.; Chen, Y.; Liu, H. P.; Ribbe, A. E.; Mao, C. D., *J Am Chem Soc* **2005,** 127, 12202-12203.

(16) Yan, H.; Park, S. H.; Finkelstein, G.; Reif, J. H.; LaBean, T. H., *Science* **2003,** 301, 1882-1884.

(17) He, Y.; Tian, Y.; Ribbe, A. E.; Mao, C. D., *J Am Chem Soc* **2006,** 128, 15978-15979.

(18) Liu, Y.; Ke, Y. G.; Yan, H., *J Am Chem Soc* **2005,** 127, 17140-17141.

(19) Yurke, B.; Turberfield, A. J.; Mills, A. P.; Simmel, F. C.; Neumann, J. L., *Nature* **2000,** 406, 605-608.

(20) Shen, Z. Y.; Yan, H.; Wang, T.; Seeman, N. C., *J Am Chem Soc* **2004,** 126, 1666-1674.

(21) Geary, C.; Rothemund, P. W. K.; Andersen, E. S., *Science* **2014,** 345, 799-804.

CHAPTER 4

RECONFIGURABLE DNA ORIGAMI TO GENERATE QUASI-FRACTAL PATTERNS

4.1 Abstract

The specificity of Watson-Crick base pairing, unique mechanical properties of DNA, and intrinsic stability of DNA double helices make DNA an ideal material for the construction of dynamic nanodevices. Rationally designed strand displacement reactions can be used to produce dynamic reconfiguration of DNA nanostructures post-assembly. Here we describe a 'fold-release-fold' strategy of multiple strand displacement and hybridization reactions to reconfigure a simple DNA origami structure into a complex, quasi-fractal pattern, demonstrating a complex transformation of DNA nanoarchitectures.

4.2 Introduction

DNA nanotechnology exploits the predictable structural and molecular recognition properties of DNA to build nanoscale assemblies and devices[1-12]. Dynamic DNA nanotechnology seeks to develop reconfigurable and autonomous DNA devices that respond to environmental cues and system inputs[13-23]. These dynamic elements are particularly important for applications in nanomedicine, nanorobotics, and DNA based information processing.

One of the most frequently used operational principles in DNA device construction is based on DNA strand displacement, a simple and robust process by which two DNA strands hybridize to each other, displacing one or more pre-hybridized strands in the process[24-27]. DNA strand displacement can be combined with structural self-assembly to produce dynamic reconfiguration of large DNA nanostructures post-assembly, and can be used to induce changes at the macroscopic scale[28]. There have been reports of the construction of rotary DNA devices that switch between different structural states[24, 29], two-dimensional crystals that contain DNA switches for controlled motion relative to the crystal lattice[30], two-dimensional frameworks that change aspect ratio with a specific input[31], and tetrahedra in which one edge adopts different lengths depending on the presence of an effector strand[32]. More recently, DNA origami technology[5] was used to construct a reconfigurable DNA box with a lid[33] that can be opened and closed by DNA strand displacement and a Möbius strip[34] that can be reconfigured into a longer strip or two interlocked rings by releasing the staple strands at certain positions. In each of these cases, reconfiguration occurs between two closely-related structures through the action of a small number of strand displacement events. We envision that many applications would benefit from the ability to realize more complex changes in state.

Here, we present a 'fold-release-fold' method in which reconfiguration of a DNA origami frame structure into a complex, quasi-fractal pattern is achieved by multiple strand displacement and hybridization steps. Unlike previous reports, our method does not involve the use of algorithmic self-assembly to generate the fractal pattern[11,35] We demonstrate that reversible reconfiguration between the structurally divergent DNA nanostructures is facilitated by priming the dynamic portions of the nanostructure. We

also show that the use of a large number of simultaneous toehold mediated strand displacement events does not hinder the reconfiguration process. Fractal-like patterns have also been produced.

4.3 Materials and Methods

See Appendix C.

4.4 Results and Discussion

The initial square frame structure (Frame 1) whose design is presented in the left panel of Figure 4.1a is assembled by folding the genome of an M13mp18 bacteriophage (~7000 nucleotides) with 156 short, single stranded staple strands ranging from 20 to 70 nucleotides (nt) long. Each of the four arms of Frame 1 is composed of 8 double helical layers, with the innermost layer denoted as layer n. Each subsequent helical layer, n+1, n+2, etc. has 16 more base pairs than the preceding layer to tolerate 90 degree bends in the helices at each corner. Of the 156 staple strands, 42 actively participate in the transformation from Frame 1 to Frame 2 (Primer Set 1, shown in blue in the left panel of Figure 4.1b). Frame 2 (middle panel of Figure 4.1a) has the same basic outline as Frame 1, but rather than one large eight layer frame, it contains four smaller, four layer frames, with two that exhibit the same symmetry as Frame 1. Because the transformation involves significant rearrangement of the four innermost helices, the corresponding dynamic portions of the nanostructure are primed before the final transformation (upper right and lower left corners of the four innermost helices).

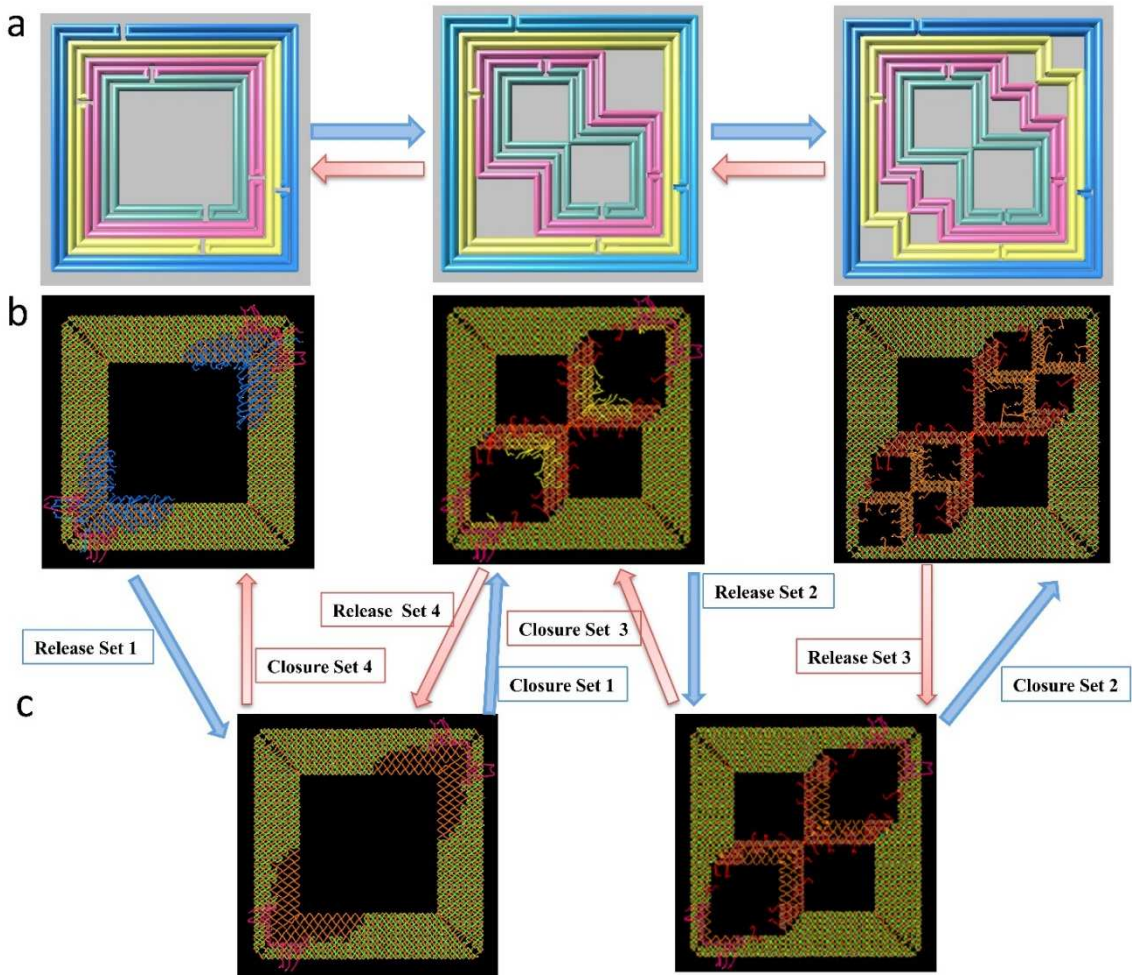


Figure 4.1 Design of reconfigurable DNA origami. (a) Left to right - models of Frame 1, Frame 2, and Frame 3. In each model, the cylinders represent DNA double helices. Several iterations of strand displacement and hybridization reactions transform the simple frame structure into a quasi-fractal pattern with self-similar features. Frame 1 has ~ 100 nm x 100 nm outer dimensions and a 50 nm x 50 nm inner cavity. In Frame 2, the four innermost layers in the upper right and lower left of Frame 1 are folded toward the center and linked together. This creates a Frame structure with 4 smaller square cavities that are ~ 25 nm x 25 nm. The upper right and lower left cavities have the same polarity as Frame 1 and are dynamically reconfigured to create Frame 3. (b) Schematics showing the detailed design features of each of the DNA origami structures. The orange strand represents the long single-stranded M13 DNA scaffold. The green strands correspond to non-transformative staples that fold the scaffold during the assembly of Frame 1, forming a rigid DNA origami structure. The remaining colored strands (blue, pink, yellow) contain toehold extensions and represent dynamic portions of the design. (c) Schematics of the partially relaxed intermediate structures that are formed during each transformation. Note that the forward transformation from Frame 1 to Frame 3 and the reverse transformation from Frame 3 to Frame 1 are both depicted in this Figure.

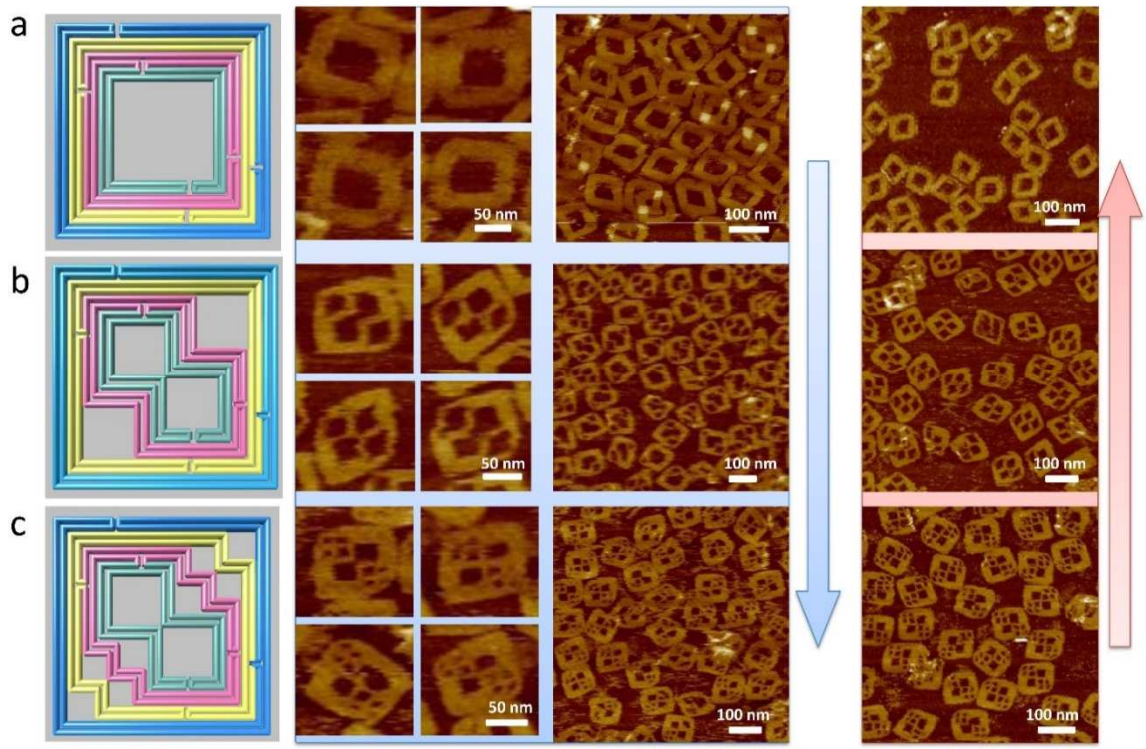


Figure 4.2 Iterations of structural reconfiguration. Left panels illustrate the designs, middle panels (in blue) contain zoom in and zoom out AFM images corresponding to the forward transformation from Frame 1 to Frame 3, and right panels (in red) show the reverse transformation from Frame 3 to Frame 2 to Frame 1. (a) Frame 1 in blue, assembled by a typical DNA origami annealing program using a long scaffold and a collection of complementary staples. Frame 1 in red, transformed from Frame 2. (b) Frame 2 in blue, formed by reconfiguration of Frame 1. Frame 2 in red, formed by reconfiguration of Frame 3. (c) Frame 3 in blue, transformed from Frame 2. Frame 3 in red, assembled by a typical DNA origami annealing program using a long scaffold and a collection of complementary staples.

Each of the 42 staples in Primer Set 1 contains two domains; one is complementary to the underlying M13 scaffold and facilitates the initial assembly of Frame 1. The second domain contains of a 6 nt toehold sequence that is used for priming the structure for reconfiguration to Frame 2 (Figure S1). Initially, Frame 1 is assembled by thermal annealing (left panel in Figure 4.1b, details in SI); primer staples are incorporated into the frame with the 6 nt toeholds projecting from the same face of the structure. The toeholds serve as the initial points of strand displacement by fully

complementary strands (Release Set 1, 42 strands). Removing the primer staples with the addition of Release Set 1 loosens the structure in the upper right and lower left corners and prepares Frame 1 for the final transformation to Frame 2 (left panel in Figure 4.1c, Figure S2). The last step involves bringing the exposed portions of M13 in the upper right and lower left corners of the structure toward the center of the frame and fixing them in place by adding Closure Set 1 staples (shown in red and yellow in the middle panel of Figure 4.1b). Closure Set 1 (38 staples in total) includes two long staple strands, 69 and 71 nts, that hold the center of Frame 2 together (Figure S3). The remainder of the staples in Closure Set 1 hybridizes to the unbound sections of M13, transforming the structure into Frame 2 (middle panel in Figure 4.1b). Selected staples in Closure Set 1 are also used in next iteration of forward transformation (yellow) and others are used for the reverse transformation from Frame 2 to Frame 1 (red).

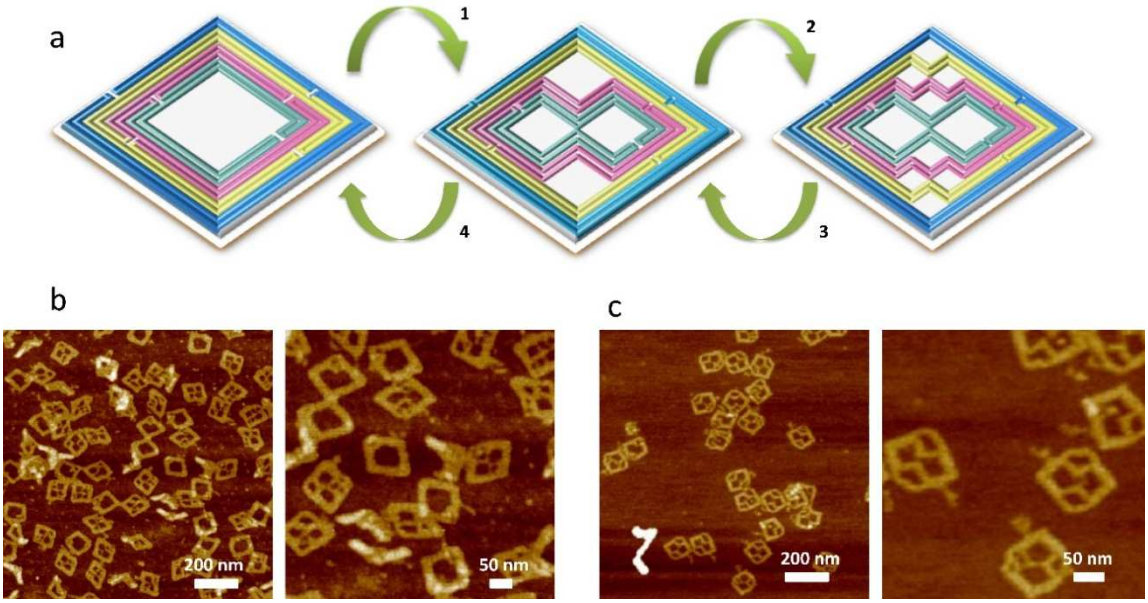


Figure 4.3 Continuous forward and reverse reconfiguration. (a) Models illustrating the fully reversible reconfiguration process. A number is assigned to each transformation step (shown next to the green arrows). (b) Zoom out and zoom in AFM images, respectively,

of the products that were obtained after transformation #4. Approximately half the observed products are Frame 1, the intended target structure. c, Zoom out and zoom in AFM images, respectively, of the products that were obtained after transformation #3. Note the small percentage of Frame 1 visible in the images, indicating that the appearance of Frame 1 after transformation #4 can mostly be attributed to the intended reconfiguration of Frame 2, and not simply structures leftover from earlier steps.

The reconfiguration of Frame 2 into Frame 3 (right panel of Figure 4.1a) proceeds in the same manner as described above. 26 staples presenting 6 nt toeholds are used to prime Frame 2 for transformation (Primer Set 2). 10 of the primer staples are incorporated into Frame 1 during the initial assembly step (shown in pink in the left panel of Figure 4.1b) and are not affected by the transformation from Frame 1 to Frame 2. The remaining 16 primer staples (shown in yellow in the middle panel of Figure 4.1b) are incorporated into Frame 2 through the addition of Closure Set 1. Removing the primer staples by adding strands from Release Set 2 prepares the dynamic portions of the structure for the final reconfiguration (right panel in Figure 4.1c). The addition of Closure Set 2 (32 staples) brings the corresponding exposed portions of M13 together and fixes them in place, transforming the structure into Frame 3 (right panel in Figure 4.1b). Ideally, if an infinitely long scaffold strand were used, this iterative process could be repeated to form more complex fractal patterns with smaller and smaller cavities. However, limited by the length of M13 and rigidity of double stranded DNA, no additional iterations were conducted. The same reconfiguration strategy can be used to transform a complex structure (Frame 3) into a simple structure (Frame 1). The reverse transformation is also depicted in Figure 4.1 (pink arrows).

Additional details about the reconfiguration process are described in the SI. Figures S4-S7 corresponds to a series of experiments that were used to identify suitable

temperatures for each reconfiguration step. In summary, we found that 45°C and 37°C were acceptable temperatures for the transformations from Frame 1 to Frame 2 and Frame 2 to Frame 3, respectively. For the former, temperatures higher than 45°C compromised the stability of the pre-annealed Frame 1 structure, while lower temperatures did not facilitate efficient transformation. Similar results were observed for the latter transformation.

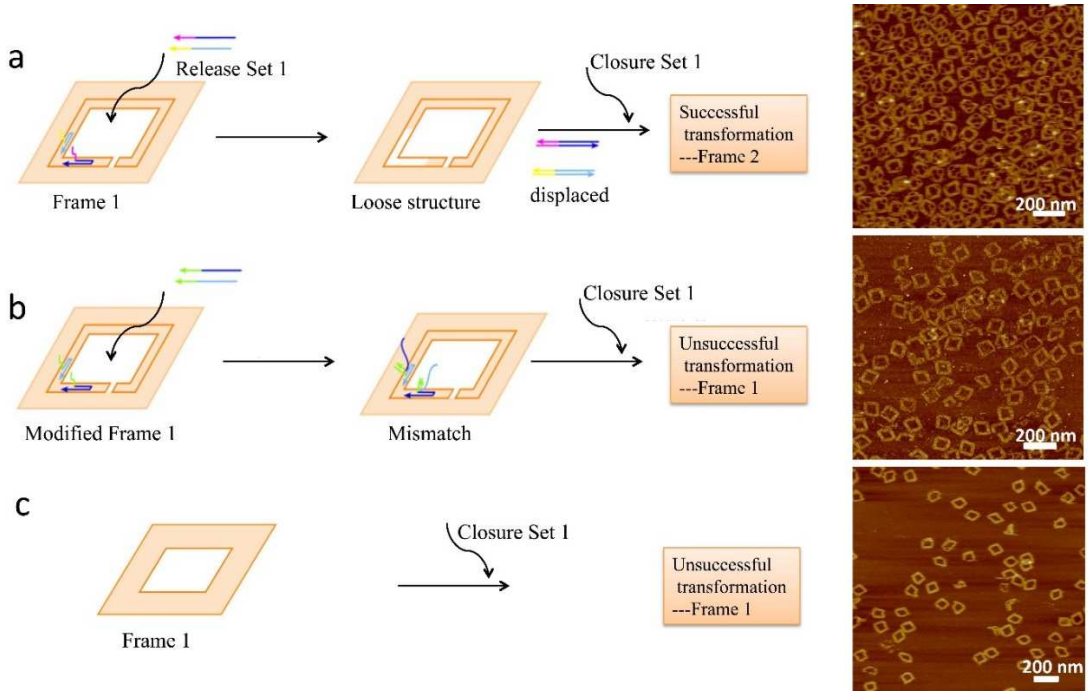


Figure 4.4 Priming the structures for reconfiguration. (a) Schematic and corresponding AFM image (products) illustrating a standard transformation from Frame 1 to Frame 2. As shown in the AFM image, the transformation yield is approximately 90%. (b) Schematic and corresponding AFM image (products) demonstrating the effect of blocking toehold mediated strand displacement. For this experiment, Frame 1 was modified with primer staples that display a non-complementary sequence not recognized by the displacement strands. To realize the transformation, Modified Frame 1 was treated with Release Set 1, followed by Closure Set 1. Notice the absence of Frame 2 in the AFM image, indicating that toehold recognition is essential to facilitate the reconfiguration process. c, Schematic and corresponding AFM image (products) illustrating the result of skipping the release step of the ‘fold-release-fold’ method. For this experiment, Closure Set 1 was added directly to Frame 1 without first adding Release Set 1. The

absence of Frame 2 in the corresponding AFM image confirms that priming the structures is an essential aspect of our approach to nanostructure reconfiguration.

Atomic Force Microscopy (AFM) was used to characterize each step of the forward and reverse reconfigurations (Figure 4.2). The far left column in Figure 4.2 depicts each DNA origami frame structure. Zoom in AFM images directly adjacent to each model confirm the integrity of the structures at each step of the forward reconfiguration. Zoom out AFM images of the forward (blue series) and reverse (red series) transformations reveals that the yield of each reconfiguration step is approximately 85%. Note that for the reverse transformation, Frame 3 is initially assembled by thermal annealing and transformed into Frame 1 at a constant temperature using the methods described above.

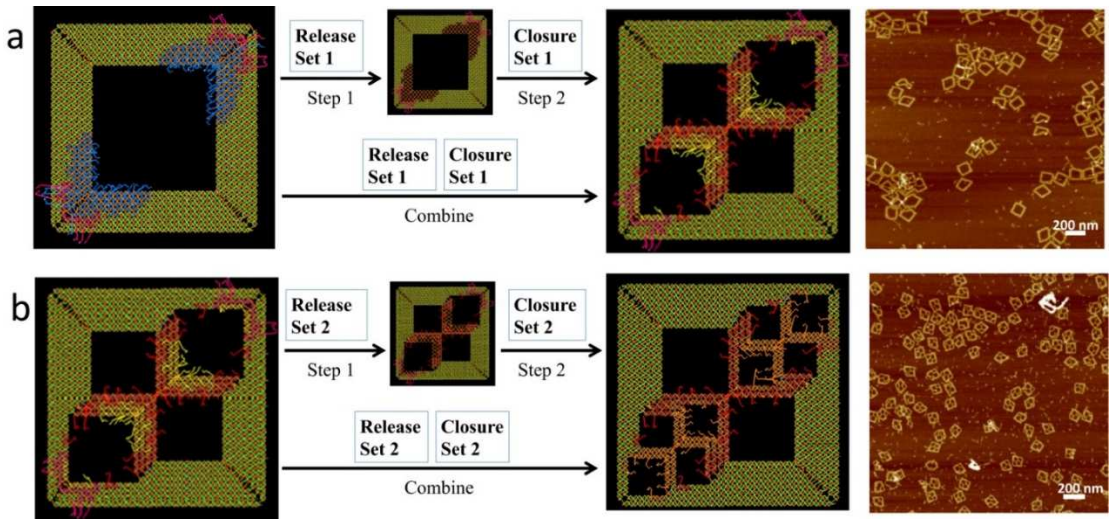


Figure 4.5 A single step reconfiguration strategy. (a) Schematic and AFM image of the products of a one-step transformation from Frame 1 to Frame 2. (b) Schematic and AFM image of the products of a one-step transformation from Frame 1 to Frame 2. As shown by the AFM images, a one-step protocol does not facilitate the reconfiguration process.

With confirmation that our fold-release-fold strategy can not only be used to transform a simple structure into a more complex one, but also a complex structure into a simple one, we sought to demonstrate a truly reversible reconfiguration. Beginning with

the thermal annealing of Frame 1, we used our step-wise method to execute continuous forward and reverse transformations (Frame 1→Frame 2→Frame 3→Frame 2→Frame1). Although we were able to show that a fully reversible reconfiguration is possible using our method, the AFM images shown in Figure 4.3 reveal that the overall yield is significantly lower than for each individual transformation (~50% as evidenced by the appearance of Frame 1 upon completion of step 4 of the transformation).

We designed several control experiments to evaluate the importance of toehold recognition and preparing the structures for reconfiguration. First, Modified Frame 1 was assembled (Figure 4.4b); the primer staples incorporated into the modified structure displayed toeholds that were not complementary to the strands in Release Set 1, and thus, could not be recognized and displaced. As a result, Modified Frame 1 was not transformed into Frame 2 after the sequential addition of Release Set 1 and Closure Set 1 (AFM image in Figure 4.4b). This result indicates that the free energy provided by toehold hybridization is necessary to initiate the primer strand displacement process and subsequent rearrangement of the structure. In a related experiment, we directly examined the need for a 'release' step (fold-release-fold) in which the structures are primed for transformation (Figure 4.4c). After assembling Frame 1, the sample was not prepared for subsequent reconfiguration into Frame 2 (Release Set 1 was not added to displace dynamic portions of the structure) but was directly treated with Closure Set 1. As shown in the AFM image in Figure 4.4c, there is no evidence that Frame 1 was successfully transformed into Frame 2. Taken together, the control experiments indicate that priming structures for reconfiguration is a critical component of our method.

We also performed control experiments to determine the need for separate 'release' and 'fold' steps. First, Frame 1 was assembled by thermal annealing. Next, the sample was simultaneously treated with Release Set 1 and Closure Set 1 in an attempt to transform the structure into Frame 2 in a single step. However, the absence of Frame 2 in the AFM images shown in Figure 4.5a reveals that the one-step displacement strategy is not effective method to transform the simple structure into a more complex pattern. A similar result was obtained for the transformation from Frame 2 to Frame 3 (Figure 4.5b). These results indicate that forming a relaxed, intermediate structure is a prerequisite for structural reconfiguration by the closure staples.

4.5 Conclusion

In summary, we demonstrated that a DNA origami nanostructure can undergo complex structural rearrangement using a 'fold-release-fold' strategy. In this work a simple DNA origami frame structure was transformed into a complex, quasi-fractal pattern and vice-versa. The initial structures were self-assembled by thermal annealing and subsequently transformed through multiple strand displacement and hybridization steps. We showed that unfastening the staples in the dynamic portions of the nanostructure before reorganizing the corresponding sections of M13 facilitated the transformation process. In addition, we demonstrated that transformation between the structurally divergent DNA objects is fully reversible. The success of our method proves that a large number of simultaneous toehold mediated strand displacement events can be used to achieve efficient structural reconfiguration. The reconfigurable quasi-fractal

structures may also be used to control metal nanoparticles to form reconfigurable self-similar chains for nanophotonic applications.

4.6 References

(1) Seeman, N. C., *J Theor Biol* **1982,** 99, 237-247.

(2) Kallenbach, N. R.; Ma, R. I.; Seeman, N. C., *Nature* **1983,** 305, 829-831.

(3) Seeman, N. C., *Annu Rev Biochem* **2010,** 79, 65-87.

(4) Mao, C. D., *Nat Nanotechnol* **2008,** 3, 75-76.

(5) Rothemund, P. W. K., *Nature* **2006,** 440, 297-302.

(6) Aldaye, F. A.; Palmer, A. L.; Sleiman, H. F., *Science* **2008,** 321, 1795-1799.

(7) Kuzuya, A.; Komiyama, M., *Nanoscale* **2010,** 2, 310-322.

(8) Shih, W. M.; Lin, C. X., *Curr Opin Struc Biol* **2010,** 20, 276-282.

(9) Torring, T.; Voigt, N. V.; Nangreave, J.; Yan, H.; Gothelf, K. V., *Chemical Society Reviews* **2011,** 40, 5636-5646.

(10) Chen, J. H.; Seeman, N. C., *Nature* **1991,** 350, 631-633.

(11) Rothemund, P. W. K.; Papadakis, N.; Winfree, E., *Plos Biol* **2004,** 2, 2041-2053.

(12) Douglas, S. M.; Dietz, H.; Liedl, T.; Hogberg, B.; Graf, F.; Shih, W. M., *Nature* **2009,** 459, 1154-1154.

(13) Gu, H.; Chao, J.; Xiao, S.-J.; Seeman, N. C., *Nat Nano* **2009,** 4, 245-248.

(14) Gu, H.; Chao, J.; Xiao, S.-J.; Seeman, N. C., *Nature* **2010,** 465, 202-205.

(15) Mao, C. D.; Sun, W. Q.; Shen, Z. Y.; Seeman, N. C., *Nature* **1999,** 397, 144-146.

(16) Maye, M. M.; Kumara, M. T.; Nykypanchuk, D.; Sherman, W. B.; Gang, O., *Nat Nanotechnol* **2010,** 5, 116-120.

(17) Gu, H. Z.; Yang, W.; Seeman, N. C., *Journal of the American Chemical Society* **2010,** 132, 4352-4357.

(18) Li, J. W. J.; Tan, W. H., *Nano Lett* **2002,** 2, 315-318.

(19) Shen, W. Q.; Bruist, M. F.; Goodman, S. D.; Seeman, N. C., *Angew Chem Int Edit* **2004,** 43, 4750-4752.

(20) Sherman, W. B.; Seeman, N. C., *Nano Lett* **2004,** 4, 1801-1801.

(21) Yan, H.; Zhang, X. P.; Shen, Z. Y.; Seeman, N. C., *Nature* **2002,** 415, 62-65.

(22) Seeman, N. C., *Trends Biochem Sci* **2005,** 30, 119-125.

(23) Lubrich, D.; Lin, J.; Yan, J., *Angew Chem Int Edit* **2008,** 47, 7026-7028.

(24) Yurke, B.; Turberfield, A. J.; Mills, A. P.; Simmel, F. C.; Neumann, J. L., *Nature* **2000,** 406, 605-608.

(25) Zhang, D. Y.; Winfree, E., *Journal of the American Chemical Society* **2009,** 131, 17303-17314.

(26) Li, Q. Q.; Luan, G. Y.; Guo, Q. P.; Liang, J. X., *Nucleic Acids Res* **2002,** 30.

(27) Simmel, F. C.; Yurke, B., *Phys Rev E* **2001,** 63.

(28) Zhang, D. Y.; Seelig, G., *Nat Chem* **2011,** 3, 103-113.

(29) Yan, H.; Chhabra, R.; Sharma, J.; Liu, Y., *Nano Lett* **2006,** 6, 978-983.

(30) Ding, B.; Seeman, N. C., *Science* **2006,** 314, 1583-1585.

(31) Feng, L. P.; Park, S. H.; Reif, J. H.; Yan, H., *Angew Chem Int Edit* **2003,** 42, 4342-4346.

(32) Goodman, R. P.; Heilemann, M.; Doose, S.; Erben, C. M.; Kapanidis, A. N.; Turberfield, A. J., *Nat Nanotechnol* **2008,** 3, 93-96.

(33) Gothelf, K. V.; Andersen, E. S.; Dong, M.; Nielsen, M. M.; Jahn, K.; Subramani, R.; Mamdouh, W.; Golas, M. M.; Sander, B.; Stark, H.; Oliveira, C. L. P.; Pedersen, J. S.; Birkedal, V.; Besenbacher, F.; Kjems, J., *Nature* **2009,** 459, 73-U75.

(34) Han, D. R.; Pal, S.; Liu, Y.; Yan, H., *Nat Nanotechnol* **2010,** 5, 712-717.

(35) Patitz, M. J.; Summers, S. M., *Nat Comput* **2010,** 9, 135-172.

CHAPTER 5

PERSPECTIVES OF STRUCTURAL DNA NANOTECHNOLOGY

5.1 Frontiers of Structural DNA Nanotechnology

The interdisciplinary nature of DNA nanotechnology crosses the traditional boundaries of physics, chemistry, biology, and engineering, and allows scientists to connect and integrate their unique perspectives in pursuit of solutions to the most pressing problems in medicine, technology, and more. From the earliest DNA junction motifs, to the most recently developed DNA nanostructures of incredible complexity, the field has started to explore various novel applications including directed material assembly, structural biology, biocatalysis, DNA computing, nano-robotics, disease diagnosis, and drug delivery, as we have mentioned briefly in the previous section. Each of these applications is made possible by the ability of DNA nanostructures to direct molecular species with nanoscale precision while maintaining the utmost structural integrity. DNA nanotechnology is progressing with such incredible speed that it is becoming more and more difficult to predict from which areas the next breakthroughs will occur. Next, we are merely providing our opinion about the critical challenges that the field faces, and which directions we believe researchers should pursue to help DNA nanotechnology reach its full potential.

We have divided the remaining outlook into three main areas: 1) Design and Assembly, which will include discussions of dynamic, developmental, quasi-crystal lattice, 3D periodic crystal lattice, scaffolded-, surface mediated-, algorithmic, and topological assembly; 2) Future Applications, which will include discussions of structural DNA nanotechnology for molecular scaffolds, sensors, robotics, and computing; and 3) Beyond Structural DNA Nanotechnology, in which we conceive of potential directions the field might explore over the longer term.

5.1.1 Dynamic assembly

George Whitesides once wrote, "Although much of current understanding of self-assembly comes from the examination of static systems, the greatest challenges, and opportunities, lie in studying dynamic systems. Perhaps the most important justification for studying self-assembly is its central role in life."[1] Dynamic self-assembly processes underlie many forms of adaptive and intelligent behaviors in natural systems, however, very little are known about the principles that govern them. One of the most intriguing, dynamic self-assembling processes in living cells is the polymerization of cytoskeletal biopolymers such as microtubules. Microtubule polymerization is characterized by two very unique phenomena referred to as tread-milling[2] and dynamic instability[3]. Tread-milling is said to occur with the net addition of tubulin monomers at one end of the microtubule, and simultaneous net loss of tubulin at the opposite end. Dynamic instability is characterized by switching between phases of relatively slow and rapid shortening of the microtubules at their ends. Although these phenomena were once thought to be incompatible, it is now known that both behaviors coexist in near steady state conditions in cells[2-3].

It would be quite interesting if we could use the desirable properties of DNA nanostructures to recapitulate these phenomena and ultimately dissect the governing dynamics of microtubule polymerization (Figure 5.1A). DNA tiles could be designed such that the rate of assembly equaled the rate of disassembly, resulting in steady-state tread-milling and fixed length nanotubes. Further, if tiles with two or three directional growth were utilized, the resulting arrays would have defined shapes. The intrinsic conformational flexibility and rigidity of different DNA building blocks could be exploited to mimic dynamic instability, where polymerization of flexible DNA tiles can be induced through seeded growth on a rigid tile and de-polymerization of the flexible tiles can be initiated by removing the rigid tile protection cap. When the association and dissociation reactions reach equilibrium, the input of additional rigid tiles will catalyze the polymerization of released flexible tiles. Studying the association and dissociation kinetics of model DNA tile species with variable flexibility is absolutely essential to recreating this, or similar dynamic self-assembling systems.

5.1.2 Developmental assembly

The creation of new life depends on a set of extraordinary developmental processes including stem cell growth, differentiation, and morphogenesis. These processes rely on nature's ability to precisely control the spatial and temporal relationship between cellular components and signaling pathways. It would be extremely interesting if we could create synthetic DNA systems that mimic this kind of spatiotemporal development. DNA tiles have the potential to develop into unique patterns through instructions embedded in the building blocks, or by external stimuli such as fuel strands that trigger new growth pathway (Figure 5.1B). Meta-stable DNA nanostructures could

be designed and used to serve as nucleation seeds and/or catalysts to increase the growth and development of particular pathways. Multivalency and/or cooperativity within DNA nanostructures could be exploited for nucleation and initiation of alternative assembly paths.

Researchers have already begun implementing certain aspects of developmental assembly. For example, Pierce and co-workers recently reported the dynamic assembly of DNA nanostructures through a seeded cascade of hybridization chain reactions based on toehold mediated strand displacement[4]. Strand displacement circuits have also been used to trigger DNA tile assembly and control their growth into DNA tubes[5]. There are several key challenges to implementing toehold mediated strand displacement in dynamic DNA systems including leakage, slow reaction rates, and the necessity for high salt conditions. Researchers are currently trying to address each of these problems. Zhang and co-workers reportedly designed toehold exchange probes and optimized the specificity of DNA hybridization, so that their system can detect single-based changes[6]. Designing robust self-assembling DNA platforms to mimic developmental systems will also certainly require a thorough understanding of the thermodynamics and kinetics of DNA self-assembly.

5.1.3 Quasi-crystal lattice assembly

In 2011, the Nobel Prize in Chemistry was awarded to Dan Shechtman for his discovery of quasicrystals, a finding that fundamentally changed how chemists understand solid matter. Prior to his report[7], scientists believed that the atoms in a crystal were always packed into symmetric patterns that repeated periodically. We have since come to understand that it is possible to form packed crystals from non-repeating patterns, an

arrangement of molecules now referred to as quasicrystalline. The distinctive properties of quasicrystals, as well as their unique structures, have intrigued scientists ever since their discovery[7-9], however, very little is currently known about the properties exhibited by synthetic and naturally occurring quasicrystals. Scientists have yet to determine what guides quasiperiodical rather than periodical growth, and what factors result in the unique properties that they display[10-11]. One of the biggest challenges facing researchers today is the lack of plausible systems from which to assemble quasicrystals and enable further studies. DNA platforms are promising candidates for the controlled, programmable growth of synthetic quasicrystals (Figure 5.1C). Interacting DNA building blocks can potentially be programmed to assemble into 2D and 3D quasi-crystal patterns, allowing us to investigate the still unknown mechanisms of quasi-crystal growth, and providing a means to organize other materials for engineering pursuits.

5.1.4 Periodic 3D crystal lattice assembly

Realizing 3D DNA lattices as hosts to organize guest protein molecules and facilitate protein crystallography necessitates that 3D DNA crystals can themselves be reliably assembled and characterized. Researchers successfully demonstrated the assembly of a 3D DNA crystal in which the triangular unit tiles were connected by sticky ends, and solved its structure to ~ 4 Å resolution using X-ray crystallography[12]. However, most DNA crystals only diffract to 7 to 10 Å, leaving scientists trying to determine why rationally designed DNA crystals do not diffract with better resolution. There are several possible explanations, including defects that arise during crystallization, impurities in the synthetic DNA, and the presence of bulky solvent molecules in the large cavities of the DNA lattices.

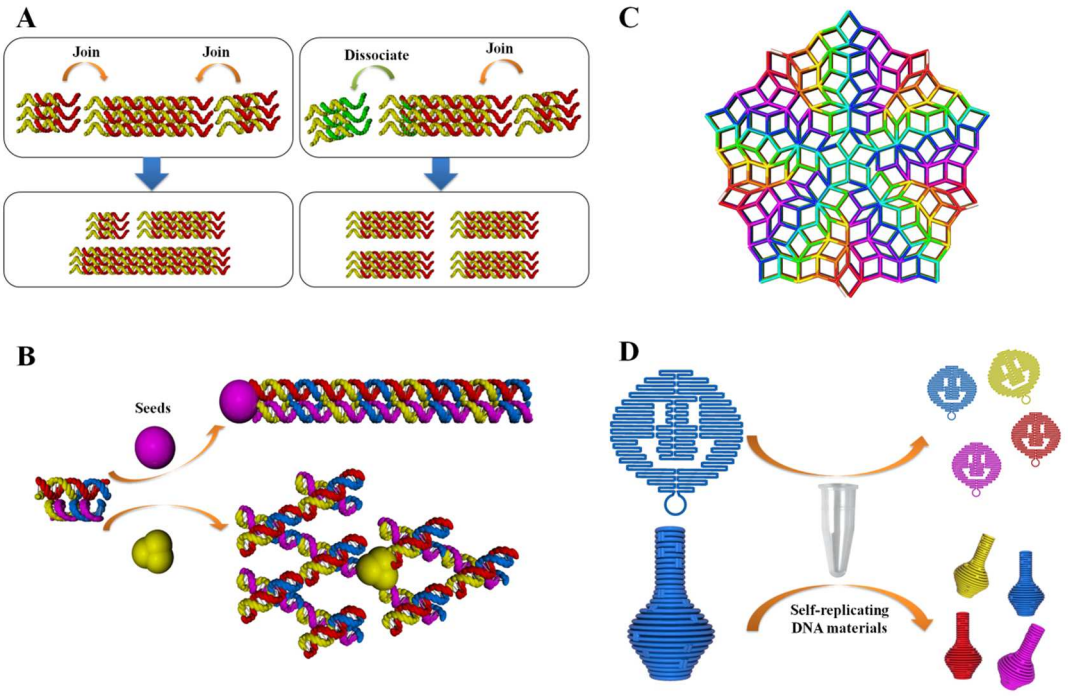


Figure 5.1 Schematic Illustrations of (A) Dynamic DNA self-assembly with simultaneous joining in one end and dissociation in the other end, (B) Developmental DNA self-assembly that the assembly process may response to different external cues to grow into different final products, (C) Example of possible Quasi-crystal (2D penrose tiling) using DNA tile self-assembly and (D) Self-replication of DNA nanostructures.

Crystal defects may be caused by the limited rigidity of DNA unit motifs, where any over- or under-twisting of the tiles causes inter-tile mismatches that are detrimental to the integrity of the crystal lattice. We surmise that imparting flexibility to certain domains of the DNA building blocks may allow the unit tiles to more reliably accommodate their neighbors and reach a lower energy state for crystal lattice formation, thereby improving the overall quality of the crystal. The Sleiman group pioneered DNA junctions with metal complex modifications that combine rigidity within the core of the junction with intrinsic flexibility in the arms[13]. This type of modified DNA unit motif has

the potential to improve the quality of DNA crystals, but has yet to be exploited for crystallization applications.

Reducing the volume of solvent present in the lattice cavities by inserting sequence specific binding proteins may improve the diffraction quality, however, sequence independent methods to orient proteins within the DNA cavities still need to be developed. This strategy is particularly attractive, as some have already demonstrated that RNA-binding proteins are useful chaperones for RNA crystallization. Piccirilli and co-workers derived RNA-specific antibodies using synthetic phage display libraries, and showed that the antibody fragments promoted crystallization of RNA molecules[14]. Similarly, DNA tile binding antibodies could be identified through in vitro evolution and used for co-assembly of the DNA units and proteins into designed 3D crystals.

Recent developments in Free Electron Laser X-ray nanocrystallography have the potential to revolutionize the field of structural biology by providing highly focused coherent X-ray beams with a peak brilliance that is 10^9 higher than the X-ray beams at the most powerful synchrotron facilities[15]. Obtaining high quality diffraction patterns using FEL X-ray requires micron sized nanocrystals; it might possible to program the growth of 3D DNA lattices into finite nanocrystals with suitable dimensions by designing a 3D box that acts as a scaffold to nucleate the growth of a periodic lattice of DNA tiles. Growing 3D crystals with designed crystal morphologies and dimensions is undoubtedly an interesting topic in itself.

5.1.5 Scaffolded assembly

The development of scaffolded DNA origami represents a milestone in structural DNA nanotechnology[16]. While the complexity and robustness of 2D and 3D DNA

origami objects has increased over the past few years, researchers still lack basic understanding of the thermodynamics and kinetics of scaffolded assembly. Understanding the minutia of DNA origami formation will allow us to guide the design of more complex DNA nanostructures, optimize annealing protocols, and manipulate functionalized DNA nanostructures more effectively. Structurally speaking, we are still a long way from being able to weave a scaffold strand along arbitrary paths within a DNA origami structure, although some progress has been made in this direction. Recently, Yan and co-workers developed a novel strategy to fold Gridiron-like DNA origami structures[17]. In this work, interconnected four-arm junctions were used as vertices within a network of DNA fragments and measured distortion of the junctions from relaxed conformation allowed the scaffold strand to traverse through individual vertices in several directions. Despite this initial success, interlacing the scaffold strand through the vertices of multi-arm junctions remains a challenge that if achieved, would dramatically improve our ability to form aperiodic tiling patterns and polyhedral 3D structures using the DNA origami technique. Besides increasing complexity, scaling up the size of DNA origami and reducing the cost of staple strand synthesis are also important issues facing DNA nanotechnologists. Various strategies to address these limitations have been explored, including the use of longer single stranded scaffolds[18], double stranded scaffolds[19], origami of origami (super-origami)[20], and enzymatic production of staple strands on micro-array chips[21], which has the added benefit of greater fidelity than chemical synthesis. Researchers are relentlessly pushing forward to achieve more robust DNA origami technology.

5.1.6 Surface mediated assembly

DNA origami has shown great success in directing the assembly of nanoelectronic and photonic elements and has been used as a lithographic mask to etch nanoscale patterns on silicon and graphene substrates[22]. For practical device applications, it is highly desirable to achieve robust patterning of self-assembled DNA nanostructures on inorganic surfaces and several groups have developed unique strategies to organize DNA origami nanostructures on solid substrates[23-25]. The next logical step is to generate chemically functional surface features to facilitate patterning of DNA origami nanostructures into spatially addressable arrays. Surface mediated assembly may be the key to scaling up DNA nanostructure assemblies into wafer size arrays. Researchers have already shown that mica and silicon dioxide surfaces will mediate the assembly of small DNA tiles into millimeter range period 2D lattices[26]. The buffer conditions, especially the concentration and species of the ions present, may play a critical role in surface mediated diffusion of DNA nanostructures, an important factor that remains to be explored[27]. It would also be interesting to use fluidic 2D surfaces such as lipid bilayers to improve the surface mediated diffusion of DNA nanostructures[28].

5.1.7 Algorithmic assembly

In mathematics and computer science, an algorithm describes a set of simple instructions for solving a problem. However, if you look beyond their traditional context in mathematics, you will see that algorithms can be used to describe the process of self-assembly in the natural world. Consider the self-assembly of lipids into membranes, or viral proteins into capsids, or even just amino acids into intricately folded protein structures. Each process involves the spontaneous, or automatic, assembly of small components into larger, more complex structures. The process by which these structures

grow can be described as algorithmic. In each example, a limited number of molecular building blocks grow into higher order structures by following the growth rules encoded into the building blocks themselves. DNA tiles are information rich building blocks ideally suited for implementing algorithmic self-assembly. Originally proposed by Winfree, algorithmically self-assembled DNA nanostructure patterns have been experimentally demonstrated. For example, Winfree and co-workers showed that DNA double crossover tiles could be programmed to compute and grow into Sierpinski triangle[29] and binary counter assemblies[30]. They also showed that prescribed DNA origami displaying sticky-end capture probes function as effective nucleation seeds to grow algorithmic arrays while suppressing spurious nucleation, which is a major source of errors during algorithmic assembly[31]. The design of novel nucleation frames could improve the fidelity and robustness of algorithmic assemblies of DNA tiles. Other errors arise from sticky end mismatches between different tiles that share certain sticky end sequences. The kinetics of tile-tile association between the algorithmic building blocks should be carefully investigated to promote the desired computations and reduce any undesirable mismatches. Also, tile sets could be expanded beyond the typical double crossover DNA tiles to more complex or optimal geometries to facilitate multivalent and cooperative binding between the tiles and allow for improved understanding of the constraints that limit the scope of algorithmic assembly.

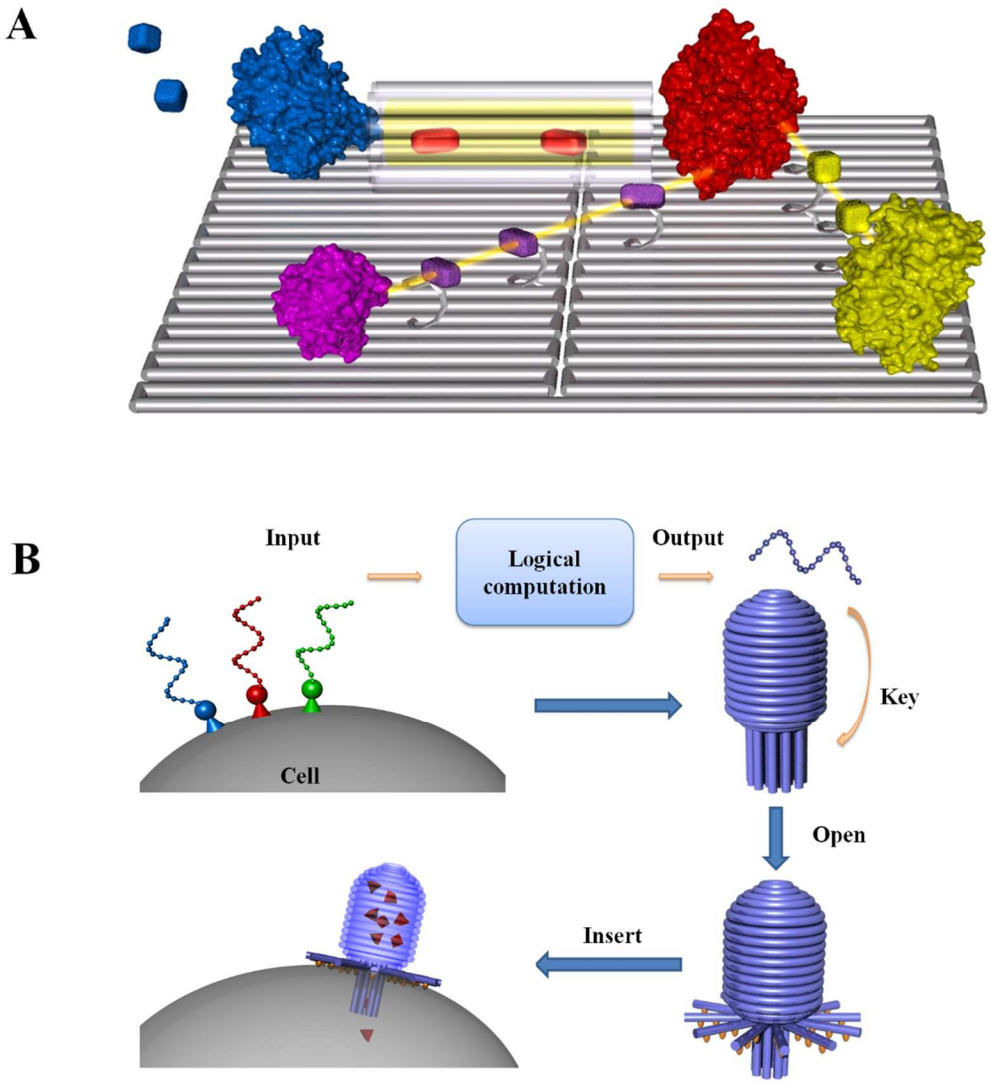


Figure 5.2 Illustration of potential applications of DNA nanotechnology: (A) Programming biochemical pathways with controlled input and output. (B) Design and implementation of theranostic nanodevices on targeted cell surfaces, that carry out functions such as compute, sense, release signal, trigger activation and deliver therapeutic molecules across the cell membrane.

5.1.8 Topological DNA nanostructures

In biological systems, there is a clear relationship between the specific structure of a biomolecule and its function. In particular, biopolymers are important molecules

whose structure supports the organization and functionality of cells. The topology of biopolymers can be exploited to facilitate tasks such as packing information bearing DNA molecules into tiny compartments within cells. Molecular topology is a fascinating and technically challenging topic that DNA nanotechnology is ideally suited to examine. Seeman and co-workers were the first to shown that topological structures such as knots and Borromean rings could be self-assembled from DNA by combining right-handed B-form and left-handed Z-form DNA together to create positive and negative nodes[32]. Yan and co-workers later used the DNA origami method to construct Möbius strip topological structures that could be reconfigured into catenanes and twisted topological ribbons through toehold mediated strand displacement[33]. More recently, Willner and co-workers developed strategies to inter-lock DNA rings into multi-ring catenanes[34]. Weizmann and coworkers just reported the assembly of complex knots and links by specifically configuring four-way DNA junctions[35]. Despite these interesting examples, the area of DNA based topological nanostructures is under-developed compared to the geometric structures that have been reported over the past decade. New construction strategies and topological targets should be identified to push the frontiers of DNA based molecular topology forward.

5.1.9 Self-replicating DNA nanostructures

Self-replication is an astounding process by which a molecule in a dynamic system makes an identical copy of itself. Biological cells, provided they have a suitable environment, reproduce by cell division. During cell division, linear DNA autonomously undergoes replication by enzyme-mediated processes and is transmitted to offspring. It is a considerable challenge to design and construct autonomous structures that mimic the

action of nucleic acid polymerases and are capable of replicating entire synthetic DNA systems non-enzymatically (Figure 5.1D). The first development in this direction was reported by Seeman's group in 2011[36]. They constructed a seven-tile seed and successfully generated several generations of progeny in a step-by-step manner. Winfree and co-workers recently showed that mechanically induced scission of 2D DNA crystals can accurately replicate self-assembled DNA nanopatterns by creating new fronts of crystal growth[37]. However, constructing autonomous self-replicating systems that do not require external manipulation remains a significant challenge. Pierce and co-workers demonstrated autocatalytic DNA duplex formation by way of a cross-catalytic circuit[38], yet extending this concept to independent formation of sophisticated DNA nanopatterns needs additional development.

5.2 Future Applications

The successful design and assembly of the DNA nano-systems discussed above will undoubtedly lead to many new opportunities and innovative applications. The information rich character of self-assembling DNA nanostructures in particular, will create many new frontiers for the application of designer DNA nanostructures as molecular scaffolds, sensors, computers, and robots. In the following section we will discuss the potential of DNA nanostructures to serve as scaffold for functional nanoelectronic and nanophotonic devices, to regulate protein interactions, and to create sense-compute-actuate elements for molecular medicine. However, these examples are in no way limiting, and the field has already demonstrated a tendency to grow in unexpected directions, surprising even the sagest of researchers.

5.2.1 Molecular scaffolds for nanophotonics or nanoelectronics

One of the most obvious applications of DNA a nanostructure is to direct the assembly of other, less controllable materials, as was discussed in the previous sections. We have seen several examples of spatially addressable DNA origami structures being used to organize nanoelectronic and photonic components. However, we have yet seen concrete examples of DNA nanostructures in functional nanoelectronic and photonic devices, where bottom-up, DNA directed assembly is interfaced with top-down, lithographic methods of micro- and macro-scale patterning. The latest developments in surface mediated self-assembly and site-specific control of chemical properties could enable more precise arrangement of these nanophotonic or nanoelectronic elements into regular, large-scale patterns that can be integrated with macroscopic systems.

5.2.2 Molecular scaffolds for enzyme cascades

DNA directed assembly of complex protein arrays is another area of development to watch for in the future. Enzymes, marvels of natural evolution, are intramolecular organizations of proteins that are capable of recognition, capture and activation of molecules, and regulation of biochemical processes. These protein complexes act as the central functional components of metabolism and reproduction in living systems[39]. The binding sites for substrates and cofactors are chemically specific, while the active sites are stereospecific and highly sensitive to conformational rearrangement. Inspired by nature, researchers have pursued a variety of strategies to regulate and control the catalytic activities of enzymes, as well as to understand the mechanism of enzyme function and pathways[40-44]. Compared to most conventional techniques, DNA

nanotechnology is a highly efficient and controllable strategy to achieve structural programmability and re-configurability through rational design and construction.

Assembling enzymes and cofactors on DNA nanostructure scaffolds have already allowed researchers to probe the essential parameters for modulating catalysis, such as intermolecular distance and relative spatial position[45-50]. One example of controlling the activity of an individual enzyme using DNA was reported in 2013, where the authors achieved mechanical regulation of the enzyme luciferase by attaching a DNA spring[51]. In the same year, a DNA tweezer-actuated enzyme nanoreactor was successfully constructed[52].

An even loftier and more valuable goal is to engineer highly programmed cascading enzyme pathways on DNA nanostructure platforms with control of input and output sequences. Achieving this goal would not only allow researchers to mimic the elegant enzyme cascades found in nature and attempt to understand their underlying mechanisms of action, but would facilitate the construction of artificial cascades that do not exist in nature (Figure 5.2A).

One major challenge in integrating multiple proteins into DNA nanostructures is to precisely define their relative orientation and position. A set of reliable and general methods for site-specific conjugation of proteins with oligonucleotides must be established in order to accommodate the diversity of proteins of interest. In an ideal system, a single protein with multiple coupling sites would be conjugated to unique DNA sequences to enable absolute orientational control of the protein relative to the DNA nanostructure. In this way, the active sites of the enzymes, in a multi-enzyme cascade for

example, could be precisely oriented to facilitate substrate-intermediate-product transfer and the overall enzymatic activity of the cascade could be optimized.

5.2.3 Molecular sense-compute-actuate devices

A far-reaching goal of structural DNA nanotechnology is to develop smart molecular machines that perform sense-compute-actuate mechanisms based on intrinsically information rich DNA molecules and structures (Figure 5.2B). For example, the development of smart molecular doctors would revolutionize the field of personalized medicine. A smart molecular doctor would have the same responsibilities as a real doctor, including diagnostic and therapeutic roles, but would operate entirely at the cellular level. Directly treating individual, diseased cells to cure them on the single cell level offers improved therapeutic efficiency and fewer side effects since smaller drug doses are required compared to that of conventional therapies.

Other targeted drug delivery systems based on multifunctional liposomes, polymersomes, and nanoparticles have already been developed[53]. DNA is an attractive material for theranostic applications, not only because of its inherent design modularity, structural programmability and biocompatibility, but also because DNA molecules of a particular sequence or with certain modifications can selectively bind, distinguish and communicate with target cells to trigger drug release. Researchers have made strides toward constructing DNA-based drug containers and DNA nanostructures that can be embedded into lipid bilayers[54], particularly after the establishment of the DNA origami method. The first DNA-origami box with a responsive lid that recognized a specific oligonucleotide key and subsequently opened was reported in 2009[55]. More recently,

researchers developed a DNA nano-barrel with two single stranded aptamer locks that were opened by the presence of target cells in vitro[56].

Performing DNA computation directly on the surface of cells, or in cellular environments, will facilitate in vivo targeting and drug release. Recently, Rudchenko, Stojanovic and colleagues engineered DNA strand displacement cascades that detected the presence of certain cell markers on the surface of cells[57]. In another report, Hemphill and Deiters successfully engineered oligonucleotide logic gates to detect specific microRNA inputs in live, mammalian cells[58]. As more complex and robust DNA based computing systems are developed, it may be possible to integrate them into cellular systems to control and trigger cellular functions such as gene expression, or interfere with the metabolic pathways[59]. Recently, researchers reported the construction of a consensus network that distinguishes between two different input signals and reports the majority signal[60]. By combining DNA computation based target cell detection with reconfigurable DNA-based drug containers, it may be possible to create a DNA nanorobot that can interface and communicate with living cells (Figure 5.2B).

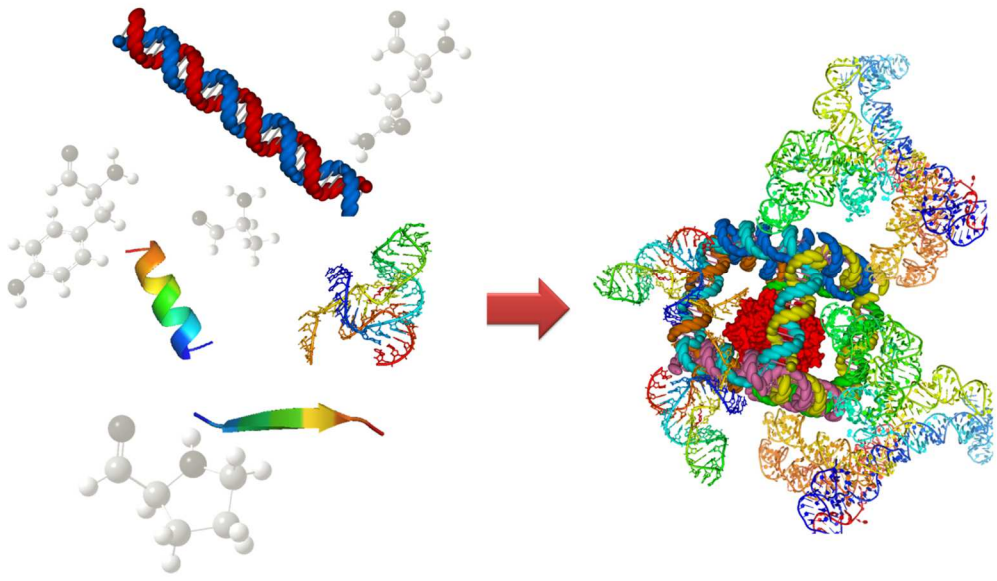


Figure 5.3 From nano- to Angstrom level control: engineering molecular tool-boxes composed of DNA, RNA, peptide, and protein molecules and their unnatural derivatives to extract new design rules and create complex, self-assembling structures with Angstrom level spatial control.

There are a number of critical issues that must be addressed before DNA nanorobots can be used for drug delivery in vivo. Researchers must find a way to protect DNA nanostructures from degradation by the intra- and extra-cellular nucleases and liver metabolism over long periods. Compact DNA nanostructures generally display relative stability against DNA nucleases for a short time (a few hours)[61-62]. In the future it will be important to increase resistance to biodegradation by using methods such as chemical cross-linking of selected DNA strands or designated DNA backbone modifications. Identifying the mechanisms by which DNA nanostructures enter cells without being damaged, and escape endosomal processing[63], is also a critical point. Other issues such as immunogenicity[64] and tissue distribution should also be considered.

The biggest obstacles to transforming DNA nanostructures from mere curiosities into real-world solutions are the cost of synthetic DNA, small production scales, typically low yield of complex 3D structures, and sensitivity of DNA to ionic strength, temperature and nucleases. Researchers have already begun to address these issues by optimizing origami design and folding strategies to increase assembly yields[65] and shorten assembly times[66], and by developing suitable purification strategies for large scale synthesis[67]. It is also important to develop biocompatible conditions for efficiently folding DNA nanostructures, rather than by thermal annealing under high magnesium concentrations[68-69].

5.3 From Nano to Angstrom Technology

Living cells are information rich, sophisticated machines that display Angstrom level organizational precision. Although DNA nanostructures are exquisitely programmable, they are only able to regulate biological molecules at a relatively coarse level compared to nature. If we want additional control, we must push the boundaries of nano-scale fabrication to the Ångstrom level. In contrast to DNA, RNA and proteins have more refined architectures with Ångstrom level features. These aspects of their organization have attracted increasing attention in the past decade. For example, several rationally designed RNA nanostructure have been constructed[70-71]. Methods for engineering designed proteins and nanostructured complexes using proteins have also begun to emerge[72-73]. The progress of characterization techniques such as cryo-EM, X-ray diffraction and NMR support the development of Ångstrom technology. In particular, the most recent developments in cryo-EM techniques allow crystallization-free structural

determination of large sized proteins that is comparable to X-ray methods[74-75]. Using DNA origami frames both as structural hosts and as references, the structure of DNA and RNA binding proteins may now be determined to Ångstrom level resolution by cryo-EM[76]. This advance will provide researchers with atomic level structural information (in conjunction with the structural solutions obtained from X-ray crystallography) that can be fed back into the design pipeline, elevating the field to unimaginable heights (Figure 5.3).

In summary, after more than 30 years of growth, structural DNA nanotechnology is transitioning from adolescence into adulthood. The field is crossing the boundaries of physics, chemistry, biology and engineering, and is poised to generate unique approaches and solutions to real world challenges in science and technology. In the next phase of structural DNA nanotechnology, novel interactions between DNA, RNA and proteins could be used to facilitate Ångstrom technology, and represent the major challenges and opportunities in molecular design, assembly, computing, and programming.

5.4 References

(1) Whitesides, G. M.; Grzybowski, B., Science 2002, 295, 2418-2421.

(2) Hyman, A. A.; Salser, S.; Drechsel, D. N.; Unwin, N.; Mitchison, T. J., Mol Biol Cell 1992, 3, 1155-1167.

(3) Desai, A.; Mitchison, T. J., Annu Rev Cell Dev Bi 1997, 13, 83-117.

(4) Dirks, R. M.; Pierce, N. A., P Natl Acad Sci USA 2004, 101, 15275-15278.

(5) Zhang, D. Y.; Hariadi, R. F.; Choi, H. M. T.; Winfree, E., Nat Commun 2013, 4, 1965.

(6) Zhang, D. Y.; Chen, S. X.; Yin, P., Nat Chem 2012, 4, 208-214.

(7) Shechtman, D.; Blech, I.; Gratias, D.; Cahn, J., Physical Review Letters 1984, 53, 1951-1953.

(8) Steurer, W., Chemical Society Reviews 2012, 41, 6717.

(9) Elser, V. V., Phys Rev Lett 1985, 54, 1730.

(10) Steurer, W., Chemical Society Reviews 2012, 41, 6719.

(11) de Boissieu, M., Chemical Society Reviews 2012, 41, 6778.

(12) Zheng, J. P.; Birktoft, J. J.; Chen, Y.; Wang, T.; Sha, R. J.; Constantinou, P. E.; Ginell, S. L.; Mao, C. D.; Seeman, N. C., Nature 2009, 461, 74-77.

(13) Yang, H.; Sleiman, H. F., Angew Chem Int Edit 2008, 47, 2443-2446.

(14) Ye, J. D.; Tereshko, V.; Frederiksen, J. K.; Koide, A.; Fellouse, F. A.; Sidhu, S. S.; Koide, S.; Kossiakoff, A. A.; Piccirilli, J. A., P Natl Acad Sci USA 2008, 105, 82-87.

(15) Chapman, H. N.; etal., Nature 2011, 470, 73-77.

(16) Rothemund, P. W. K., Nature 2006, 440, 297-302.

(17) Han, D. R.; Pal, S.; Yang, Y.; Jiang, S. X.; Nangreave, J.; Liu, Y.; Yan, H., Science 2013, 339, 1412-1415.

(18) Zhang, H. L.; Chao, J.; Pan, D.; Liu, H. J.; Huang, Q.; Fan, C. H., Chem Commun 2012, 48, 6405-6407.

(19) Hogberg, B.; Liedl, T.; Shih, W. M., J Am Chem Soc 2009, 131, 9154-9155.

(20) Zhao, Z.; Liu, Y.; Yan, H., Nano Lett 2011, 11, 2997-3002.

(21) Ducani, C.; Kaul, C.; Moche, M.; Shih, W. M.; Högberg, B., Nat Methods 2013, 10, 647-652.

(22) Jin, Z.; Sun, W.; Ke, Y.; Shih, C. J.; Paulus, G. L. C.; Wang, Q. H.; Mu, B.; Yin, P.; Strano, M. S., Nat Commun 2013, 4, 1663.

(23) Hung, A. M.; Cha, J. N., 2011, 749, 187-197.

(24) Kershner, R. J.; Bozano, L. D.; Micheel, C. M.; Hung, A. M.; Fornof, A. R.; Cha, J. N.; Rettner, C. T.; Bersani, M.; Frommer, J.; Rothemund, P. W. K.; Wallraff, G. M., Nat Nanotechnol 2009, 4, 557-561.

(25) Ding, B.; Wu, H.; Xu, W.; Zhao, Z.; Liu, Y.; Yu, H.; Yan, H., Nano Lett 2010, 10, 5065-5069.

(26) Sun, X. P.; Ko, S. H.; Zhang, C. A.; Ribbe, A. E.; Mao, C. D., J Am Chem Soc 2009, 131, 13248-13249.

(27) Woo, S.; Rothemund, P. W. K. In Self-assembly of two-dimensional DNA origami lattices using cation-controlled surface diffusion, Foundations of Nanoscience: Self-Assembled Architectures and Devices., Snowbird, Utah, Snowbird, Utah, 2014; p 93.

(28) Endo, M.; Suzuki, Y.; Yang, Y.; Sugiyama, H. In Photoresponsive DNA nanostructures, single-molecule imaging and controlled assembly, Foundations of Nanoscience: Self-Assembled Architectures and Devices., Snowbird, Utah, Snowbird, Utah, 2014; p 7.

(29) Rothemund, P. W. K.; Papadakis, N.; Winfree, E., Plos Biol 2004, 2, 2041-2053.

(30) Barish, R. D.; Rothemund, P. W. K.; Winfree, E., Nano Lett 2005, 5, 2586-2592.

(31) Barish, R. D.; Schulman, R.; Rothemund, P. W. K.; Winfree, E., P Natl Acad Sci USA 2009, 106, 6054-6059.

(32) Mao, C. D.; Sun, W. Q.; Seeman, N. C., Nature 1997, 386, 137-138.

(33) Han, D. R.; Pal, S.; Liu, Y.; Yan, H., Nat Nanotechnol 2010, 5, 712-717.

(34) Elbaz, J.; Cecconello, A.; Fan, Z. Y.; Govorov, A. O.; Willner, I., Nat Commun 2013, 4.

(35) Liu, D.; Chen, G.; Weizmann, Y. In Creating Complex molecular Topologies by Configuring DNA Four-Way Junctions Foundations of Nanoscience: Self-Assembled Architectures and Devices., Snowbird, Utah, Snowbird, Utah, 2014; p 34.

(36) Wang, T.; Sha, R. J.; Dreyfus, R.; Leunissen, M. E.; Maass, C.; Pine, D. J.; Chaikin, P. M.; Seeman, N. C., Nature 2011, 478, 225-228.

(37) Schulman, R.; Yurke, B.; Winfree, E., P Natl Acad Sci USA 2012, 109, 6405-6410.

(38) Yin, P.; Choi, H. M. T.; Calvert, C. R.; Pierce, N. A., Nature 2008, 451, 318-322.

(39) Bairoch, A., Nucleic Acids Res 2000, 28, 304-305.

(40) Hammes, G. G.; Wu, C. W., Science 1971, 172, 1205-1211.

(41) Drews, J., Science 2000, 287, 1960-1964.

(42) Khosla, C.; Harbury, P. B., Nature 2001, 409, 247-252.

(43) Ostermeier, M., Curr Opin Struc Biol 2009, 19, 442-448.

(44) Fu, Y.; Zeng, D.; Chao, J.; Jin, Y.; Zhang, Z.; Liu, H.; Li, D.; Ma, H.; Huang, Q.; Gothelf, K. V.; Fan, C., J Am Chem Soc 2013, 135, 696-702.

(45) Wilner, O. I.; Weizmann, Y.; Gill, R.; Lioubashevski, O.; Freeman, R.; Willner, I., Nat Nanotechnol 2009, 4, 249-254.

(46) Saghatelian, A.; Guckian, K. M.; Thayer, D. A.; Ghadiri, M. R., J Am Chem Soc 2003, 125, 344-345.

(47) Simon, P.; Dueymes, C.; Fontecave, M.; Decout, J. L., Angew Chem Int Edit 2005, 44, 2764-2767.

(48) Fu, J. L.; Liu, M. H.; Liu, Y.; Woodbury, N. W.; Yan, H., J Am Chem Soc 2012, 134, 5516-5519.

(49) Fu, Y. M.; Zeng, D. D.; Chao, J.; Jin, Y. Q.; Zhang, Z.; Liu, H. J.; Li, D.; Ma, H. W.; Huang, Q.; Gothelf, K. V.; Fan, C. H., J Am Chem Soc 2013, 135, 696-702.

(50) Fu, J.; Yang, Y.; Buck, A. J.; Liu, M.; Liu, Y.; Walter, N. G.; Woodbury, N. W.; Yan, H., Nat Nanotechnol 2014, 9, 531-536.

(51) Tseng, C. Y.; Zocchi, G., J Am Chem Soc 2013, 135, 11879-11886.

(52) Liu, M. H.; Fu, J. L.; Hejesen, C.; Yang, Y. H.; Woodbury, N. W.; Gothelf, K.; Liu, Y.; Yan, H., Nat Commun 2013, 4, 2127.

(53) Lammers, T.; Aime, S.; Hennink, W. E.; Storm, G.; Kiessling, F., Accounts Chem Res 2011, 44, 1029-1038.

(54) Langecker, M.; Arnaut, V.; Martin, T. G.; List, J.; Renner, S.; Mayer, M.; Dietz, H.; Simmel, F. C., Science 2012, 338, 932-936.

(55) Andersen, E. S.; Dong, M.; Nielsen, M. M.; Jahn, K.; Subramani, R.; Mamdouh, W.; Golas, M. M.; Sander, B.; Stark, H.; Oliveira, C. L. P.; Pedersen, J. S.; Birkedal, V.; Besenbacher, F.; Gothelf, K. V.; Kjems, J., Nature 2009, 459, 73-75.

(56) Douglas, S. M.; Bachelet, I.; Church, G. M., Science 2012, 335, 831-834.

(57) Rudchenko, M.; Taylor, S.; Pallavi, P.; Dechkovskaia, A.; Khan, S.; Butler, V. P.; Rudchenko, S.; Stojanovic, M. N., Nat Nanotechnol 2013, 8, 580-586.

(58) Hemphill, J.; Deiters, A., J Am Chem Soc 2013, 135, 10512-10518.

(59) Yin, P. In Synthetic organization and regulation with DNA/RNA bricks, Foundations of Nanoscience: Self-Assembled Architectures and Devices., Snowbird, Utah, Snowbird, Utah, 2014; p 20.

(60) Chen, Y. J.; Dalchau, N.; Srinivas, N.; Phillips, A.; Cardelli, L.; Soloveichik, D.; Seelig, G., Nat Nanotechnol 2013, 8, 755-762.

(61) Mei, Q. A.; Wei, X. X.; Su, F. Y.; Liu, Y.; Youngbull, C.; Johnson, R.; Lindsay, S.; Yan, H.; Meldrum, D., Nano Lett 2011, 11, 1477-1482.

(62) Castro, C. E.; Kilchherr, F.; Kim, D. N.; Shiao, E. L.; Wauer, T.; Wortmann, P.; Bathe, M.; Dietz, H., Nat Methods 2011, 8, 221-229.

(63) Modi, S.; Nizak, C.; Surana, S.; Halder, S.; Krishnan, Y., Nat Nanotechnol 2013, 8, 459-467.

(64) Perrault, S. D.; Shih, W. M., Acs Nano 2014, 140423081328001.

(65) Ke, Y. G.; Bellot, G.; Voigt, N. V.; Fradkov, E.; Shih, W. M., Chem Sci 2012, 3, 2587-2597.

(66) Sobczak, J. P. J.; Martin, T. G.; Gerling, T.; Dietz, H., Science 2012, 338, 1458-1461.

(67) Lin, C. X.; Perrault, S. D.; Kwak, M.; Graf, F.; Shih, W. M., Nucleic Acids Res 2013, 41, e40.

(68) Martin, T. G.; Dietz, H., Nat Commun 2012, 3, 1103.

(69) Myhrvold, C.; Dai, M. J.; Silver, P. A.; Yin, P., Nano Lett 2013, 13, 4242-4248.

(70) Guo, P. X., Nat Nanotechnol 2010, 5, 833-842.

(71) Shukla, G. C.; Haque, F.; Tor, Y.; Wilhelmsson, L. M.; Toulmé, J.-J.; Isambert, H.; Guo, P.; Rossi, J. J.; Tenenbaum, S. A.; Shapiro, B. A., Acs Nano 2011, 5, 3405-3418.

(72) Lai, Y. T.; Cascio, D.; Yeates, T. O., Science 2012, 336, 1129-1129.

(73) King, N. P.; Sheffler, W.; Sawaya, M. R.; Vollmar, B. S.; Sumida, J. P.; Andre, I.; Gonen, T.; Yeates, T. O.; Baker, D., Science 2012, 336, 1171-1174.

(74) Li, X. M.; Mooney, P.; Zheng, S.; Booth, C. R.; Braunfeld, M. B.; Gubbens, S.; Agard, D. A.; Cheng, Y. F., Nat Methods 2013, 10, 584-590.

(75) Liao, M.; Cao, E.; Julius, D.; Cheng, Y., Nature 2013, 504, 107-112.

(76) Scheres, S. In How self-assembly could exploit high-resolution cryo-TEM and vice versa, Foundations of Nanoscience: Self-Assembled Architectures and Devices., Snowbird, Utah, Snowbird, Utah, 2014; p 55.

REFERENCES

Chapter 1 Reference

(1) Aldaye, F. A.; Palmer, A. L.; Sleiman, H. F., *Science* **2008,** 321, 1795-1799.

(2) Li, H. Y.; Carter, J. D.; LaBean, T. H., *Mater Today* **2009,** 12, 24-32.

(3) Seeman, N. C., *Annu Rev Biochem* **2010,** 79, 65-87.

(4) Service, R. F., *Science* **2011,** 332, 1140-1142.

(5) Zadegan, R. M.; Norton, M. L., *Int J Mol Sci* **2012,** 13, 7149-7162.

(6) Seeman, N. C., *J Theor Biol* **1982,** 99, 237-247.

(7) Seeman, N. C., *Journal of Biomolecular Structure and Dynamics* **1990,** 8, 573-581.

(8) Andersen, E. S.; Dong, M. D.; Nielsen, M. M.; Jahn, K.; Lind-Thomsen, A.; Mamdouh, W.; Gothelf, K. V.; Besenbacher, F.; Kjems, J., *Acs Nano* **2008,** 2, 1213-1218.

(9) Douglas, S. M.; Marblestone, A. H.; Teerapittayanon, S.; Vazquez, A.; Church, G. M.; Shih, W. M., *Nucleic Acids Res* **2009,** 37, 5001-5006.

(10) Williams, S.; Lund, K.; Lin, C.; Wonka, P.; Lindsay, S.; Yan, H., **2009,** 5347, 90-101.

(11) Zadeh, J. N.; Steenberg, C. D.; Bois, J. S.; Wolfe, B. R.; Pierce, M. B.; Khan, A. R.; Dirks, R. M.; Pierce, N. A., *J Comput Chem* **2011,** 32, 170-173.

(12) Kim, D. N.; Kilchherr, F.; Dietz, H.; Bathe, M., *Nucleic Acids Res* **2012,** 40, 2862-2868.

(13) Winfree, E.; Liu, F. R.; Wenzler, L. A.; Seeman, N. C., *Nature* **1998,** 394, 539-544.

(14) Yan, H.; Park, S. H.; Finkelstein, G.; Reif, J. H.; LaBean, T. H., *Science* **2003,** 301, 1882-1884.

(15) He, Y.; Tian, Y.; Ribbe, A. E.; Mao, C. D., *J Am Chem Soc* **2006,** 128, 15978-15979.

(16) Zheng, J. P.; Birktoft, J. J.; Chen, Y.; Wang, T.; Sha, R. J.; Constantinou, P. E.; Ginell, S. L.; Mao, C. D.; Seeman, N. C., *Nature* **2009,** 461, 74-77.

(17) LaBean, T. H.; Yan, H.; Kopatsch, J.; Liu, F.; Winfree, E.; Reif, J. H.; Seeman, N. C., *J Am Chem Soc* **2000,** 122, 1848-1860.

(18) He, Y.; Chen, Y.; Liu, H. P.; Ribbe, A. E.; Mao, C. D., *J Am Chem Soc* **2005,** 127, 12202-12203.

(19) Malo, J.; Mitchell, J. C.; Venien-Bryan, C.; Harris, J. R.; Wille, H.; Sherratt, D. J.; Turberfield, A. J., *Angew Chem Int Edit* **2005,** 44, 3057-3061.

(20) Majumder, U.; Rangnekar, A.; Gothelf, K. V.; Reif, J. H.; LaBean, T. H., *J Am Chem Soc* **2011,** 133, 3843-3845.

(21) Chen, J. H.; Seeman, N. C., *Nature* **1991,** 350, 631-633.

(22) Goodman, R. P.; Schaap, I. A. T.; Tardin, C. F.; Erben, C. M.; Berry, R. M.; Schmidt, C. F.; Turberfield, A. J., *Science* **2005,** 310, 1661-1665.

(23) He, Y.; Ye, T.; Su, M.; Zhang, C.; Ribbe, A. E.; Jiang, W.; Mao, C. D., *Nature* **2008,** 452, 198-201.

(24) Aldaye, F. A.; Sleiman, H. F., *J Am Chem Soc* **2007,** 129, 13376-13377.

(25) Zhang, Y.; Seeman, N. C., *J Am Chem Soc* **1994,** 116, 1661-1669.

(26) Shih, W. M.; Quispe, J. D.; Joyce, G. F., *Nature* **2004,** 427, 618-621.

(27) Erben, C. M.; Goodman, R. P.; Turberfield, A. J., *J Am Chem Soc* **2007,** 129, 6992-6993.

(28) Rothemund, P. W. K.; Papadakis, N.; Winfree, E., *Plos Biol* **2004,** 2, 2041-2053.

(29) Barish, R. D.; Rothemund, P. W. K.; Winfree, E., *Nano Lett* **2005,** 5, 2586-2592.

(30) Winfree, E.; Bekbolatov, R., *Lect Notes Comput Sc* **2004,** 2943, 126-144.

(31) Sahu, S.; Reif, J. H., *Algorithmica* **2010,** 56, 480-504.

(32) Rothemund, P. W. K., *Nature* **2006,** 440, 297-302.

(33) Ke, Y. G.; Sharma, J.; Liu, M. H.; Jahn, K.; Liu, Y.; Yan, H., *Nano Lett* **2009,** 9, 2445-2447.

(34) Andersen, E. S.; Dong, M.; Nielsen, M. M.; Jahn, K.; Subramani, R.; Mamdouh, W.; Golas, M. M.; Sander, B.; Stark, H.; Oliveira, C. L. P.; Pedersen, J. S.; Birkedal, V.; Besenbacher, F.; Gothelf, K. V.; Kjems, J., *Nature* **2009,** 459, 73-75.

(35) Douglas, S. M.; Dietz, H.; Liedl, T.; Hogberg, B.; Graf, F.; Shih, W. M., *Nature* **2009,** 459, 414-418.

(36) Endo, M.; Hidaka, K.; Kato, T.; Namba, K.; Sugiyama, H., *J Am Chem Soc* **2009,** 131, 15570-15571.

(37) Dietz, H.; Douglas, S. M.; Shih, W. M., *Science* **2009,** 325, 725-730.

(38) Han, D. R.; Pal, S.; Nangreave, J.; Deng, Z. T.; Liu, Y.; Yan, H., *Science* **2011,** 332, 342-346.

(39) Woo, S.; Rothemund, P. W. K., *Nat Chem* **2011,** 3, 620-627.

(40) Liu, W. Y.; Zhong, H.; Wang, R. S.; Seeman, N. C., *Angew Chem Int Edit* **2011,** 50, 264-267.

(41) Zhao, Z.; Liu, Y.; Yan, H., *Nano Lett* **2011,** 11, 2997-3002.

(42) Zhang, H. L.; Chao, J.; Pan, D.; Liu, H. J.; Huang, Q.; Fan, C. H., *Chem Commun* **2012,** 48, 6405-6407.

(43) Pound, E.; Ashton, J. R.; Becerril, H. c. A.; Woolley, A. T., *Nano Lett* **2009,** 9, 4302-4305.

(44) Han, D. R.; Pal, S.; Yang, Y.; Jiang, S. X.; Nangreave, J.; Liu, Y.; Yan, H., *Science* **2013,** 339, 1412-1415.

(45) Wei, B.; Dai, M. J.; Yin, P., *Nature* **2012,** 485, 623-626.

(46) Ke, Y. G.; Ong, L. L.; Shih, W. M.; Yin, P., *Science* **2012,** 338, 1177-1183.

(47) Yurke, B.; Turberfield, A. J.; Mills, A. P.; Simmel, F. C.; Neumann, J. L., *Nature* **2000,** 406, 605-608.

(48) Yin, P.; Choi, H. M. T.; Calvert, C. R.; Pierce, N. A., *Nature* **2008,** 451, 318-322.

(49) Lund, K.; Manzo, A. J.; Dabby, N.; Michelotti, N.; Johnson-Buck, A.; Nangreave, J.; Taylor, S.; Pei, R. J.; Stojanovic, M. N.; Walter, N. G.; Winfree, E.; Yan, H., *Nature* **2010,** 465, 206-210.

(50) Gu, H. Z.; Chao, J.; Xiao, S. J.; Seeman, N. C., *Nature* **2010,** 465, 202-205.

(51) Yin, P.; Hariadi, R. F.; Sahu, S.; Choi, H. M. T.; Park, S. H.; LaBean, T. H.; Reif, J. H., *Science* **2008,** 321, 824-826.

(52) Mao, C. D.; Sun, W. Q.; Shen, Z. Y.; Seeman, N. C., *Nature* **1999,** 397, 144-146.

(53) Yan, H.; Zhang, X. P.; Shen, Z. Y.; Seeman, N. C., *Nature* **2002,** 415, 62-65.

(54) Li, J. W. J.; Tan, W. H., *Nano Lett* **2002,** 2, 315-318.

(55) Liu, D. S.; Balasubramanian, S., *Angew Chem Int Edit* **2003,** 42, 5734-5736.

(56) Li, Y. M.; Zhang, C.; Tian, C.; Mao, C. D., *Org Biomol Chem* **2014,** 12, 2543-2546.

(57) Feng, L. P.; Park, S. H.; Reif, J. H.; Yan, H., *Angew Chem Int Edit* **2003,** 42, 4342-4346.

(58) Goodman, R. P.; Heilemann, M.; Doose, S.; Erben, C. M.; Kapanidis, A. N.; Turberfield, A. J., *Nat Nanotechnol* **2008,** 3, 93-96.

(59) Maye, M. M.; Kumara, M. T.; Nykypanchuk, D.; Sherman, W. B.; Gang, O., *Nat Nanotechnol* **2010,** 5, 116-120.

(60) Han, D. R.; Pal, S.; Liu, Y.; Yan, H., *Nat Nanotechnol* **2010,** 5, 712-717.

(61) Zhang, F.; Nangreave, J.; Liu, Y.; Yan, H., *Nano Lett* **2012,** 12, 3290-3295.

(62) Yang, Y. Y.; Endo, M.; Hidaka, K.; Sugiyama, H., *J Am Chem Soc* **2012,** 134, 20645-20653.

(63) Tian, Y.; Mao, C. D., *J Am Chem Soc* **2004,** 126, 11410-11411.

(64) Rajendran, A.; Endo, M.; Hidaka, K.; Sugiyama, H., *J Am Chem Soc* **2013,** 135, 1117-1123.

(65) Omabegho, T.; Sha, R.; Seeman, N. C., *Science* **2009,** 324, 67-71.

(66) Wickham, S. F. J.; Endo, M.; Katsuda, Y.; Hidaka, K.; Bath, J.; Sugiyama, H.; Turberfield, A. J., *Nat Nanotechnol* **2011,** 6, 166-169.

(67) Wickham, S. F. J.; Bath, J.; Katsuda, Y.; Endo, M.; Hidaka, K.; Sugiyama, H.; Turberfield, A. J., *Nat Nanotechnol* **2012,** 7, 169-173.

(68) He, Y.; Liu, D. R., *Nat Nanotechnol* **2010,** 5, 778-782.

(69) Zheng, J. W.; Constantinou, P. E.; Micheel, C.; Alivisatos, A. P.; Kiehl, R. A.; Seeman, N. C., *Nano Lett* **2006,** 6, 1502-1504.

(70) Pal, S.; Deng, Z. T.; Wang, H. N.; Zou, S. L.; Liu, Y.; Yan, H., *J Am Chem Soc* **2011,** 133, 17606-17609.

(71) Deng, Z. T.; Samanta, A.; Nangreave, J.; Yan, H.; Liu, Y., *J Am Chem Soc* **2012,** 134, 17424-17427.

(72) Kuzyk, A.; Schreiber, R.; Fan, Z. Y.; Pardatscher, G.; Roller, E. M.; Hogele, A.; Simmel, F. C.; Govorov, A. O.; Liedl, T., *Nature* **2012,** 483, 311-314.

(73) Chhabra, R.; Sharma, J.; Ke, Y. G.; Liu, Y.; Rinker, S.; Lindsay, S.; Yan, H., *J Am Chem Soc* **2007,** 129, 10304-10305.

(74) Sacca, B.; Meyer, R.; Erkelenz, M.; Kiko, K.; Arndt, A.; Schroeder, H.; Rabe, K. S.; Niemeyer, C. M., *Angew Chem Int Edit* **2010,** 49, 9378-9383.

(75) Tan, S. J.; Campolongo, M. J.; Luo, D.; Cheng, W. L., *Nat Nanotechnol* **2011,** 6, 268-276.

(76) Ding, B.; Deng, Z.; Yan, H.; Cabrini, S.; Zuckermann, R. N.; Bokor, J., *J Am Chem Soc* **2010,** 132, 3248-3249.

(77) Klein, W. P.; Schmidt, C. N.; Rapp, B.; Takabayashi, S.; Knowlton, W. B.; Lee, J.; Yurke, B.; Hughes, W. L.; Graugnard, E.; Kuang, W., *Nano Lett* **2013,** 13, 3850-3856.

(78) Dutta, P. K.; Varghese, R.; Nangreave, J.; Lin, S.; Yan, H.; Liu, Y., *J Am Chem Soc* **2011,** 133, 11985-11993.

(79) Langecker, M.; Arnaut, V.; Martin, T. G.; List, J.; Renner, S.; Mayer, M.; Dietz, H.; Simmel, F. C., *Science* **2012,** 338, 932-936.

(80) Chhabra, R.; Sharma, J.; Wang, H.; Zou, S.; Lin, S.; Yan, H.; Lindsay, S.; Liu, Y., *Nanotechnology* **2009,** 20, 485201.

(81) Pal, S.; Dutta, P.; Wang, H.; Deng, Z.; Zou, S.; Yan, H.; Liu, Y., *The Journal of Physical Chemistry C* **2013,** 117, 12735-12744.

(82) Acuna, G. P.; Moller, F. M.; Holzmeister, P.; Beater, S.; Lalkens, B.; Tinnefeld, P., *Science* **2012,** 338, 506-510.

(83) Berardi, M. J.; Shih, W. M.; Harrison, S. C.; Chou, J. J., *Nature* **2011,** 476, 109-113.

(84) Selmi, D. N.; Adamson, R. J.; Attrill, H.; Goddard, A. D.; Gilbert, R. J. C.; Watts, A.; Turberfield, A. J., *Nano Lett* **2011,** 11, 657-660.

(85) Delebecque, C. J.; Lindner, A. B.; Silver, P. A.; Aldaye, F. A., *Science* **2011,** 333, 470-474.

(86) Erkelenz, M.; Kuo, C. H.; Niemeyer, C. M., *J Am Chem Soc* **2011,** 133, 16111-16118.

(87) Wilner, O. I.; Weizmann, Y.; Gill, R.; Lioubashevski, O.; Freeman, R.; Willner, I., *Nat Nanotechnol* **2009,** 4, 249-254.

(88) Fu, J. L.; Liu, M. H.; Liu, Y.; Woodbury, N. W.; Yan, H., *J Am Chem Soc* **2012,** 134, 5516-5519.

(89) Fu, J.; Yang, Y.; Buck, A. J.; Liu, M.; Liu, Y.; Walter, N. G.; Woodbury, N. W.; Yan, H., *Nat Nanotechnol* **2014,** 9, 531-536.

(90) Hariadi, R. F.; Cale, M.; Sivaramakrishnan, S., *P Natl Acad Sci USA* **2014,** 111, 4091-4096.

(91) Derr, N. D.; Goodman, B. S.; Jungmann, R.; Leschziner, A. E.; Shih, W. M.; Reck-Peterson, S. L., *Science* **2012,** 338, 662-665.

(92) Endo, M.; Katsuda, Y.; Hidaka, K.; Sugiyama, H., *J Am Chem Soc* **2010,** 132, 1592-1597.

(93) Douglas, S. M.; Bachelet, I.; Church, G. M., *Science* **2012,** 335, 831-834.

(94) Lee, H.; Lytton-Jean, A. K. R.; Chen, Y.; Love, K. T.; Park, A. I.; Karagiannis, E. D.; Sehgal, A.; Querbes, W.; Zurenko, C. S.; Jayaraman, M.; Peng, C. G.; Charisse, K.; Borodovsky, A.; Manoharan, M.; Donahoe, J. S.; Truelove, J.; Nahrendorf, M.; Langer, R.; Anderson, D. G., *Nat Nanotechnol* **2012,** 7, 389-393.

(95) Liu, X. W.; Xu, Y.; Yu, T.; Clifford, C.; Liu, Y.; Yan, H.; Chang, Y., *Nano Lett* **2012,** 12, 4254-4259.

(96) Amir, Y.; Ben-Ishay, E.; Levner, D.; Ittah, S.; Abu-Horowitz, A.; Bachelet, I., *Nat Nanotechnol* **2014,** 9, 353-357.

(97) Ke, Y.; Lindsay, S.; Chang, Y.; Liu, Y.; Yan, H., *Science* **2008,** 319, 180-183.

(98) Subramanian, H. K. K.; Chakraborty, B.; Sha, R.; Seeman, N. C., *Nano Lett* **2011,** 11, 910-913.

(99) Endo, M.; Tatsumi, K.; Terushima, K.; Katsuda, Y.; Hidaka, K.; Harada, Y.; Sugiyama, H., *Angew Chem Int Edit* **2012,** 51, 8778-8782.

(100) Endo, M.; Yang, Y. Y.; Suzuki, Y.; Hidaka, K.; Sugiyama, H., *Angew Chem Int Edit* **2012,** 51, 10518-10522.

(101) Li, J.; Pei, H.; Zhu, B.; Liang, L.; Wei, M.; He, Y.; Chen, N.; Li, D.; Huang, Q.; Fan, C. H., *Acs Nano* **2011,** 5, 8783-8789.

Chapter 2 Reference

(1) Grünbaum, B.; Shephard, G. C., *Tilings and Patterns*. W.H. Freeman: 1987.

(2) David, S.; Chelnokov, A.; Lourtioz, J. M., *Ieee J Quantum Elect* **2001,** 37, 1427-1434.

(3) Talapin, D. V.; Shevchenko, E. V.; Bodnarchuk, M. I.; Ye, X. C.; Chen, J.; Murray, C. B., *Nature* **2009,** 461, 964-967.

(4) Seeman, N. C., *J Theor Biol* **1982,** 99, 237-247.

(5) Rothemund, P. W. K., *Nature* **2006,** 440, 297-302.

(6) Winfree, E.; Liu, F. R.; Wenzler, L. A.; Seeman, N. C., *Nature* **1998,** 394, 539-544.

(7) Seeman, N. C., *Nature* **2003,** 421, 427-431.

(8) Gothelf, K. V.; LaBean, T. H., *Org Biomol Chem* **2005,** 3, 4023-4037.

(9) Voigt, N. V.; Torring, T.; Rotaru, A.; Jacobsen, M. F.; Ravnsbaek, J. B.; Subramani, R.; Mamdouh, W.; Kjems, J.; Mokhir, A.; Besenbacher, F.; Gothelf, K. V., *Nat Nanotechnol* **2010,** 5, 200-203.

(10) Deng, Z. T.; Samanta, A.; Nangreave, J.; Yan, H.; Liu, Y., *J Am Chem Soc* **2012,** 134, 17424-17427.

(11) Yan, H.; Park, S. H.; Finkelstein, G.; Reif, J. H.; LaBean, T. H., *Science* **2003,** 301, 1882-1884.

(12) He, Y.; Chen, Y.; Liu, H. P.; Ribbe, A. E.; Mao, C. D., *J Am Chem Soc* **2005,** 127, 12202-12203.

(13) He, Y.; Tian, Y.; Ribbe, A. E.; Mao, C. D., *J Am Chem Soc* **2006,** 128, 15978-15979.

(14) He, Y.; Tian, Y.; Chen, Y.; Ribbe, A. E.; Mao, C. D., *Chem Commun* **2007**, 165-167.

(15) Zhang, C.; Su, M.; He, Y.; Zhao, X.; Fang, P. A.; Ribbe, A. E.; Jiang, W.; Mao, C. D., *P Natl Acad Sci USA* **2008,** 105, 10665-10669.

(16) Chavey, D., *Comput Math Appl* **1989,** 17, 147-165.

(17) He, Y.; Ye, T.; Su, M.; Zhang, C.; Ribbe, A. E.; Jiang, W.; Mao, C. D., *Nature* **2008,** 452, 198-201.

(18) Ke, Y. G.; Liu, Y.; Zhang, J. P.; Yan, H., *J Am Chem Soc* **2006,** 128, 4414-4421.

(19) Maune, H. T.; Han, S. P.; Barish, R. D.; Bockrath, M.; Goddard, W. A.; Rothemund, P. W. K.; Winfree, E., *Nat Nanotechnol* **2010,** 5, 61-66.

(20) Mitchell, J. C.; Harris, J. R.; Malo, J.; Bath, J.; Turberfield, A. J., *J Am Chem Soc* **2004,** 126, 16342-16343.

(21) Mikhael, J.; Roth, J.; Helden, L.; Bechinger, C., *Nature* **2008,** 454, 501-504.

Chapter 3 Reference

(1) Aldaye, F. A.; Palmer, A. L.; Sleiman, H. F., *Science* **2008,** 321, 1795-1799.

(2) Seeman, N. C., *Annu Rev Biochem* **2010,** 79, 65-87.

(3) Pinheiro, A. V.; Han, D. R.; Shih, W. M.; Yan, H., *Nat Nanotechnol* **2011,** 6, 763-772.

(4) Zhang, F.; Nangreave, J.; Liu, Y.; Yan, H., *J Am Chem Soc* **2014,** 136, 11198-11211.

(5) Winfree, E.; Liu, F. R.; Wenzler, L. A.; Seeman, N. C., *Nature* **1998,** 394, 539-544.

(6) Rothemund, P. W. K., *Nature* **2006,** 440, 297-302.

(7) He, Y.; Ye, T.; Su, M.; Zhang, C.; Ribbe, A. E.; Jiang, W.; Mao, C. D., *Nature* **2008,** 452, 198-U41.

(8) Douglas, S. M.; Dietz, H.; Liedl, T.; Hogberg, B.; Graf, F.; Shih, W. M., *Nature* **2009,** 459, 414-418.

(9) Wei, B.; Dai, M. J.; Yin, P., *Nature* **2012,** 485, 623.

(10) Ke, Y. G.; Ong, L. L.; Shih, W. M.; Yin, P., *Science* **2012,** 338, 1177-1183.

(11) Andersen, E. S.; Dong, M.; Nielsen, M. M.; Jahn, K.; Subramani, R.; Mamdouh, W.; Golas, M. M.; Sander, B.; Stark, H.; Oliveira, C. L. P.; Pedersen, J. S.; Birkedal, V.; Besenbacher, F.; Gothelf, K. V.; Kjems, J., *Nature* **2009,** 459, 73-U75.

(12) Dietz, H.; Douglas, S. M.; Shih, W. M., *Science* **2009,** 325, 725-730.

(13) Han, D. R.; Pal, S.; Nangreave, J.; Deng, Z. T.; Liu, Y.; Yan, H., *Science* **2011,** 332, 342-346.

(14) Han, D. R.; Pal, S.; Yang, Y.; Jiang, S. X.; Nangreave, J.; Liu, Y.; Yan, H., *Science* **2013,** 339, 1412-1415.

(15) He, Y.; Chen, Y.; Liu, H. P.; Ribbe, A. E.; Mao, C. D., *J Am Chem Soc* **2005,** 127, 12202-12203.

(16) Yan, H.; Park, S. H.; Finkelstein, G.; Reif, J. H.; LaBean, T. H., *Science* **2003,** 301, 1882-1884.

(17) He, Y.; Tian, Y.; Ribbe, A. E.; Mao, C. D., *J Am Chem Soc* **2006,** 128, 15978-15979.

(18) Liu, Y.; Ke, Y. G.; Yan, H., *J Am Chem Soc* **2005,** 127, 17140-17141.

(19) Yurke, B.; Turberfield, A. J.; Mills, A. P.; Simmel, F. C.; Neumann, J. L., *Nature* **2000,** 406, 605-608.

(20) Shen, Z. Y.; Yan, H.; Wang, T.; Seeman, N. C., *J Am Chem Soc* **2004,** 126, 1666-1674.

(21) Geary, C.; Rothemund, P. W. K.; Andersen, E. S., *Science* **2014,** 345, 799-804.

Chapter 4 Reference

(1) Seeman, N. C., *J Theor Biol* **1982,** 99, 237-247.

(2) Kallenbach, N. R.; Ma, R. I.; Seeman, N. C., *Nature* **1983,** 305, 829-831.

(3) Seeman, N. C., *Annu Rev Biochem* **2010,** 79, 65-87.

(4) Mao, C. D., *Nat Nanotechnol* **2008,** 3, 75-76.

(5) Rothemund, P. W. K., *Nature* **2006,** 440, 297-302.

(6) Aldaye, F. A.; Palmer, A. L.; Sleiman, H. F., *Science* **2008,** 321, 1795-1799.

(7) Kuzuya, A.; Komiyama, M., *Nanoscale* **2010,** 2, 310-322.

(8) Shih, W. M.; Lin, C. X., *Curr Opin Struc Biol* **2010,** 20, 276-282.

(9) Torring, T.; Voigt, N. V.; Nangreave, J.; Yan, H.; Gothelf, K. V., *Chemical Society Reviews* **2011,** 40, 5636-5646.

(10) Chen, J. H.; Seeman, N. C., *Nature* **1991,** 350, 631-633.

(11) Rothemund, P. W. K.; Papadakis, N.; Winfree, E., *Plos Biol* **2004,** 2, 2041-2053.

(12) Douglas, S. M.; Dietz, H.; Liedl, T.; Hogberg, B.; Graf, F.; Shih, W. M., *Nature* **2009,** 459, 1154-1154.

(13) Gu, H.; Chao, J.; Xiao, S.-J.; Seeman, N. C., *Nat Nano* **2009,** 4, 245-248.

(14) Gu, H.; Chao, J.; Xiao, S.-J.; Seeman, N. C., *Nature* **2010,** 465, 202-205.

(15) Mao, C. D.; Sun, W. Q.; Shen, Z. Y.; Seeman, N. C., *Nature* **1999,** 397, 144-146.

(16) Maye, M. M.; Kumara, M. T.; Nykypanchuk, D.; Sherman, W. B.; Gang, O., *Nat Nanotechnol* **2010,** 5, 116-120.

(17) Gu, H. Z.; Yang, W.; Seeman, N. C., *Journal of the American Chemical Society* **2010,** 132, 4352-4357.

(18) Li, J. W. J.; Tan, W. H., *Nano Lett* **2002,** 2, 315-318.

(19) Shen, W. Q.; Bruist, M. F.; Goodman, S. D.; Seeman, N. C., *Angew Chem Int Edit* **2004,** 43, 4750-4752.

(20) Sherman, W. B.; Seeman, N. C., *Nano Lett* **2004,** 4, 1801-1801.

(21) Yan, H.; Zhang, X. P.; Shen, Z. Y.; Seeman, N. C., *Nature* **2002,** 415, 62-65.

(22) Seeman, N. C., *Trends Biochem Sci* **2005,** 30, 119-125.

(23) Lubrich, D.; Lin, J.; Yan, J., *Angew Chem Int Edit* **2008,** 47, 7026-7028.

(24) Yurke, B.; Turberfield, A. J.; Mills, A. P.; Simmel, F. C.; Neumann, J. L., *Nature* **2000,** 406, 605-608.

(25) Zhang, D. Y.; Winfree, E., *Journal of the American Chemical Society* **2009,** 131, 17303-17314.

(26) Li, Q. Q.; Luan, G. Y.; Guo, Q. P.; Liang, J. X., *Nucleic Acids Res* **2002,** 30.

(27) Simmel, F. C.; Yurke, B., *Phys Rev E* **2001,** 63.

(28) Zhang, D. Y.; Seelig, G., *Nat Chem* **2011,** 3, 103-113.

(29) Yan, H.; Chhabra, R.; Sharma, J.; Liu, Y., *Nano Lett* **2006,** 6, 978-983.

(30) Ding, B.; Seeman, N. C., *Science* **2006,** 314, 1583-1585.

(31) Feng, L. P.; Park, S. H.; Reif, J. H.; Yan, H., *Angew Chem Int Edit* **2003,** 42, 4342-4346.

(32) Goodman, R. P.; Heilemann, M.; Doose, S.; Erben, C. M.; Kapanidis, A. N.; Turberfield, A. J., *Nat Nanotechnol* **2008,** 3, 93-96.

(33) Gothelf, K. V.; Andersen, E. S.; Dong, M.; Nielsen, M. M.; Jahn, K.; Subramani, R.; Mamdouh, W.; Golas, M. M.; Sander, B.; Stark, H.; Oliveira, C. L. P.; Pedersen, J. S.; Birkedal, V.; Besenbacher, F.; Kjems, J., *Nature* **2009,** 459, 73-U75.

(34) Han, D. R.; Pal, S.; Liu, Y.; Yan, H., *Nat Nanotechnol* **2010,** 5, 712-717.

(35) Patitz, M. J.; Summers, S. M., *Nat Comput* **2010,** 9, 135-172.

Chapter 5 Reference

(1) Whitesides, G. M.; Grzybowski, B., Science 2002, 295, 2418-2421.

(2) Hyman, A. A.; Salser, S.; Drechsel, D. N.; Unwin, N.; Mitchison, T. J., Mol Biol Cell 1992, 3, 1155-1167.

(3) Desai, A.; Mitchison, T. J., Annu Rev Cell Dev Bi 1997, 13, 83-117.

(4) Dirks, R. M.; Pierce, N. A., P Natl Acad Sci USA 2004, 101, 15275-15278.

(5) Zhang, D. Y.; Hariadi, R. F.; Choi, H. M. T.; Winfree, E., Nat Commun 2013, 4, 1965.

(6) Zhang, D. Y.; Chen, S. X.; Yin, P., Nat Chem 2012, 4, 208-214.

(7) Shechtman, D.; Blech, I.; Gratias, D.; Cahn, J., Physical Review Letters 1984, 53, 1951-1953.

(8) Steurer, W., Chemical Society Reviews 2012, 41, 6717.

(9) Elser, V. V., Phys Rev Lett 1985, 54, 1730.

(10) Steurer, W., Chemical Society Reviews 2012, 41, 6719.

(11) de Boissieu, M., Chemical Society Reviews 2012, 41, 6778.

(12) Zheng, J. P.; Birktoft, J. J.; Chen, Y.; Wang, T.; Sha, R. J.; Constantinou, P. E.; Ginell, S. L.; Mao, C. D.; Seeman, N. C., Nature 2009, 461, 74-77.

(13) Yang, H.; Sleiman, H. F., Angew Chem Int Edit 2008, 47, 2443-2446.

(14) Ye, J. D.; Tereshko, V.; Frederiksen, J. K.; Koide, A.; Fellouse, F. A.; Sidhu, S. S.; Koide, S.; Kossiakoff, A. A.; Piccirilli, J. A., P Natl Acad Sci USA 2008, 105, 82-87.

(15) Chapman, H. N.; etal., Nature 2011, 470, 73-77.

(16) Rothemund, P. W. K., Nature 2006, 440, 297-302.

(17) Han, D. R.; Pal, S.; Yang, Y.; Jiang, S. X.; Nangreave, J.; Liu, Y.; Yan, H., Science 2013, 339, 1412-1415.

(18) Zhang, H. L.; Chao, J.; Pan, D.; Liu, H. J.; Huang, Q.; Fan, C. H., Chem Commun 2012, 48, 6405-6407.

(19) Hogberg, B.; Liedl, T.; Shih, W. M., J Am Chem Soc 2009, 131, 9154-9155.

(20) Zhao, Z.; Liu, Y.; Yan, H., Nano Lett 2011, 11, 2997-3002.

(21) Ducani, C.; Kaul, C.; Moche, M.; Shih, W. M.; Högberg, B., Nat Methods 2013, 10, 647-652.

(22) Jin, Z.; Sun, W.; Ke, Y.; Shih, C. J.; Paulus, G. L. C.; Wang, Q. H.; Mu, B.; Yin, P.; Strano, M. S., Nat Commun 2013, 4, 1663.

(23) Hung, A. M.; Cha, J. N., 2011, 749, 187-197.

(24) Kershner, R. J.; Bozano, L. D.; Micheel, C. M.; Hung, A. M.; Fornof, A. R.; Cha, J. N.; Rettner, C. T.; Bersani, M.; Frommer, J.; Rothemund, P. W. K.; Wallraff, G. M., Nat Nanotechnol 2009, 4, 557-561.

(25) Ding, B.; Wu, H.; Xu, W.; Zhao, Z.; Liu, Y.; Yu, H.; Yan, H., Nano Lett 2010, 10, 5065-5069.

(26) Sun, X. P.; Ko, S. H.; Zhang, C. A.; Ribbe, A. E.; Mao, C. D., J Am Chem Soc 2009, 131, 13248-13249.

(27) Woo, S.; Rothemund, P. W. K. In Self-assembly of two-dimensional DNA origami lattices using cation-controlled surface diffusion, Foundations of Nanoscience: Self-Assembled Architectures and Devices., Snowbird, Utah, Snowbird, Utah, 2014; p 93.

(28) Endo, M.; Suzuki, Y.; Yang, Y.; Sugiyama, H. In Photoresponsive DNA nanostructures, single-molecule imaging and controlled assembly, Foundations of Nanoscience: Self-Assembled Architectures and Devices., Snowbird, Utah, Snowbird, Utah, 2014; p 7.

(29) Rothemund, P. W. K.; Papadakis, N.; Winfree, E., Plos Biol 2004, 2, 2041-2053.

(30) Barish, R. D.; Rothemund, P. W. K.; Winfree, E., Nano Lett 2005, 5, 2586-2592.

(31) Barish, R. D.; Schulman, R.; Rothemund, P. W. K.; Winfree, E., P Natl Acad Sci USA 2009, 106, 6054-6059.

(32) Mao, C. D.; Sun, W. Q.; Seeman, N. C., Nature 1997, 386, 137-138.

(33) Han, D. R.; Pal, S.; Liu, Y.; Yan, H., Nat Nanotechnol 2010, 5, 712-717.

(34) Elbaz, J.; Cecconello, A.; Fan, Z. Y.; Govorov, A. O.; Willner, I., Nat Commun 2013, 4.

(35) Liu, D.; Chen, G.; Weizmann, Y. In Creating Complex molecular Topologies by Configuring DNA Four-Way Junctions Foundations of Nanoscience: Self-Assembled Architectures and Devices., Snowbird, Utah, Snowbird, Utah, 2014; p 34.

(36) Wang, T.; Sha, R. J.; Dreyfus, R.; Leunissen, M. E.; Maass, C.; Pine, D. J.; Chaikin, P. M.; Seeman, N. C., Nature 2011, 478, 225-228.

(37) Schulman, R.; Yurke, B.; Winfree, E., P Natl Acad Sci USA 2012, 109, 6405-6410.

(38) Yin, P.; Choi, H. M. T.; Calvert, C. R.; Pierce, N. A., Nature 2008, 451, 318-322.

(39) Bairoch, A., Nucleic Acids Res 2000, 28, 304-305.

(40) Hammes, G. G.; Wu, C. W., Science 1971, 172, 1205-1211.

(41) Drews, J., Science 2000, 287, 1960-1964.

(42) Khosla, C.; Harbury, P. B., Nature 2001, 409, 247-252.

(43) Ostermeier, M., Curr Opin Struc Biol 2009, 19, 442-448.

(44) Fu, Y.; Zeng, D.; Chao, J.; Jin, Y.; Zhang, Z.; Liu, H.; Li, D.; Ma, H.; Huang, Q.; Gothelf, K. V.; Fan, C., J Am Chem Soc 2013, 135, 696-702.

(45) Wilner, O. I.; Weizmann, Y.; Gill, R.; Lioubashevski, O.; Freeman, R.; Willner, I., Nat Nanotechnol 2009, 4, 249-254.

(46) Saghatelian, A.; Guckian, K. M.; Thayer, D. A.; Ghadiri, M. R., J Am Chem Soc 2003, 125, 344-345.

(47) Simon, P.; Dueymes, C.; Fontecave, M.; Decout, J. L., Angew Chem Int Edit 2005, 44, 2764-2767.

(48) Fu, J. L.; Liu, M. H.; Liu, Y.; Woodbury, N. W.; Yan, H., J Am Chem Soc 2012, 134, 5516-5519.

(49) Fu, Y. M.; Zeng, D. D.; Chao, J.; Jin, Y. Q.; Zhang, Z.; Liu, H. J.; Li, D.; Ma, H. W.; Huang, Q.; Gothelf, K. V.; Fan, C. H., J Am Chem Soc 2013, 135, 696-702.

(50) Fu, J.; Yang, Y.; Buck, A. J.; Liu, M.; Liu, Y.; Walter, N. G.; Woodbury, N. W.; Yan, H., Nat Nanotechnol 2014, 9, 531-536.

(51) Tseng, C. Y.; Zocchi, G., J Am Chem Soc 2013, 135, 11879-11886.

(52) Liu, M. H.; Fu, J. L.; Hejesen, C.; Yang, Y. H.; Woodbury, N. W.; Gothelf, K.; Liu, Y.; Yan, H., Nat Commun 2013, 4, 2127.

(53) Lammers, T.; Aime, S.; Hennink, W. E.; Storm, G.; Kiessling, F., Accounts Chem Res 2011, 44, 1029-1038.

(54) Langecker, M.; Arnaut, V.; Martin, T. G.; List, J.; Renner, S.; Mayer, M.; Dietz, H.; Simmel, F. C., Science 2012, 338, 932-936.

(55) Andersen, E. S.; Dong, M.; Nielsen, M. M.; Jahn, K.; Subramani, R.; Mamdouh, W.; Golas, M. M.; Sander, B.; Stark, H.; Oliveira, C. L. P.; Pedersen, J. S.; Birkedal, V.; Besenbacher, F.; Gothelf, K. V.; Kjems, J., Nature 2009, 459, 73-75.

(56) Douglas, S. M.; Bachelet, I.; Church, G. M., Science 2012, 335, 831-834.

(57) Rudchenko, M.; Taylor, S.; Pallavi, P.; Dechkovskaia, A.; Khan, S.; Butler, V. P.; Rudchenko, S.; Stojanovic, M. N., Nat Nanotechnol 2013, 8, 580-586.

(58) Hemphill, J.; Deiters, A., J Am Chem Soc 2013, 135, 10512-10518.

(59) Yin, P. In Synthetic organization and regulation with DNA/RNA bricks, Foundations of Nanoscience: Self-Assembled Architectures and Devices., Snowbird, Utah, Snowbird, Utah, 2014; p 20.

(60) Chen, Y. J.; Dalchau, N.; Srinivas, N.; Phillips, A.; Cardelli, L.; Soloveichik, D.; Seelig, G., Nat Nanotechnol 2013, 8, 755-762.

(61) Mei, Q. A.; Wei, X. X.; Su, F. Y.; Liu, Y.; Youngbull, C.; Johnson, R.; Lindsay, S.; Yan, H.; Meldrum, D., Nano Lett 2011, 11, 1477-1482.

(62) Castro, C. E.; Kilchherr, F.; Kim, D. N.; Shiao, E. L.; Wauer, T.; Wortmann, P.; Bathe, M.; Dietz, H., Nat Methods 2011, 8, 221-229.

(63) Modi, S.; Nizak, C.; Surana, S.; Halder, S.; Krishnan, Y., Nat Nanotechnol 2013, 8, 459-467.

(64) Perrault, S. D.; Shih, W. M., Acs Nano 2014, 140423081328001.

(65) Ke, Y. G.; Bellot, G.; Voigt, N. V.; Fradkov, E.; Shih, W. M., Chem Sci 2012, 3, 2587-2597.

(66) Sobczak, J. P. J.; Martin, T. G.; Gerling, T.; Dietz, H., Science 2012, 338, 1458-1461.

(67) Lin, C. X.; Perrault, S. D.; Kwak, M.; Graf, F.; Shih, W. M., Nucleic Acids Res 2013, 41, e40.

(68) Martin, T. G.; Dietz, H., Nat Commun 2012, 3, 1103.

(69) Myhrvold, C.; Dai, M. J.; Silver, P. A.; Yin, P., Nano Lett 2013, 13, 4242-4248.

(70) Guo, P. X., Nat Nanotechnol 2010, 5, 833-842.

(71) Shukla, G. C.; Haque, F.; Tor, Y.; Wilhelmsson, L. M.; Toulmé, J.-J.; Isambert, H.; Guo, P.; Rossi, J. J.; Tenenbaum, S. A.; Shapiro, B. A., Acs Nano 2011, 5, 3405-3418.

(72) Lai, Y. T.; Cascio, D.; Yeates, T. O., Science 2012, 336, 1129-1129.

(73) King, N. P.; Sheffler, W.; Sawaya, M. R.; Vollmar, B. S.; Sumida, J. P.; Andre, I.; Gonen, T.; Yeates, T. O.; Baker, D., Science 2012, 336, 1171-1174.

(74) Li, X. M.; Mooney, P.; Zheng, S.; Booth, C. R.; Braunfeld, M. B.; Gubbens, S.; Agard, D. A.; Cheng, Y. F., Nat Methods 2013, 10, 584-590.

(75) Liao, M.; Cao, E.; Julius, D.; Cheng, Y., Nature 2013, 504, 107-112.

(76) Scheres, S. In How self-assembly could exploit high-resolution cryo-TEM and vice versa, Foundations of Nanoscience: Self-Assembled Architectures and Devices., Snowbird, Utah, Snowbird, Utah, 2014; p 55.

APPENDIX A

SUPPLEMENTARY INFORMATION FOR CHAPTER 2

COMPLEX ARCHIMEDEAN TILING SELF-ASSEMLED FROM DNA NANOSTRUCTURES

Experimental Methods

Materials. All the strands were purchased from Integrated DNA Technologies Inc. (www.IDTDNA.com) at a 25 or 100 nmole synthesis scale, and were further purified by denaturing PAGE gel electrophoresis.

Nanostructure assembly. The designs and sequences of the DNA oligos used to form each of the structures are shown in a later section of the supporting information. A one-pot annealing reaction was used to form each pattern. The strands for all the building blocks in each design are mixed at the designed ratio with a final concentration of the unit cell of 0.6 □M in 1xTAE-Mg^{2+} buffer (20 μM Tris-acetate buffer, pH 7.6, 2 mM EDTA, 12.5 mM $MgCl_2$). The oligonucleotide mixture was annealed in a thermocycler (Eppendorf) that was programmed to cool from 95°C to 4°C over 12 hours: 94°C to 86°C at 4°C per 5 minutes; 85°C to 70°C at 1 °C per 5 minutes; 70°C to 40°C at 1°C per 15 minutes; 40°C to 25°C at 1°C per 10 minutes; then hold at 4°C. The 2-hour annealing program used in Figures S4 and S8 is from 95°C to 4°C over 2 hours: 94°C to 76°C at 2°C per 5 minutes; 76°C to 24°C at 4°C per 5 minutes; then hold at 4°C. The 48-hour annealing program used in Figure S2.8 is from 95°C to 4°C over 48 hours: 95°C for 5 minutes; 90°C for 10 minutes; 86°C to 76°C at 1 °C per 10 minutes; 76°C to 50°C at 1°C per 30 minutes; 50°C to 15°C at 1°C per 60 minutes; then hold at 4°C.

Ratio of building blocks (corresponding to the designs in Figures 1&2 in the main text).

(a) Cairo pentagonal tiling corresponding to Archimedean tiling (3^2.4.3.4).

Red four-arm: yellow four-arm: three-arm=1:1:4

(b) Prismatic pentagonal tiling corresponding to Archimedean tiling ($3^3.4^2$), (c) shortened Prismatic pentagonal tiling and (d) shortened Prismatic pentagonal tiling with two faces.

Four-arm: three-arm=1:2

(e) 2-uniform tiling.

Four-arm: three-arm=1:1

AFM imaging. 2 □L samples were deposited onto freshly pealed mica (Ted Pella, Inc.) and left for 2 minutes for adsorption to the mica surface. 80 □L of 1x TAE-Mg^{2+} buffer was added to the samples and an extra 40 □L of the same buffer were deposited on the AFM tip. The samples were scanned in "ScanAssyst mode in fluid" using an AFM (Dimension FastScan, Bruker Corporation) with SCANASSYST-FLUID+ tips (Bruker, Inc.).

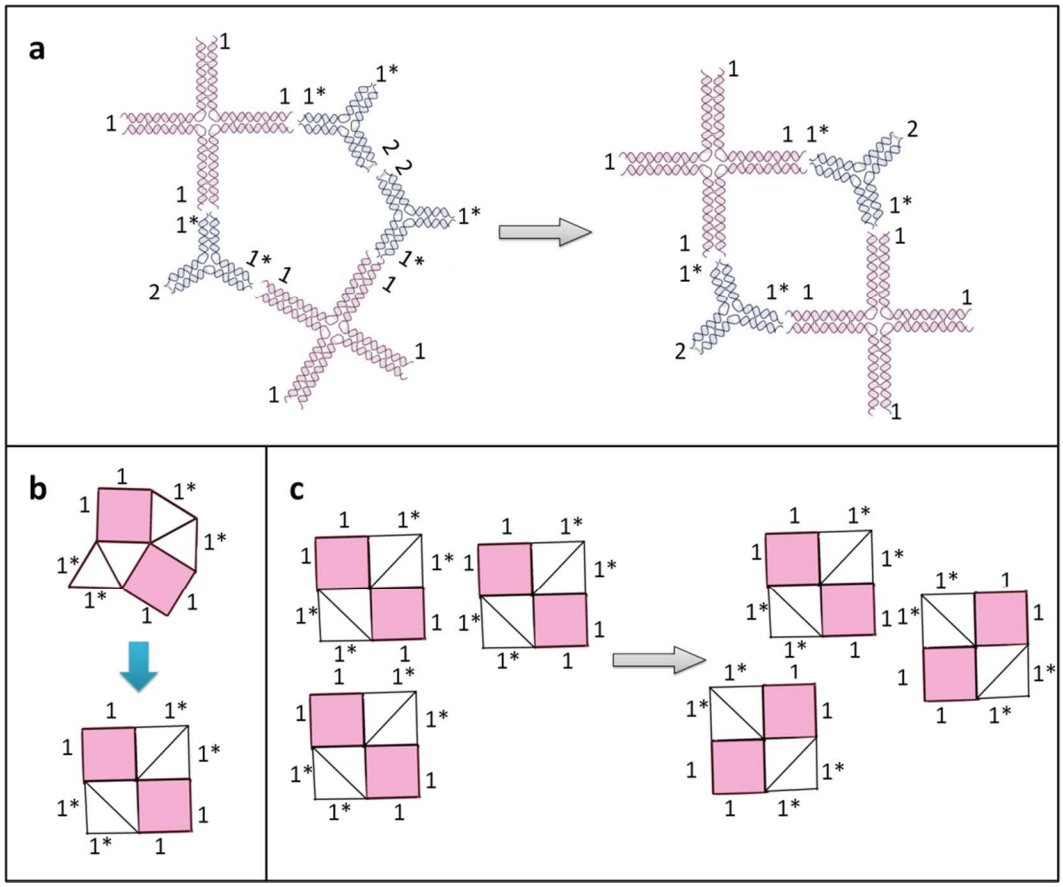


Figure S2.1. Possible mismatches in the case when over-simplified building blocks are used for the Cairo pentagonal tiling. Here the unit cell of the desired pattern includes cyclization of two four-arm junctions and three three-arm junctions in a 3−3− 4−3−4

pattern to form an asymmetric pentagon. The symmetry of the sticky-end sequences in the four-arm tile may promote the exclusion of one of the three-arm motifs, causing the remaining four tiles to connect in a 3−4−3−4 pattern to form a square (or rhombus) unit. Deformation of the junction angles is possible because the arms of these four-arm and three-arm DNA junction motifs are relatively long and less rigid than shorter ones and the angles at the branch points of each individual tile may be flexible enough to deviate from the expected values of 90° and 120°. It is kinetically more favorable to form a four-membered ring than a five-membered ring, but the four-membered unit cells do not grow into large 2D arrays because of excess structural strain. Even when the correct numbers of junction motifs combine to form the expected unit cells, they cannot assemble with perfect edge-to-edge tiling in the presence of incorrectly formed four-membered rings. **a,** The correct assembly intermediate with a 5-member ring assembled from a 3-3-4-3-4 building block pattern is shown in the left. However, when the two 4-arm junction tiles are equivalent they may adopt an alternative linkage in which four building blocks instead of five twist and connect to form a 4-member ring. **b,** The unit cell can be topologically treated as a square since the angles at the junction points are flexible. **c,** Possible mismatches between the unit cells by shifting the tiles.

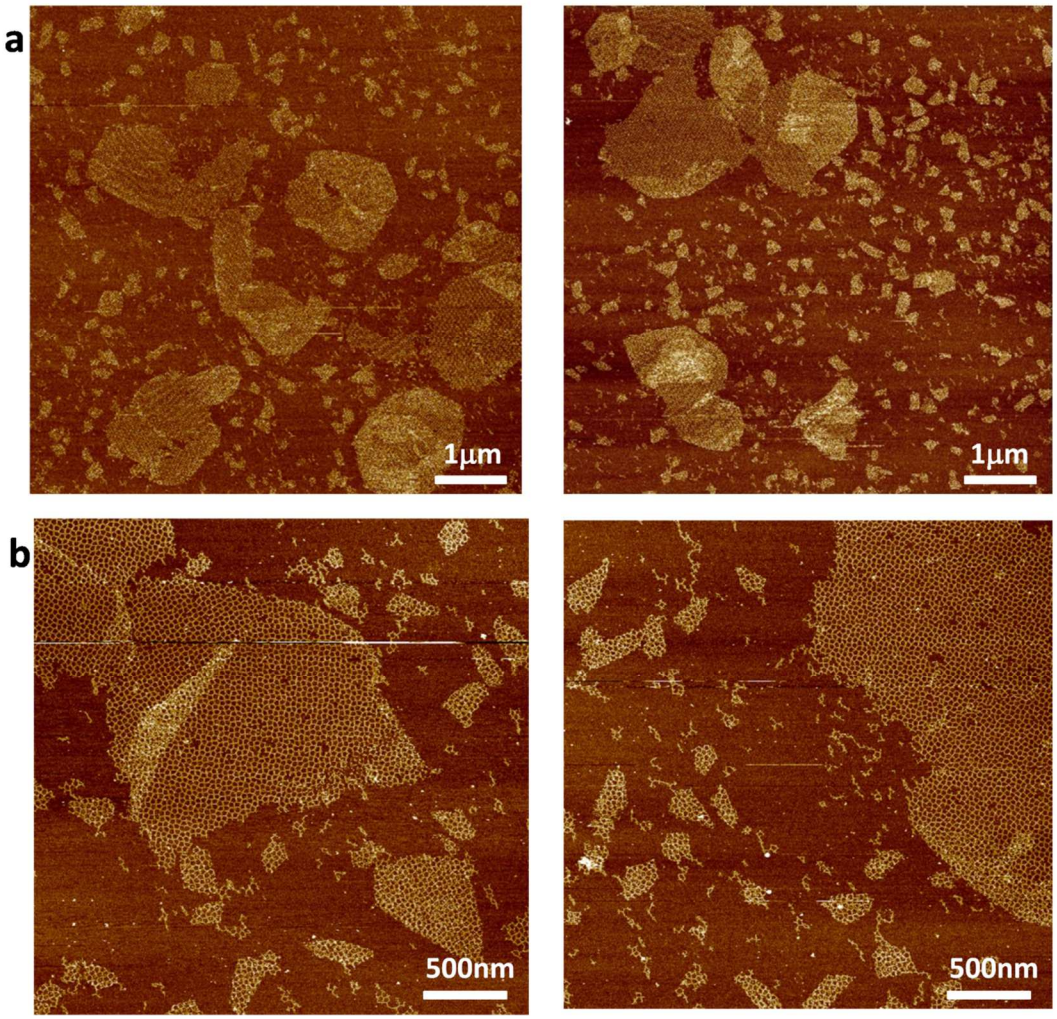


Figure S2.2. Additional AFM images for the Cairo pentagonal tiling. **a,** zoom out images indicate that the large 2D arrays tend to curl up at the edges. **b,** zoom in images show some smaller pockets (2 layers with 100-300 nm dimensions) also form along with the larger 1–layer 2D arrays.

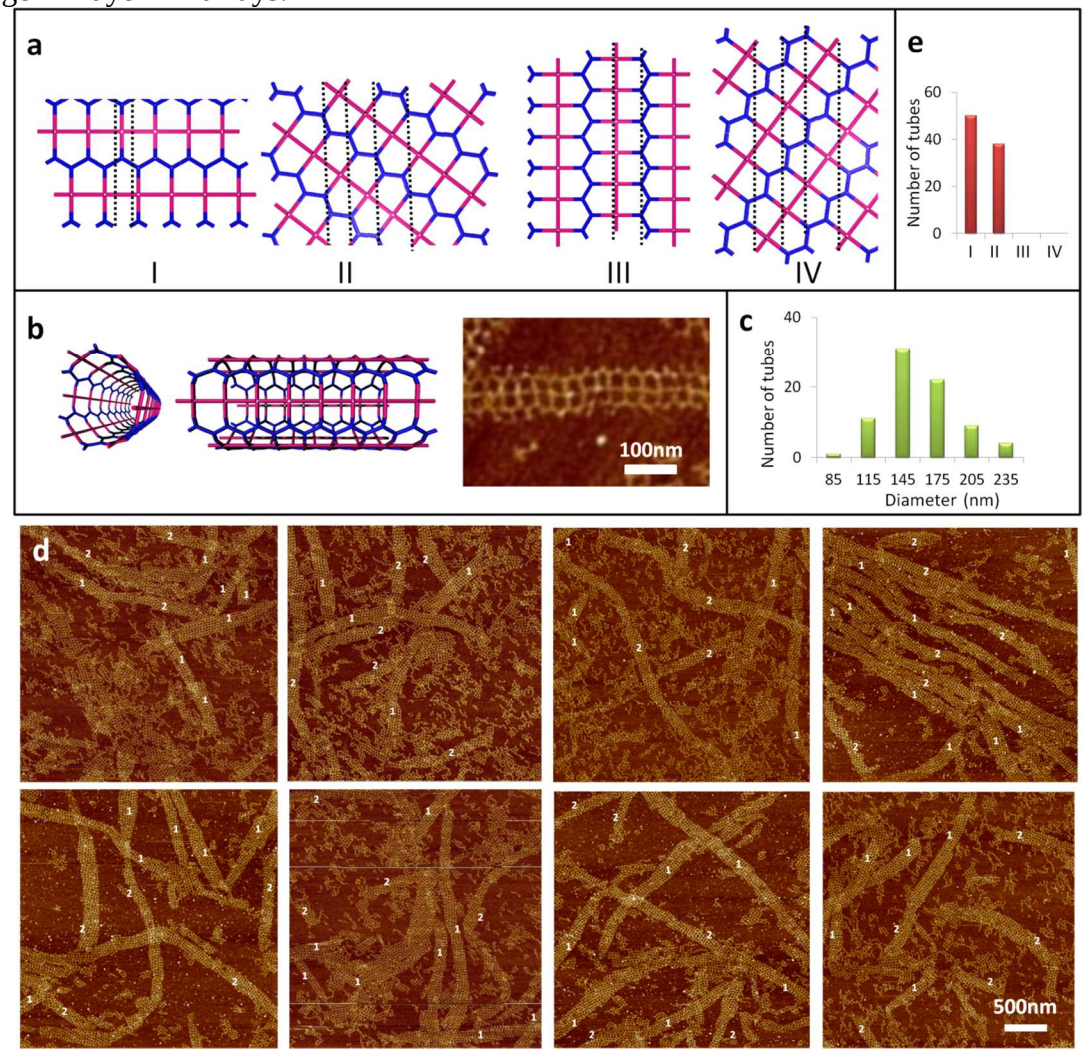


Figure S2.3. Tube folding mechanism for the standard Prismatic pentagonal tiling pattern. **a,** four possible folds I-IV with matching edges indicated by the dashed lines. **b**, 3D view of the tube formed by fold I with a zoom in AFM image of a tube flattened on surface (assuming this folding mechanism). **c,** the measured diameter (or half perimeter) distribution of the tubes from AFM. **d,** the sample AFM images for assigning/counting the type of tube folds. The numbers (1 or 2) marked on the tubes indicate the assigned fold type I or II from the AFM images. **e.** The distribution of different tube folds shows that all of the tubes assume either fold I or II, but none assumes fold III or IV.

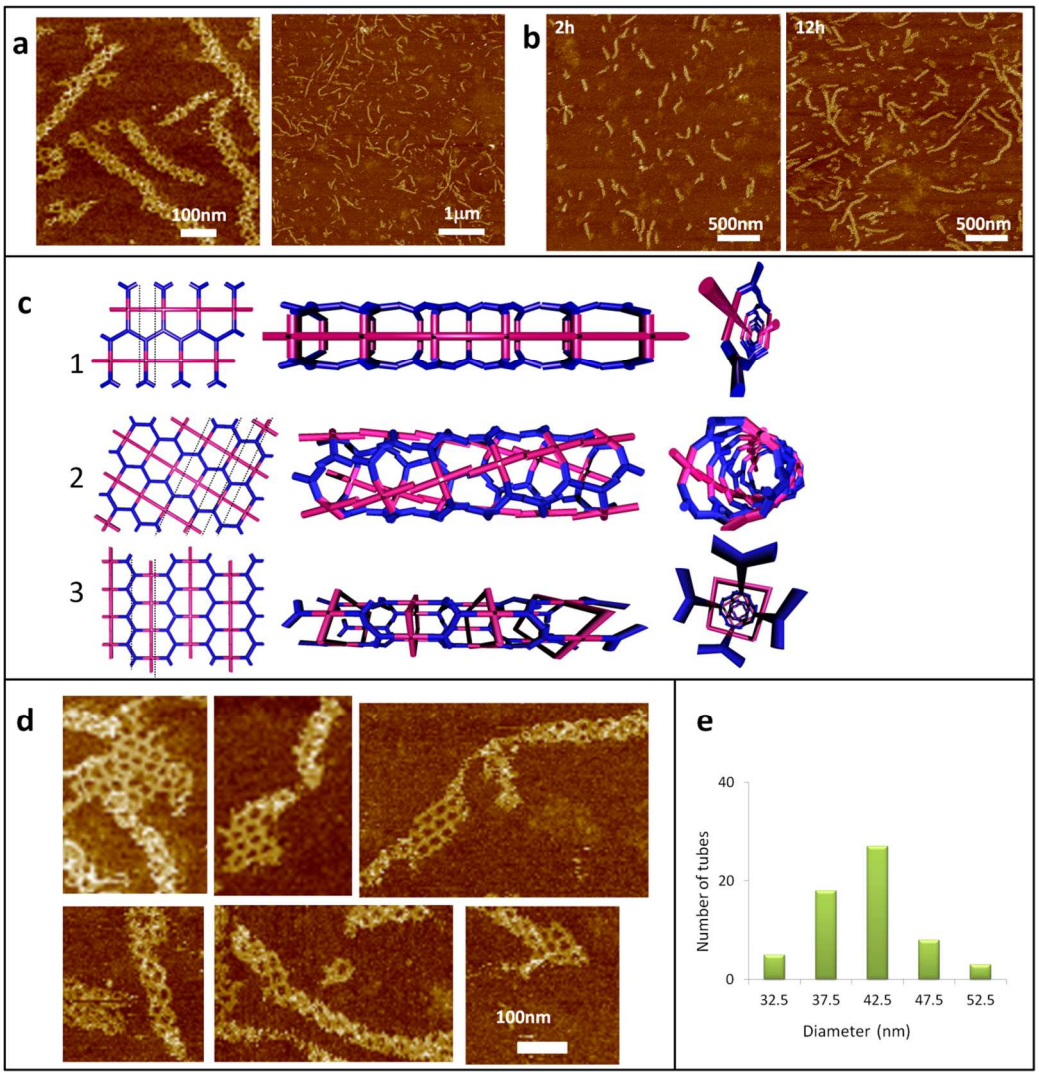


Figure S2.4. Tube folding mechanism for the shortened Prismatic pentagonal tiling pattern. **a,** additional zoom in and zoom out AFM images for the standard annealing program (12 h). **b,** AFM images for different annealing times, 2 h and 12 h, respectively. Scale bars are marked in each image. **c,** Schematics for the three possible fold types. **d,** The first row shows the AFM images of tubes assuming fold 1 and the second row shows the AFM images of tubes assuming fold 2. It is noted that none of the tube are observed to assume the fold 3. Scale bars are the same in all images, 100 nm. The tubes can be easily opened either by deposition on the mica surface or broken by the scanning AFM tip. **e,** The diameter (or half perimeter) distribution of the tubes measured from AFM. The tubes are remarkably more narrow in this design compared to the one shown in Figure S2.3.

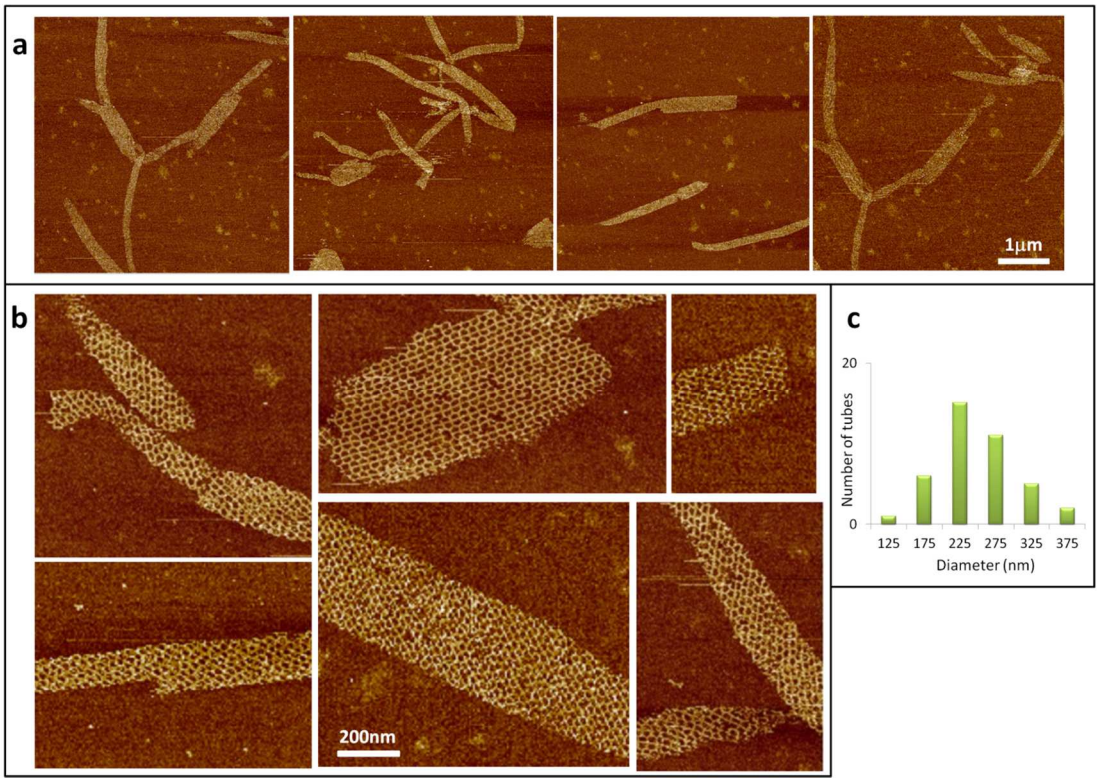


Figure S2.5. **a,b,** Additional zoom out and zoom in AFM images for the shortened Prismatic pentagonal tiling with a corrugated design. **c,** the diameter (or half perimeter) distribution of the tubes measured from AFM. The tubes are wider and have a broader distribution in this design compared to those shown in Figure S2.3 and S4.

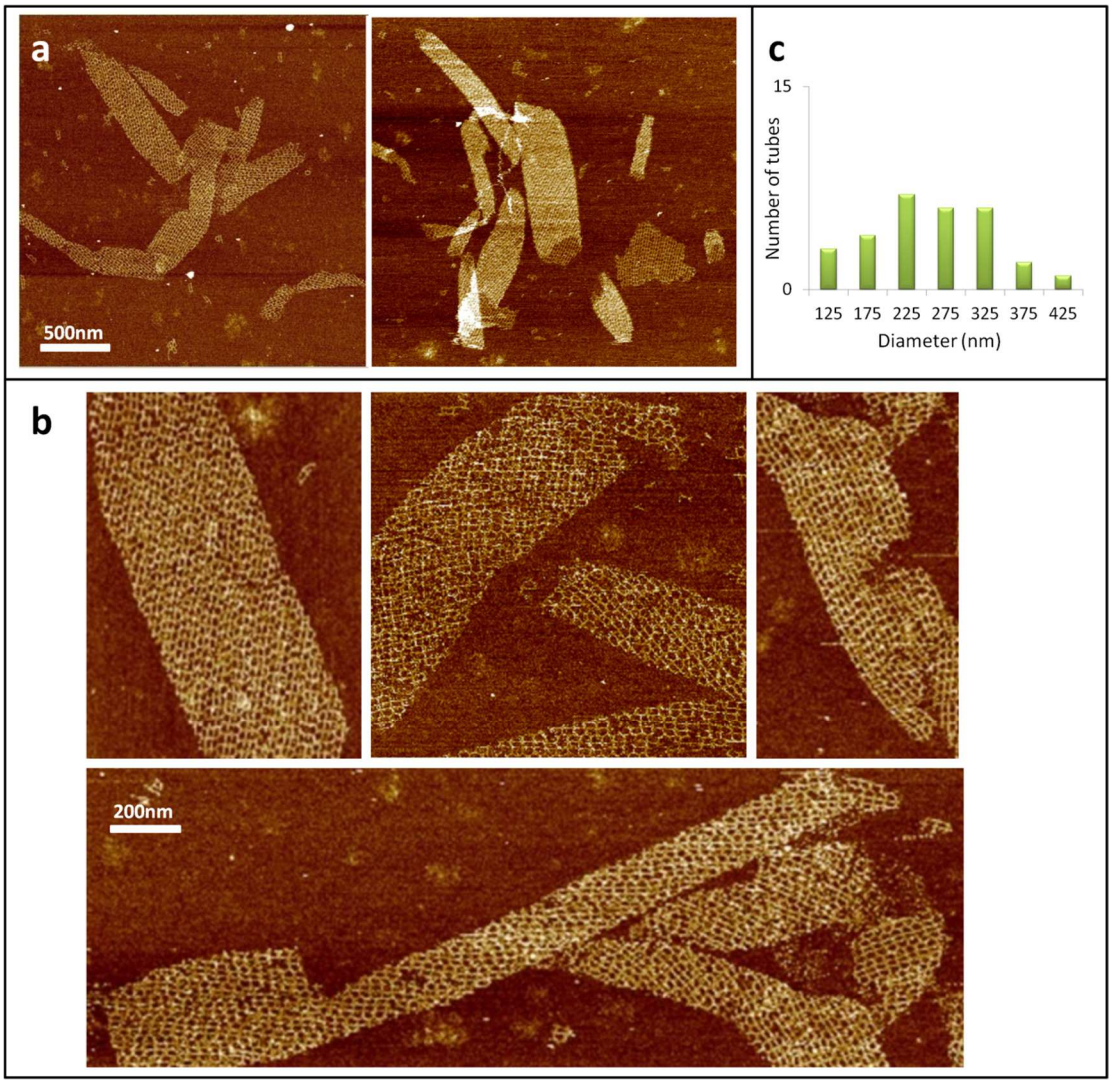


Figure S2.6. **a,b,** Additional zoom out and zoom in AFM images of the 2-uniform tiling with a corrugated design. Both rectangular and pentagonal cavities are observed. Tubes with fold types 1 and 2 are both present. Pieces of 2D array co-exist with tubes. **c,** diameter (or half perimeter) distribution of tubes measured from AFM. The average tube diameters are even larger compared to those shown in Figure S2.5.

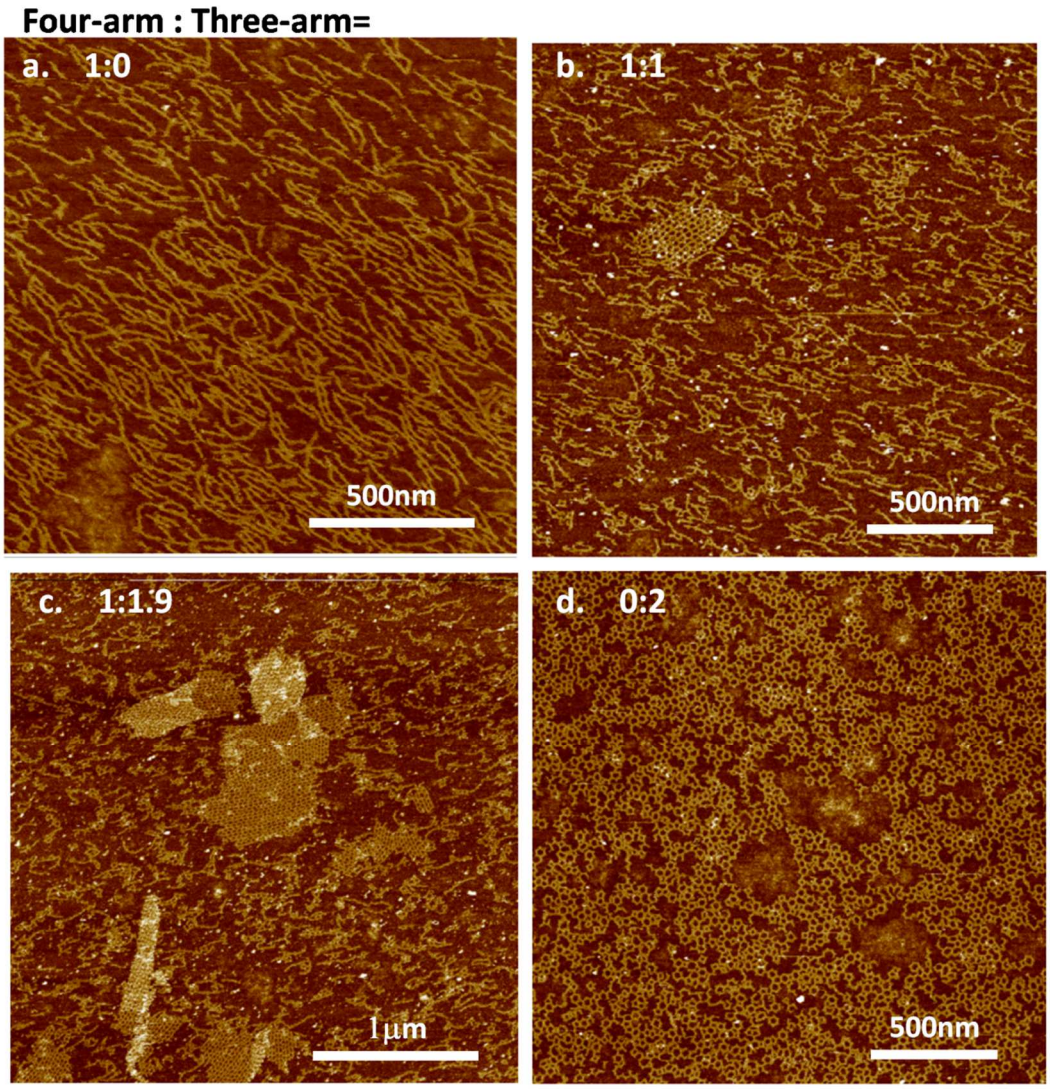


Figure S2.7. For the Prismatic pentagonal tiling pattern with a corrugated design, mixing the two types of building blocks in different ratios (that deviate from the ideal 1:2 ratio) results in the formation of various products. **a.** With 4-arm junction tiles only a linear array is observed. **b.** With a 1:1 ratio, there is a shortage of 3-arm junction tiles and the 2D array does not grow properly. Only very small fragments that displayed the correct pattern were observed. **c.** With a 1:1.9 ratio there is still a shortage of 3-arm junction tiles and the arrays do not grow very large. In addition there are many small fragments in the background. **d.** With 3-arm junction tiles only, 4, 5, 6-member rings and random, higher order cross-linked products were observed.This result indicates that the junction points are flexible and assume a range of different angles from 90-120 degrees.

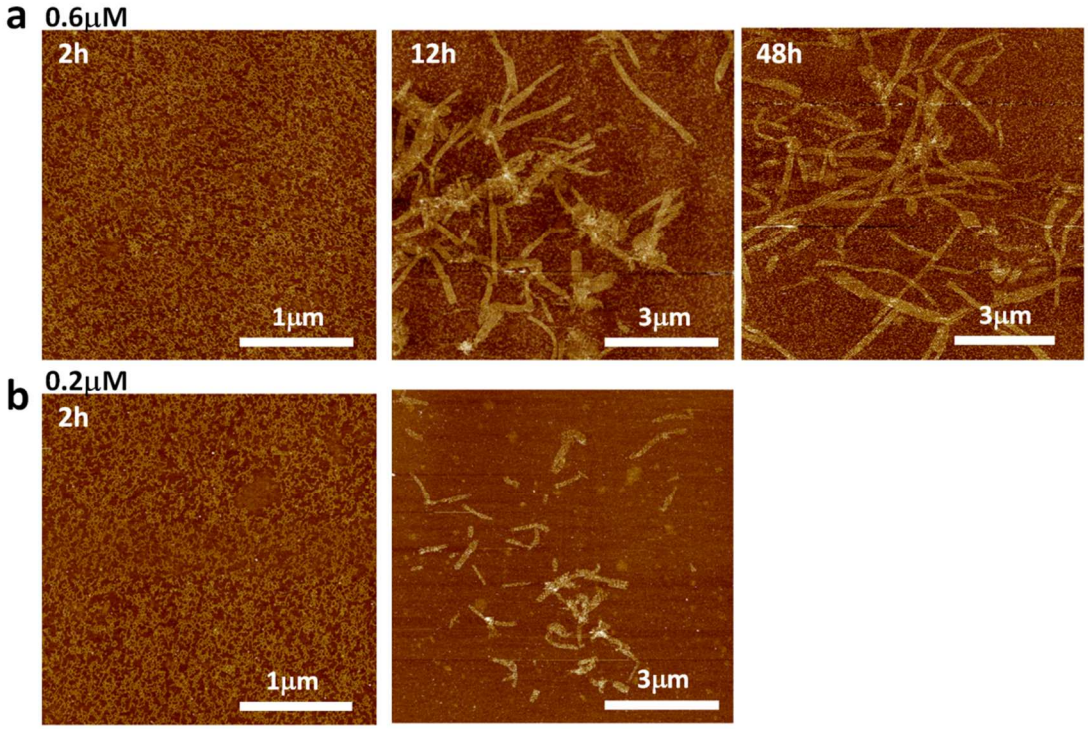


Figure S2.8. Examining how concentration and annealing time influences the assembly of the shortened Prismatic pentagonal tiling with a corrugated design. Two concentrations (relative to the unit cell) were considered: 0.2 □M and 0.6 □M. In both cases a relatively fast annealing program (90ºC to 4ºC over 2 hours) does not allow the tile array to grow as expected. Relatively long annealing programs (90ºC to 4ºC over 48 hours) do not necessary cause a change in the dimensions of the tubes. Lower concentrations result in shorter and narrower tubes, possibly due to slow nucleation and growth kinetics.

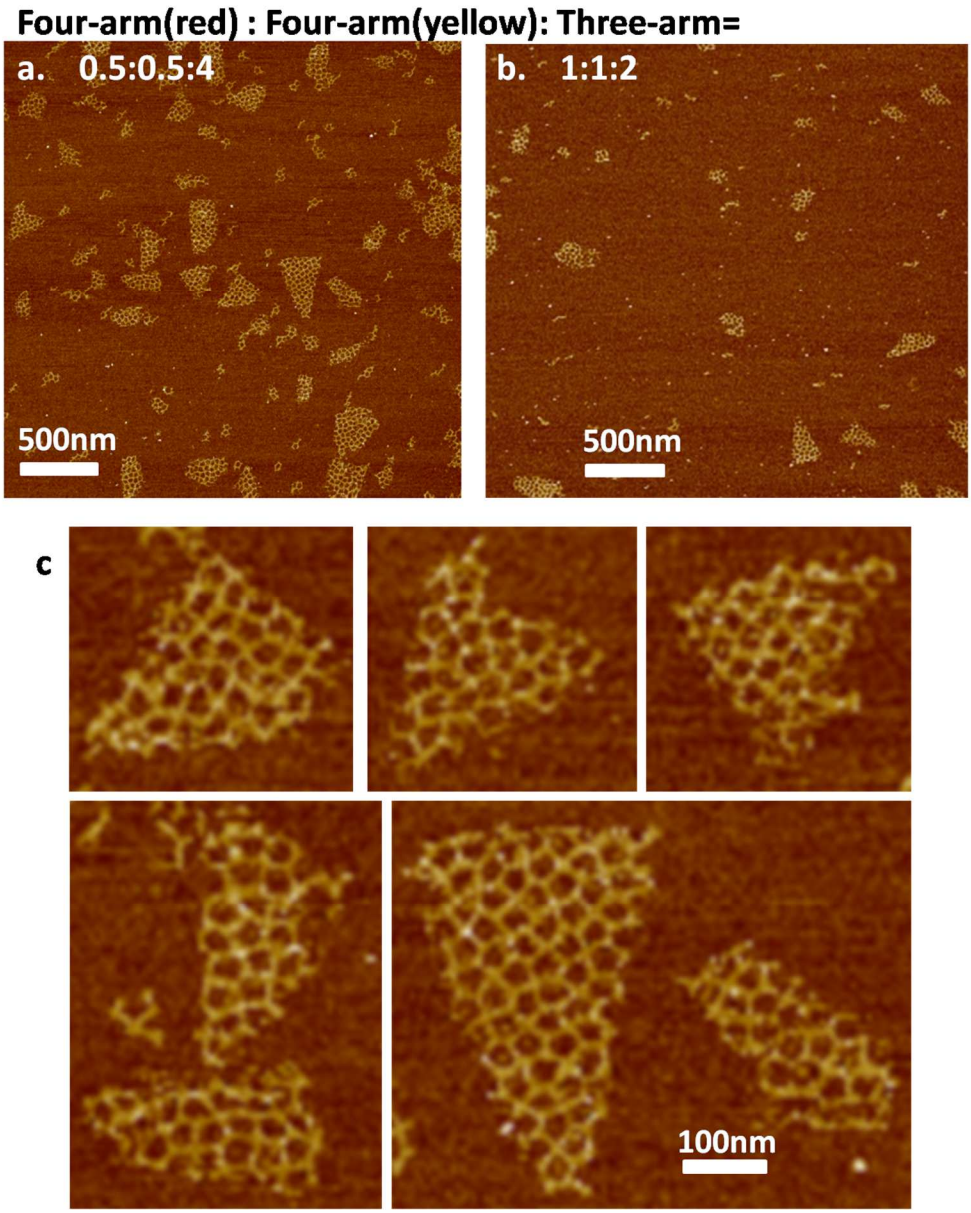


Figure S2.9. Additional AFM images of the formation of small 2-layer pockets for the Cairo pentagonal tiling pattern. **a,b,** two tile ratio conditions that result in formation of 2-layer small pockets with dimensions in the range of 100-300 nm. Both the ratios deviate from the ideal ratio of 1:1:4 for the 2D array. **c,** zoom in AFM images. All images in **c.** have the same scale bar of 100 nm.

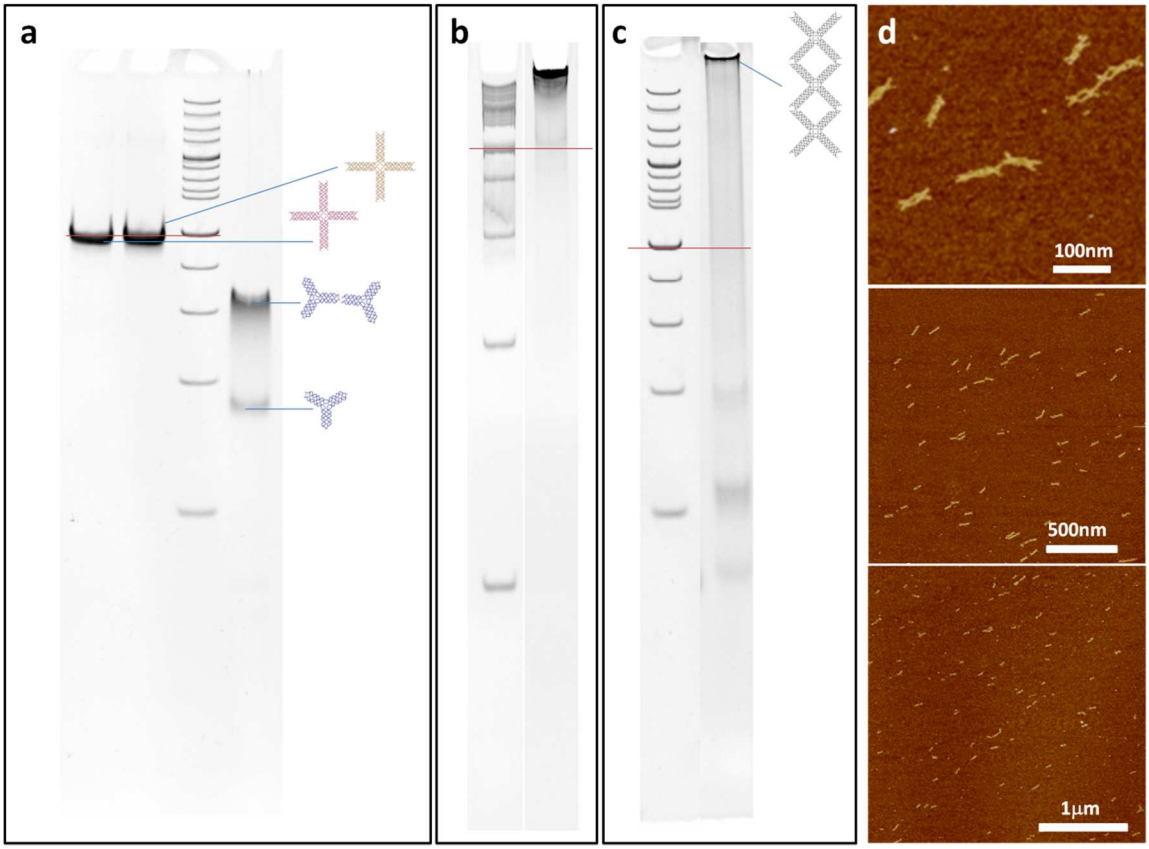
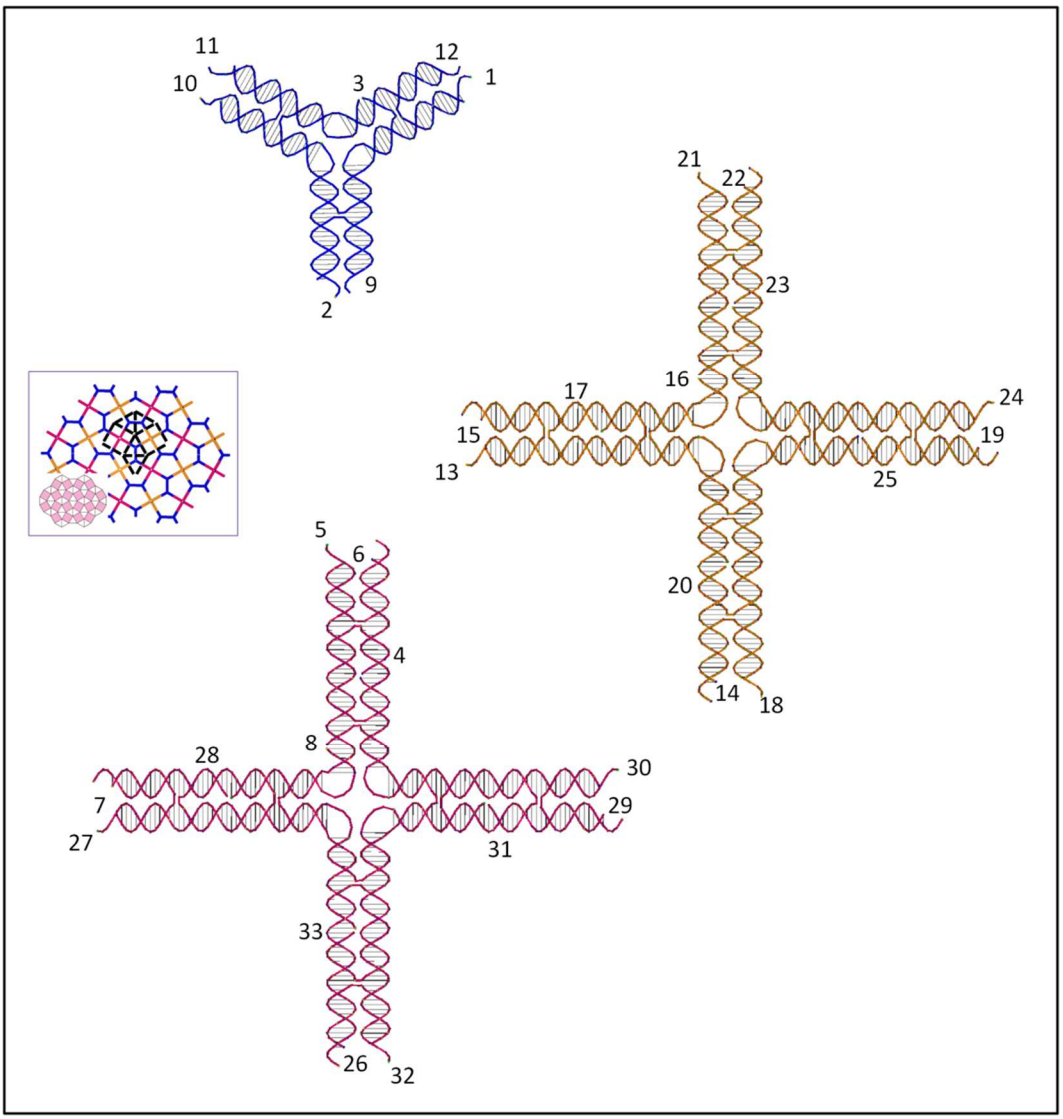


Figure S2.10. we observed that the four-arm junction tiles with four identical sticky-end sequences (GCAG) self-associate in an end-to-end manner to form linear oligomers ranging from dimers to tetramers, with the final assemblies resembling rhombuslike ribbon structures in which each tile exhibits a twisted junction. **a,** Native gel characterization of the individual four-arm and three-arm junction motifs with optimized sequences. The tiles display the sticky ends corresponding to the matching rules in the Cairo pentagonal tiling pattern. **b,** Native gel image of a four-arm junction motif with symmetrical sequence in the arms (the red line indicates the position where the four-arm motifs should migrate). The gel reveals random aggregations that do not run out of the gel wells. **c,** Native gel image of a four-arm junction motif with GCAG sticky end. The tile also self-aggregates. **d,** AFM images of the same sample in **c** which shows the non-specific linear oligomerization of the 4-arm junction tiles.

Sequences of the DNA strands in the Junction tiles used in all of the designs

(numbers are marked in the scheme at the 5' ends)

A. The Cairo Pentagonal tiling design



Three-arm Junction tile

1,CTGCGTTCGTGAGCAAACCTAAGAATGAGATAGCCCGACGCCCCAGGAGG

GTCCTA

2,GAGGTTGACTACGCACTTCCAAAGAACAGACCGGCCATTGCAGGGGTCCTA

C

3,ATGGGTTTTCTTTCTTACTAATGGCCGGTCTTTTGTTCTTTGGGTCGGGCTAT

CTTTTCATTCTTAGCAACC

9,ACCCTCCTGGGGCAAGTGCGTAGTCAA

10,GCATGTAGGACCCCTGCTAGGCCGACTTATA

11,ATGCTATAAGTCGGCCTAAGTAAGAAAGACCCATGGTTGTTACAACTCTGT

ATGAT

12,TACAGAGTTGTAAGTTTGCTCACGAAC

Red four-arm junction tile

8,GGCTTATTTTCTCGTTCCATGATCCGGACTATTTTCGTCCAGACTCGGCAGT

CTGATTTTGCAACCCGCCGACGGACCGGCTTTTACACAGACCGTGGTA

4,AACATCTTGCGTCGGGCGATCAGGGGTGAAGCTGTATAGCTA

28,TACCAGTATGTGGAGAAGACGTAACGCACCATGGGGCTTGCC

31,ACAGGCCCTTTAAGGATTAGCGGCCAACTAGGATTAAGTGAC

33,TTAGCCCTCCGGTCCATGCGTTTTACATTCCGGCGTAGTCAG

6,ACCCGAACGACCAGCGCAAGATGTTTAGCTATACACGGTCTGTGTGCCGGT

CCGTCCCTAGTTGGCCGCTAATCCTTGCGAATGGAGGGA

7,GCCATGCTTCCCGTACATACTGGTAGGCAAGCCCCATGGAACGAGTAAGCC

TACCAGCTTCACCCCTGATCGCCCGAGTTACTCTGTTGC

26,ACTCAGCGTAGAAACGGAGGGCTAACTGACTACGCAGTCTGGACGTAGTC

CGGATCATGGTGCGTTACGTCTTCTCCCGGGCTAATTGTA

29,TAAAGCGACGTCTAAAAGGGCCTGTGTCACTTAATGGCGGGTTGCTCAGA

CTGCCGCGGAATGTAAAACGCATGGACCGCGACCCACGGT

5,CCTCGCAACAGAGTAACCTGGTCGTTCGGGTTAGG

27,CCTCTACAATTAGCCCGACGGGAAGCATGGCTAGG

30,CCTCTCCCTCCATTCGCTAGACGTCGCTTTATAGG

32,CCTCACCGTGGGTCGCGTTTCTACGCTGAGTTAGG

Yellow four-arm junction tile

16,GGCTCGTTTTCGTTGAGGGAGCCACCTGCCCTTTTAAAACCAGCCCCTCCA

GGGTATTTTGTCGAGCTCGGGGCGCCAGCCTTTTACCCCTGCCACTCAC

17,GCGGTGCGACGGCCTGGCTGTGAGAGCAGGTGTGCCCGATAC

20,GGCGAGTAATGCAGCTGATCCATCAGGGCGCGTGTCCTTGCC

23,AGAGGTCGAACGCGTTCCGTTTGGCCCGGGGTTGCCGTCCCG

25,AGTAGGTGCCCAAGCGCTAGAAAATCAGGGTAGGTCTCATCA

14,ACGACCTCACCCCACATTACTCGCCGGCAAGGACAGGCTGGTTTTGGGCA

GGTGGCCACCTGCTCTCACAGCCAGGCAGGCCCCTGAAAT

15,AAGCGTAGAGAGAGCGTCGCACCGCGTATCGGGCATCCCTCAACGCGAGC

CGTGAGACCCCGGGCCAAACGGAACGCGGTCCCAGATGAG

19,TCGCCCGAGTCTACGGGCACCTACTTGATGAGACCCGAGCTCGACTACCCT

GGAGGCGCGCCCTGATGGATCAGCTGACAGTGCGCGAAG

22,TCGGACGCACTGCGGTTCGACCTCTCGGGACGGCATGGCAGGGGTGGCTG

GCGCCCTACCCTGATTTTCTAGCGCTTCGCCACATTTTGT

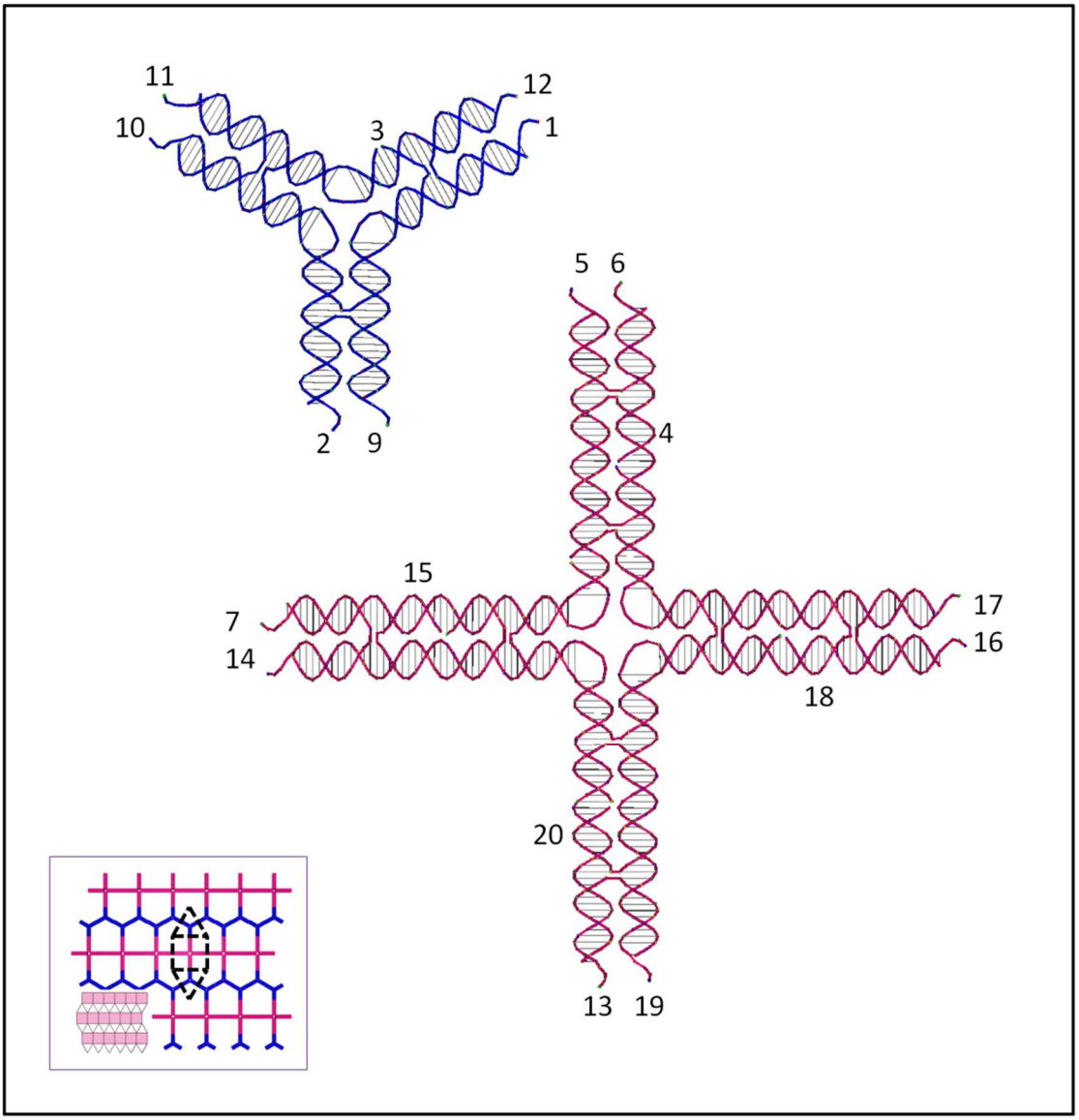
13,GCAGATTTCAGGGGCCTCTCTCTCTACGCTTATCA

18,GCAGCTTCGCGCACTGTTGGGGTGAGGTCGTATCA

21,GCAGCTCATCTGGGACCCGCAGTGCGTCCGAATCA

24,GCAGACAAAATGTGGCGGTAGACTCGGGCGAATCA

B. The standard Prismatic pentagonal tiling design



Three-arm junctione tile

1,TCGCGTTCGTGAGCAAACCTAAGAATGAGATAGCCCGACGCCCCAGGAGG

GT

2,ATCTTTGACTACGCACTTCCAAAGAACAGACCGGCCATTGCAGGGGTCCTA

C

11,CAGTTATAAGTCGGCCTAAGTAAGAAAGACCCATGGTTGTTACAACTCTGT

A

3,ATGGGTTTTCTTTCTTACTAATGGCCGGTCTTTTGTTCTTTGGGTCGGGCTAT

CTTTTCATTCTTAGCAACC

9,CACGACCCTCCTGGGGCAAGTGCGTAGTCAA

10,ACTGGTAGGACCCCTGCTAGGCCGACTTATA

12,GCGATACAGAGTTGTAAGTTTGCTCACGAAC

Four-arm junction tile

8,GGCTTATTTTCTCGTTCCATGATCCGGACTATTTTCGTCCAGACTCGGCAGT

CTGATTTTGCAACCCGCCGACGGACCGGCTTTTACACAGACCGTGGTA

4,AACATCTTGCGTCGGGCGATCAGGGGTGAAGCTGTATAGCTA

15,TACCAGTATGTGGAGAAGACGTAACGCACCATGGGGCTTGCC

18,ACAGGCCCTTTAAGGATTAGCGGCCAACTAGGATTAAGTGAC

20,TTAGCCCTCCGGTCCATGCGTTTTACATTCCGGCGTAGTCAG

6,CGTGAGACGCATAGAACGCGCAAGATGTTTAGCTATACACGGTCTGTGTGC

CGGTCCGTCCCTAGTTGGCCGCTAATCCTTGCTACCCGCTTGA

7,CTAGGATCAGGTGCCCGTACATACTGGTAGGCAAGCCCCATGGAACGAGTA

AGCCTACCAGCTTCACCCCTGATCGCCCGACGGCATTAGTGAT

13,CGTGACACAAACCGGCAACGGAGGGCTAACTGACTACGCAGTCTGGACGT

AGTCCGGATCATGGTGCGTTACGTCTTCTCCCGGAGCATGACAG

16,ACTTAATCGCGGAACATAAAAGGGCCTGTGTCACTTAATGGCGGGTTGCT

CAGACTGCCGCGGAATGTAAAACGCATGGACCGTCGCCTTGGAT

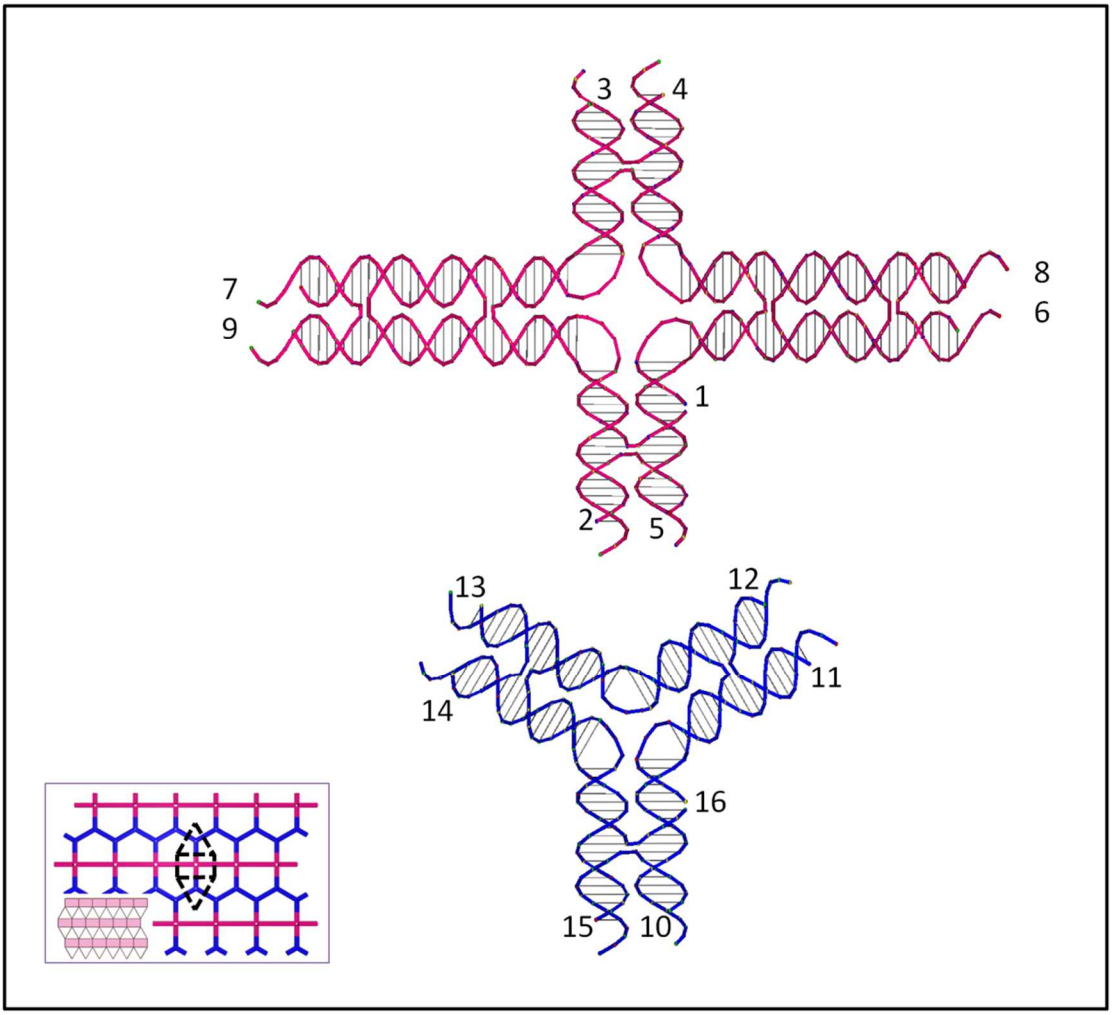
5,AGATATCACTAATGCCGCGTTCTATGCGTCT

14,AAGTCTGTCATGCTCCGACGGGCACCTGATC

17,CTAGTCAAGCGGGTAGCTATGTTCCGCGATT

19,AGATATCCAAGGCGACGTTGCCGGTTTGTGT

C. The shortened Prismatic pentagonal tiling design



Four-arm junction tile

1,AAGGAATTTTCGTCATGATGCCGGTCCCATTTTTTGAATCGATTCCGTAAGT

AGGGTTTTAATGCATGCCACGCGGATAATTTTTCGCCCTAGCAGATCA

2,ACAATCCGTTGCTAGGGCGATTATCCGCGTTTATGAACGCGACGCTAGGTA

ACGCGATTCGGCATGCATTCCCTACTTACGCATCACTGTAGA

3,CAGTGATGTAGTGCTACGTGC

4,GTAGCACTAGAATCGATTCAATGGGACCGGAAGTTTGTTTACACTTGTCCCC

GCAACACACATCATGACGTTCCTTTGATCCCGGGTTATAGA

5,TAACCCGGACGGATTGTGTGC

6,TAAGGGTCGAGGTGCAACGCGT

7,CCCTAAGAGGTCTATAGGGTCC

8,TAGGGACGCGTTGCAGTGTAAACAAACTTTGTGTTGCGGGGACAACCTCGA

C

9,CCTTAGGACCCTATGCGTCGCGTTCATAAGAATCGCGTTACCTAAGACCTCT

Three-arm junction tile

10,CCGACCCTACCATGCGATCTA

11,ACTCTTCCCTGGGAAACGAAACCGCGCGCCAGGGTCGGGCAC

12,CGGGTGGTGGGAAGAGTCGCT

13,GTCATCTGTGACGACCGATCGCGCCGTCGCACCACCCGAGCG

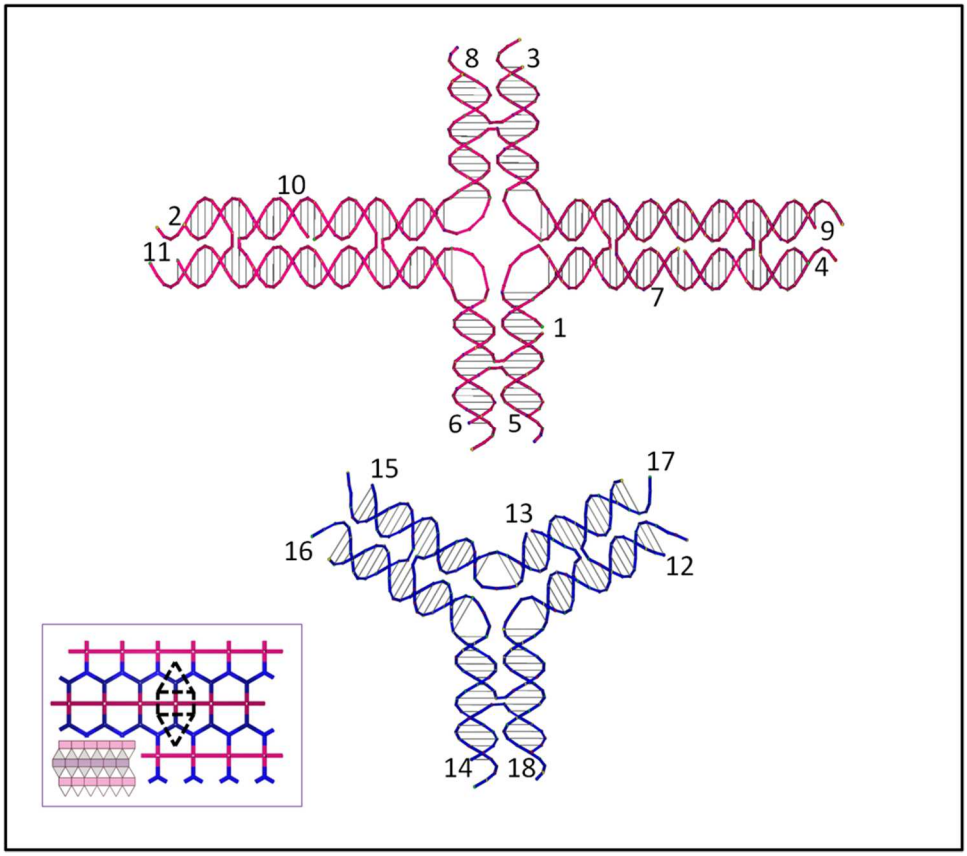
14,GTCTAAACACAGATGACTGAC

15,TCGCATGGTCCAGAGACTCGCCCCGGCCCCGTTTAGACGTCA

16,GCGGTTTTTTCGTTTCCCAGCGACGGCGCGTTTATCGGTCGTCGGGGCCGG

GGCTTTGAGTCTCTGGGGCGC

D. The shortened Prismatic pentagonal tiling with corrugated design



Four-arm junction tile

1,CTTAGATTTTGGGATTAGTGTGTGGCGTATCTTTTCTTCATTGGATCCGAAC

GGGCTTTTACGCTATGGCGTCACCCGCTCTTTTGAACCAGCTCGATCG

2,GGCGGAGCGCAGCCTATAAGGTAGGGAAAGCCATAGCGTGCCCGTTCGGA

TCTCTACCGCAA

3,GGCGTGTTATCCAATGAAGGATACGCCACAAAGTGTCGGTCGCGCTGTATT

CGGATCACACTCG

4,CACGTGACGGAAATGTATCCTTGCTGACTCACTAATCCCTCTAAGCGATCTG

TGTCGAGCAA

6,AGTAGTATGGAGCTGGTTCGAGCGGGTGACGGATGGGTCCCAATCGATCTT

CCGGCCCGGAATC

7,GATACATTTCCAATACAGCGCGACCGACACTTAGTCAGCAAG

10,TTATAGGCTGCAAGATCGATTGGGACCCATCCTTTCCCTACC

5,TCGACACACATACTACTTCTG

8,GGTAGAGATAACACGCCTCTG

9,TGTGATCCGGTCACGTGGATT

11,CCGGGCCGGGCTCCGCCCGAG

Three-arm junction tile

13,ACGAGGTTTCTTCCCAGGCAATTTGGGCCCTTTCGGTGCAGGGGTGTACCC

CGGTTTAGTCCCAGGTCCCAT

12,ACGCAGCTCAGACCTGGGACTCCGGGGTACACGCGCTAAACAGA

14,ATACCCATACCCTGCACCGGGGCCCAAATTACAGGTCCGCG

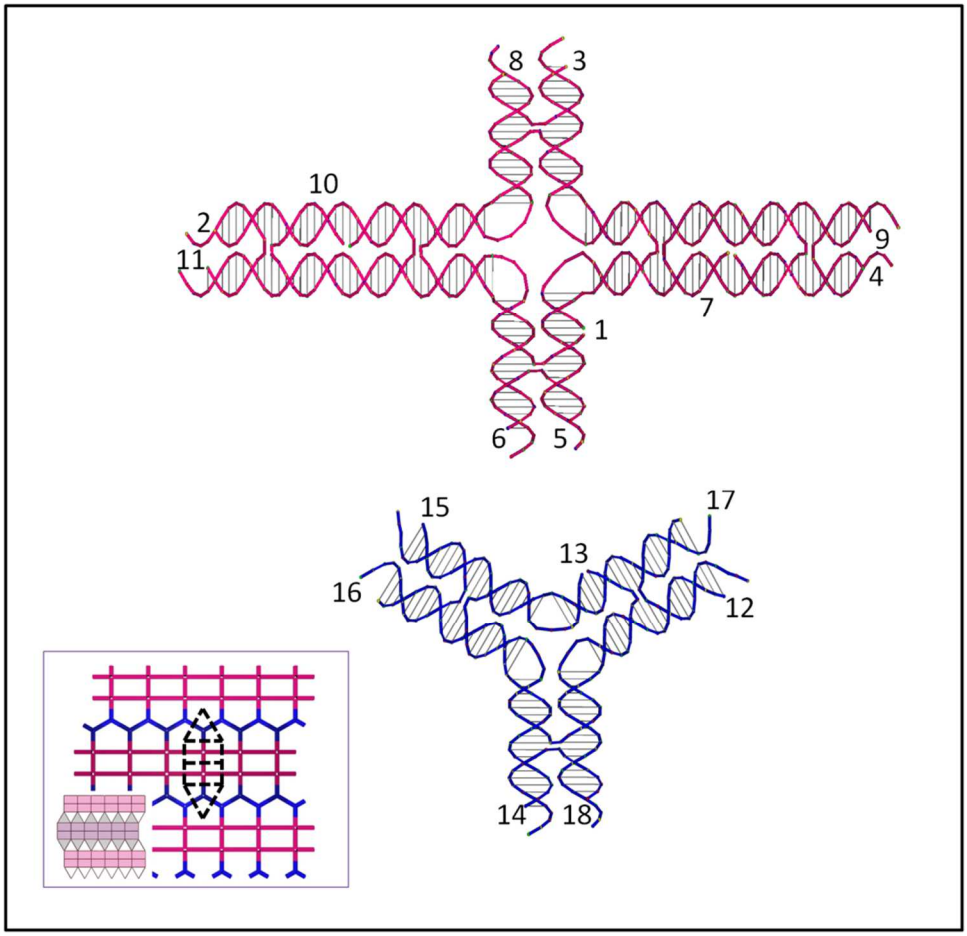
15,ATACCGAAAAGGCCTGGGAAGCCTCGTATGGGAGCTCCGGCGG

16,CGCGCGCGGACCTGTCTTTTCGGTATCATG

17,CGCGCCGCCGGAGCTCTGAGCTGCGTCATG

18,TTTAGCGCTATGGGTATTTGC

E. The 2-uniform tiling with corrugated design



Four-arm junction tile

1,CTTAGATTTTGGGATTAGTGTGTGGCGTATCTTTTCTTCATTGGATCCGAAC

GGGCTTTTACGCTATGGCGTCACCCGCTCTTTTGAACCAGCTCGATCG

2,GGCGGAGCGCAGCCTATAAGGTAGGGAAAGCCATAGCGTGCCCGTTCGGA

TCTCTACCGCAA

3,GGCGTGTTATCCAATGAAGGATACGCCACAAAGTGTCGGTCGCGCTGTATT

CGGATCACACTCG

4,CACGTGACGGAAATGTATCCTTGCTGACTCACTAATCCCTCTAAGCGATCTG

TGTCGAAGGA

6,AGTAGTATGGAGCTGGTTCGAGCGGGTGACGGATGGGTCCCAATCGATCTT

CCGGCCCGGAATC

7,GATACATTTCCAATACAGCGCGACCGACACTTAGTCAGCAAG

10,TTATAGGCTGCAAGATCGATTGGGACCCATCCTTTCCCTACC

5,TCGACACACATACTACTTCCT

8,GGTAGAGATAACACGCCTCTG

9,TGTGATCCGGTCACGTGGATT

11,CCGGGCCGGGCTCCGCCCGAG

Three-arm junction tile

13,ACGAGGTTTCTTCCCAGGCAATTTGGGCCCTTTCGGTGCAGGGGTGTACCC

CGGTTTAGTCCCAGGTCCCAT

12,ACGCAGCTCAGACCTGGGACTCCGGGGTACACGCGCTAAACAGA

14,ATACCCATACCCTGCACCGGGGCCCAAATTACAGGTCCGCG

15,ATACCGAAAAGGCCTGGGAAGCCTCGTATGGGAGCTCCGGCGG

16,CGCGCGCGGACCTGTCTTTTCGGTATCATG

17,CGCGCCGCCGGAGCTCTGAGCTGCGTCATG

18,TTTAGCGCTATGGGTATTTGC

APPENDIX B

SUPPORTING INFORMATION FOR CHAPTER 3

OMPLEX WIREFRAME DNA ORIGAMI NANOSTRUCTURES

WITH MULTIARM JUNCTION VERTICES

Section 1. Wireframe DNA Origami Design

S1.1 Details of the design strategy

The goal of our design was to construct arbitrarily shaped wireframe architectures composed of line segments and vertices. N × 4 multiarm junctions are ideal candidates to represent vertices because different types of junctions have been constructed and studied in the tile self-assembly system, such as 4×4 (H. Yan *et al., Science* 301, 1882-1884, 2003), 3×4 (Y. He *et al., J Am Chem Soc* 127, 12202-12203, 2005), and 6×4 (Y. He, Y. Tian, A. E. Ribbe, C. D. Mao, *J Am Chem Soc* 128, 15978-15979, 2006) junctions. These previous studies provided basic guidance for the design of more complex and various vertices in our wire-frame DNA origami nanostructures. One important consideration for our origami design was how to route the scaffold strand through all of the vertices and lines. Between every two adjacent vertices (junctions), two antiparallel cross-linked DNA helixes are employed as the line segments. We designed a “loop” and “bridge” method to realize scaffold routing.

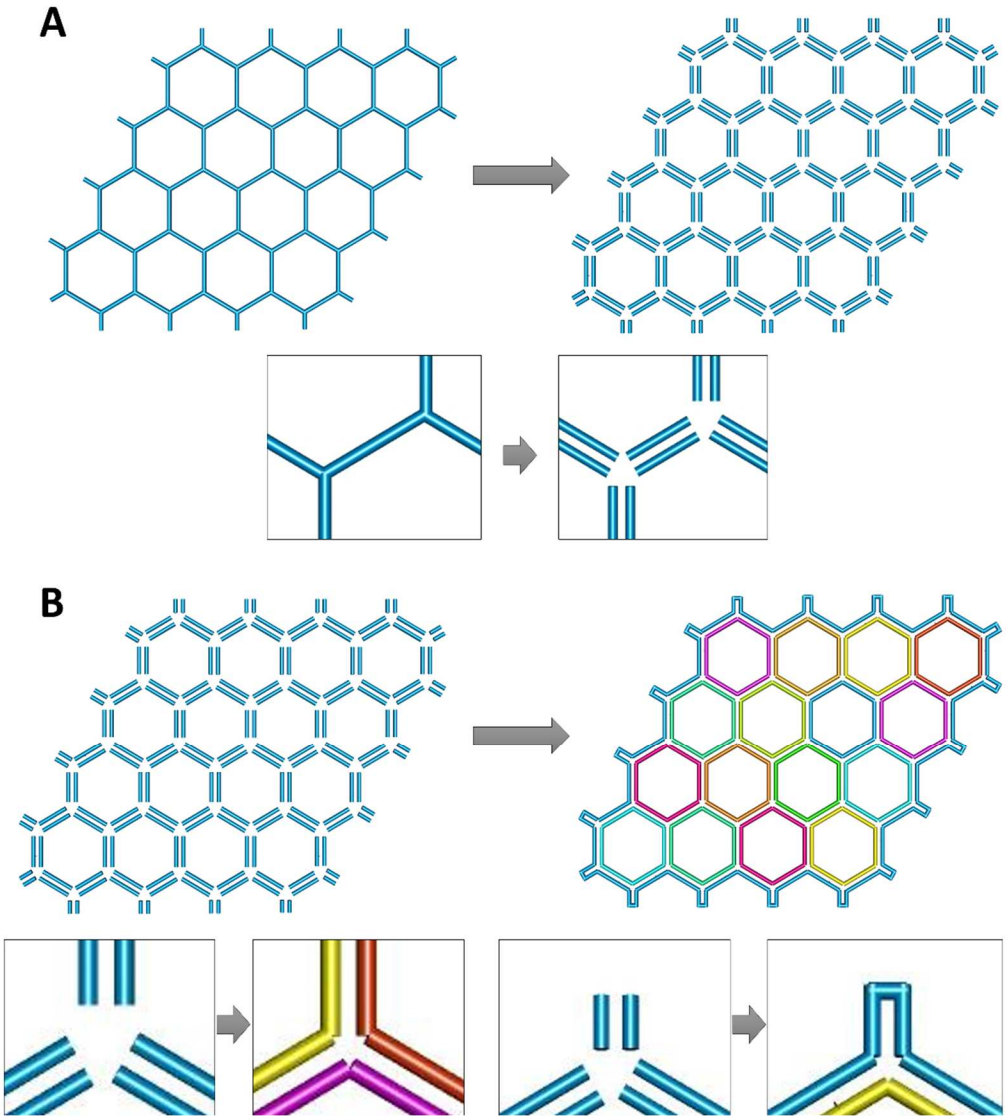


Fig. S1. Illustration of the design. Step 1: Double lines and loop scaffold. The hexagonal Platonic tiling is used here as an example to illustrate the design steps. (**A**) The first design step was to double any lines in the pattern. (**B**) Next, the lines meeting at each vertex were connected as shown, and the double lines in the end were linked to form an end loop.

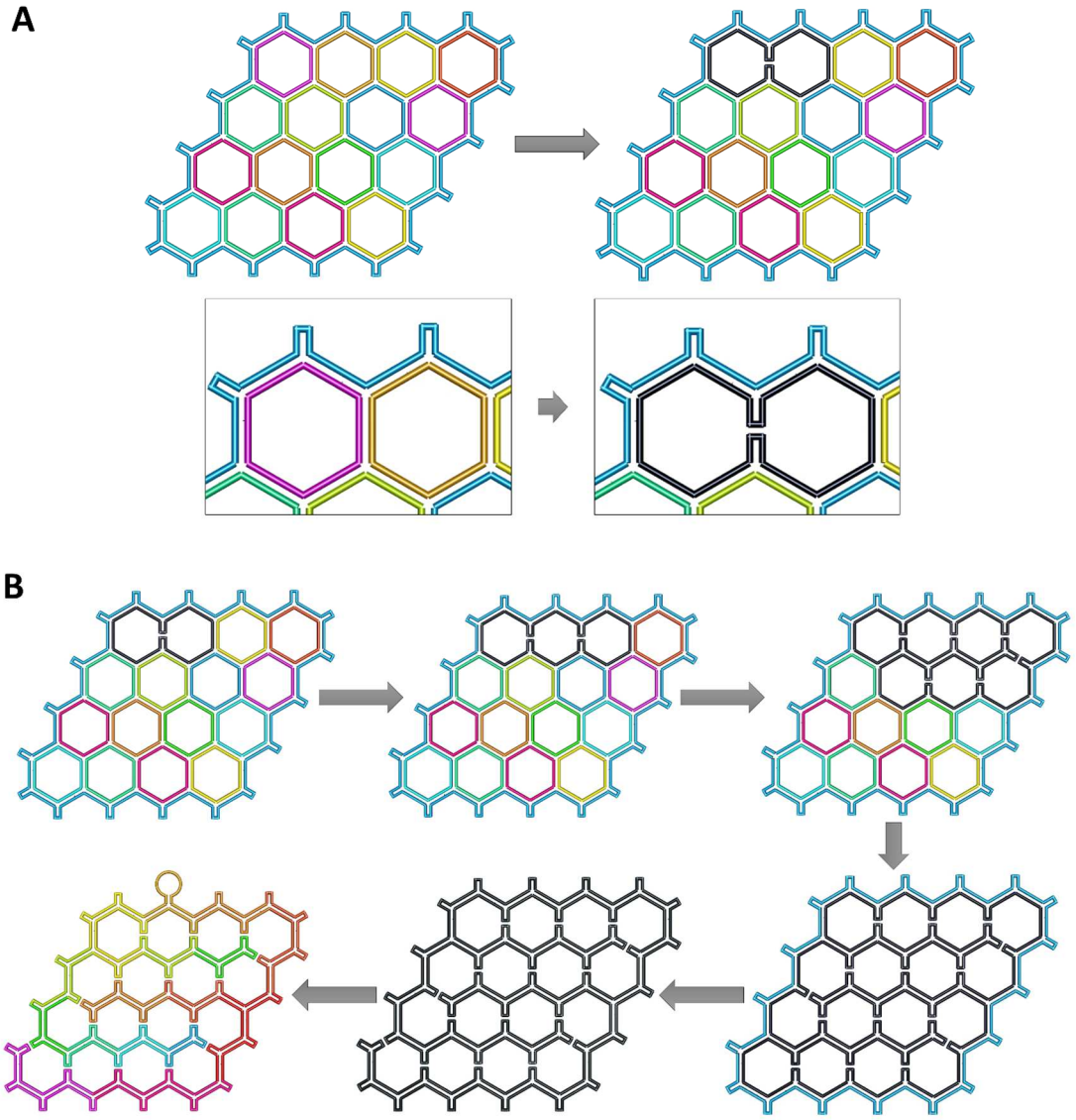


Fig. S2. Illustration of the design. Step 2: Bridge scaffold. The looping step produces several individual loops. In this hexagonal tiling example, we obtained 16 hexagons inside and one long loop outside. (**A**) The first bridge is constructed between two neighboring loops (pink and orange). (**B**) The inner loops are interconnected one by one, and finally the whole structure is composed of a single loop by bridging the inner loop (black) with the outside long loop (blue). The color changes serve as a visual guide for the loop's travel direction.

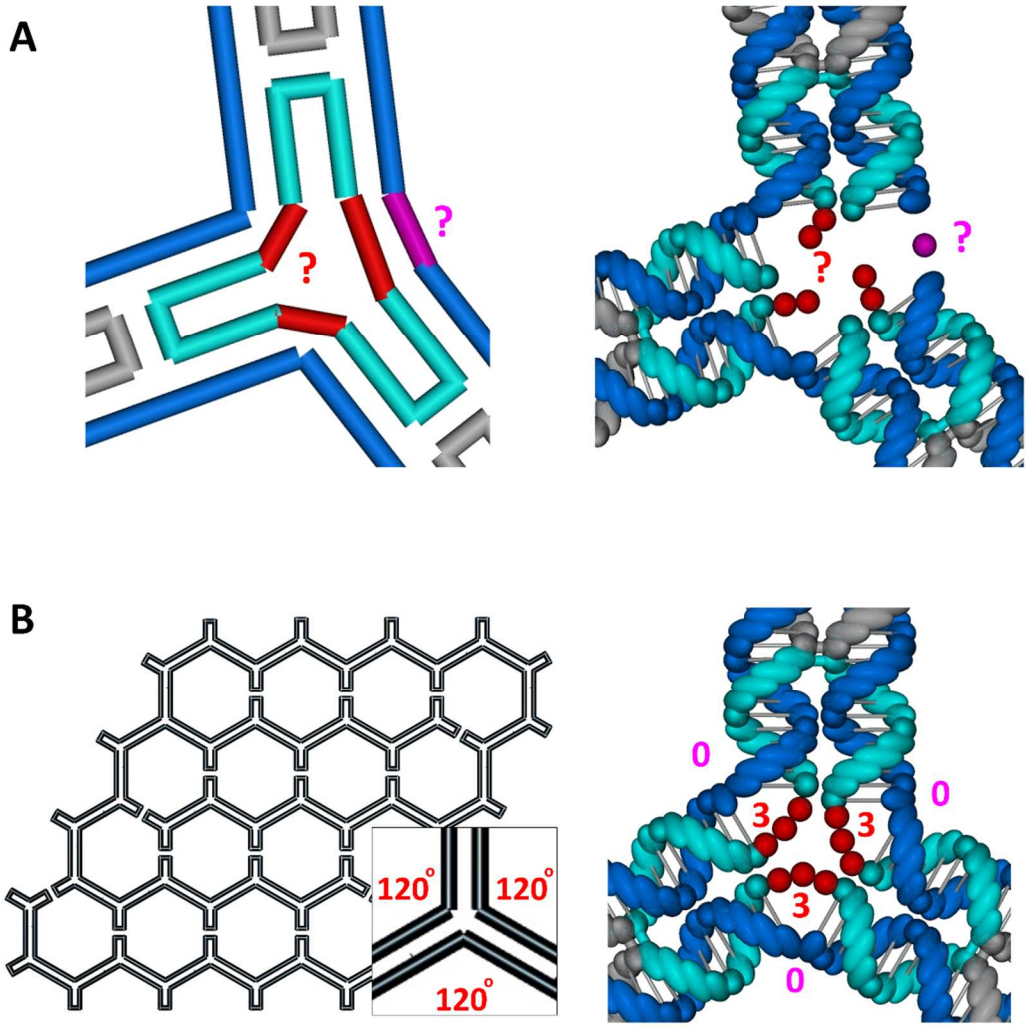


Fig. S3. Illustration of the design. Steps 3: Angle adjustment at vertex. At each vertex, neighboring arms require certain angles determined by the target pattern. (**A**) Poly T loops (red) and unpaired bases in the scaffold strand (pink) need to be inserted surrounding the vertex position to satisfy the angle requirements. The number of unpaired bases to be inserted need to be determined on an individual basis. (**B**) In our hexagonal tiling example, the pattern has homogenous vertices with three 120º angles. T3 loops in the staple strands and zero unpaired base in the scaffold were selected to generate this angle.

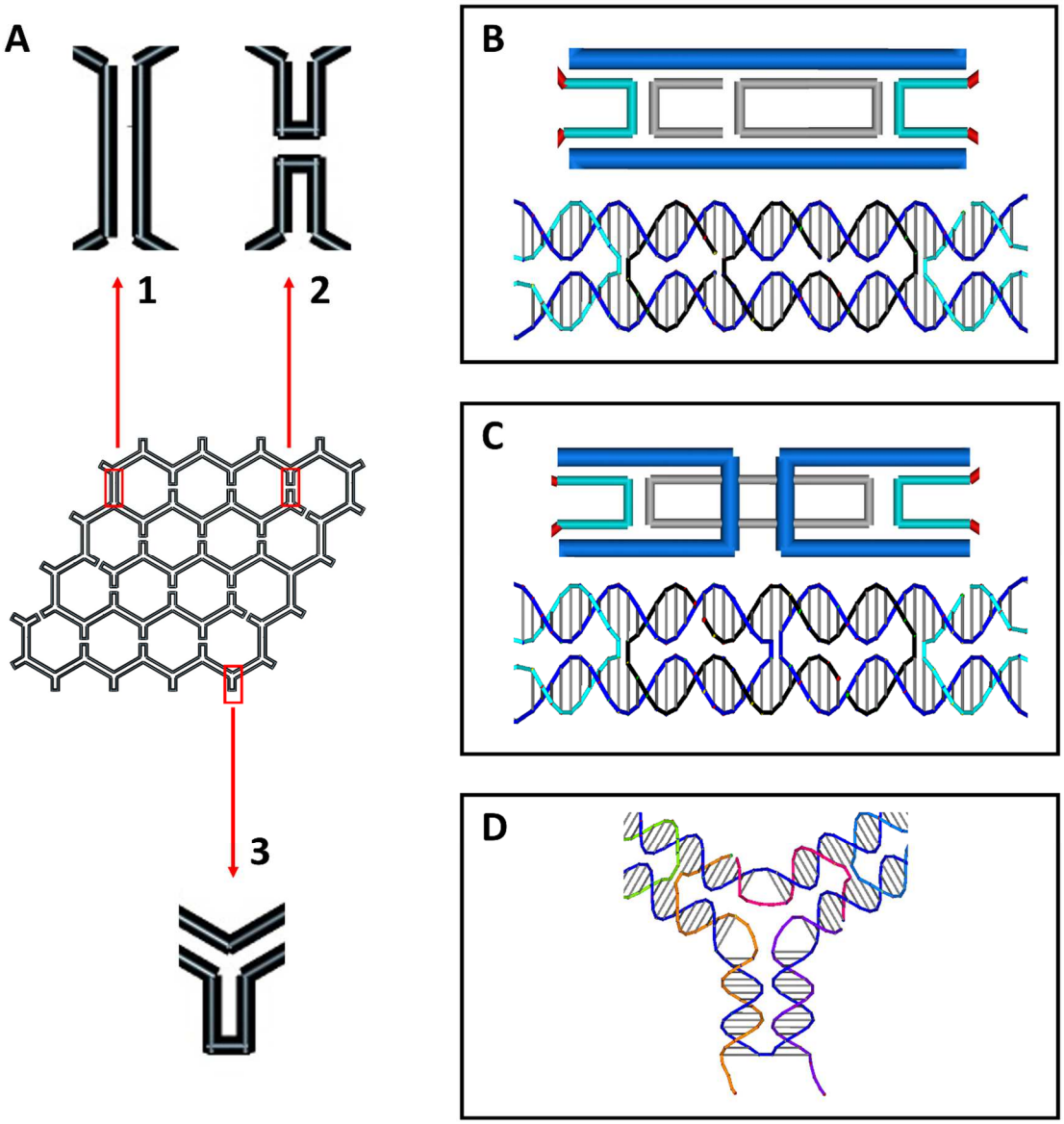


Fig. S4. Illustration of the design. Steps 4: Filling in the staple strands. (**A**) In any target pattern, three different types of edges can be found: the first type consists of two antiparallel strands without a scaffold bridge (**1**), the second is the edge containing a double crossover of the scaffold (**2**), and the third is the end loop in the outer rim of the pattern (**3**). (**B-D**) How the staple strands are arranged into the three types of scaffold patterns, respectively. (**B** and **C**) Every edge has five turns of DNA helix in length. (**D**) T4 tails are attached to the ends of the staple DNA and are hanging out to prevent blunt-end stacking.

S1.2 Design of simple Platonic tilings based on hexagon, square, and triangle geometries

S1.2.1 Scaffold routing and angle adjustment

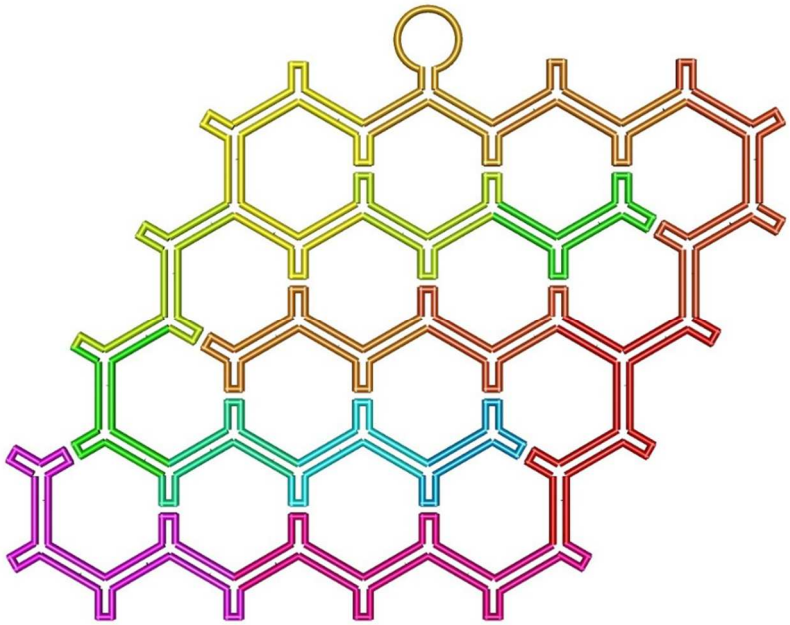


Fig. S5. Scaffold-folding path of the honeycomb Platonic tiling. The changing colors provide a visual guide showing how the scaffold travels.

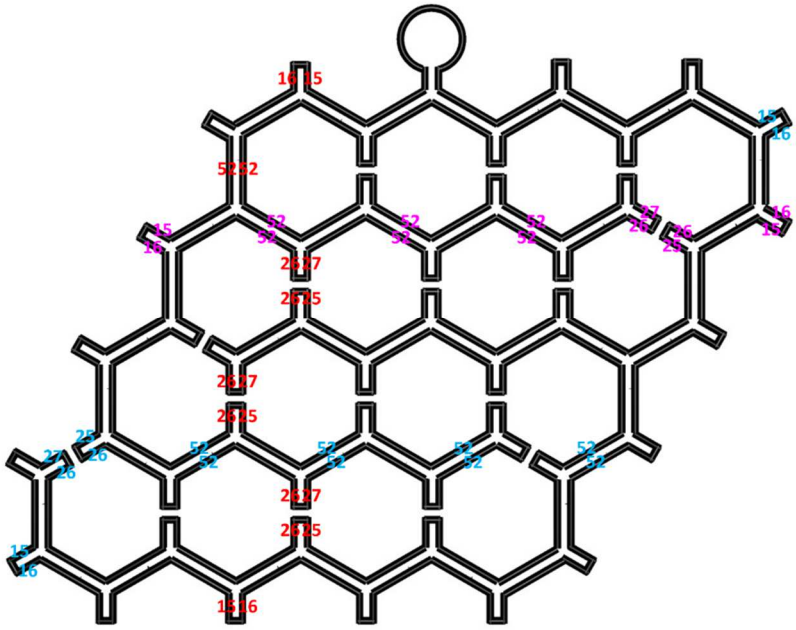


Fig. S6. Number of bases for the honeycomb Platonic tiling scaffold. The numbers refer to the numbers of nucleotides between the junction points.

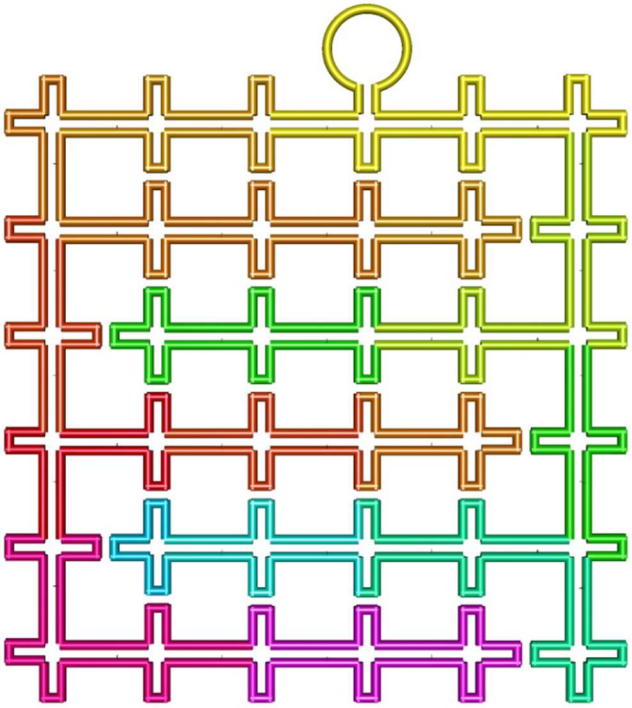


Fig. S7. Scaffold-folding path of the square Platonic tiling. The changing colors provide a visual guide showing how the scaffold travels.

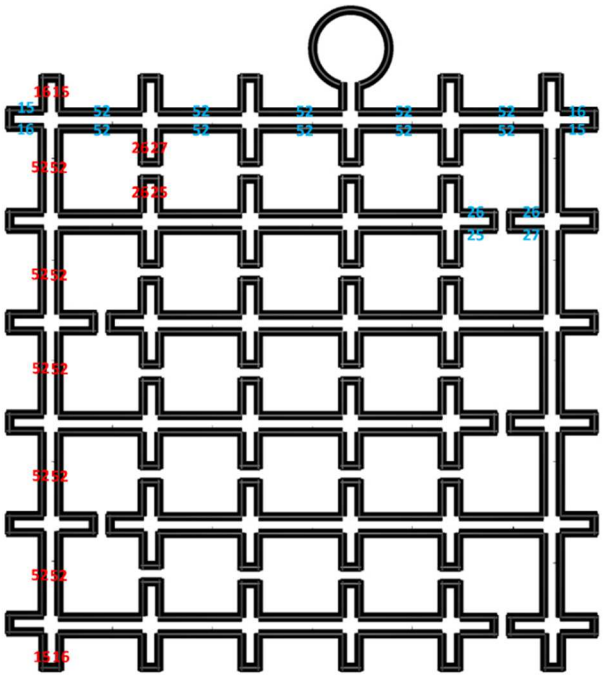


Fig. S8. Number of bases for the square Platonic tiling scaffold. The numbers refer to the distances (number of nucleotides) between the junction points.

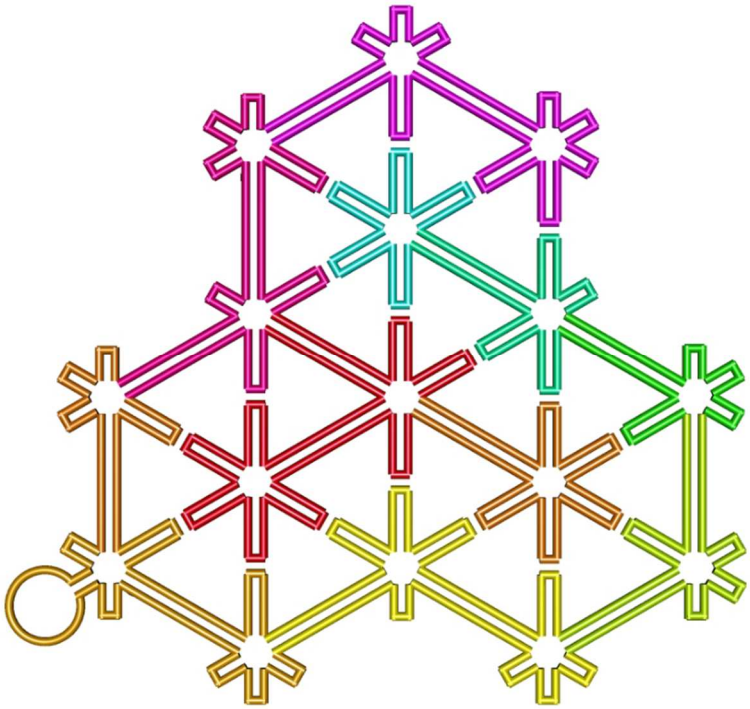


Fig. S9. Scaffold-folding path of the triangular Platonic tiling. The changing colors provide a visual guide showing how the scaffold travels.

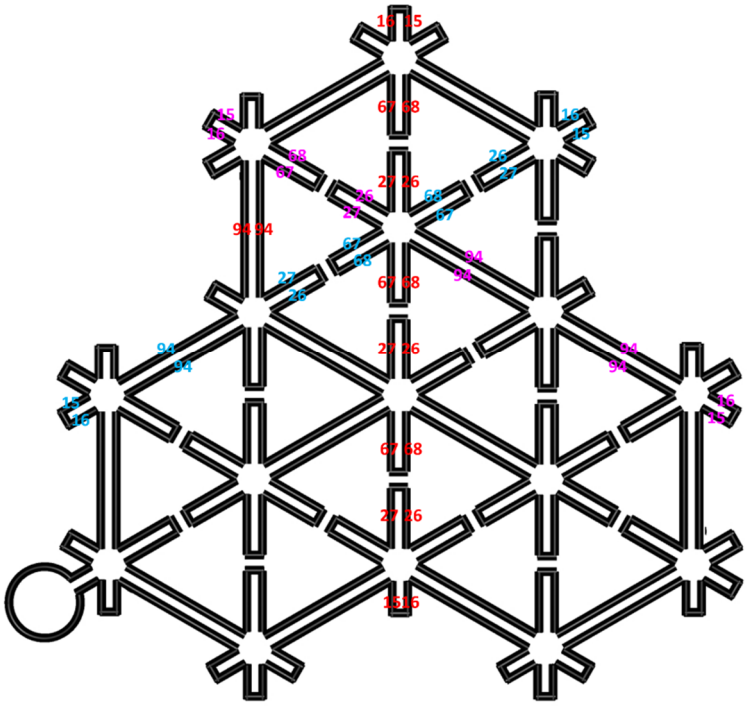


Fig. S10. Number of bases for the triangular Platonic tiling scaffold. The numbers refer to the distances (number of nucleotides) between the junction points.

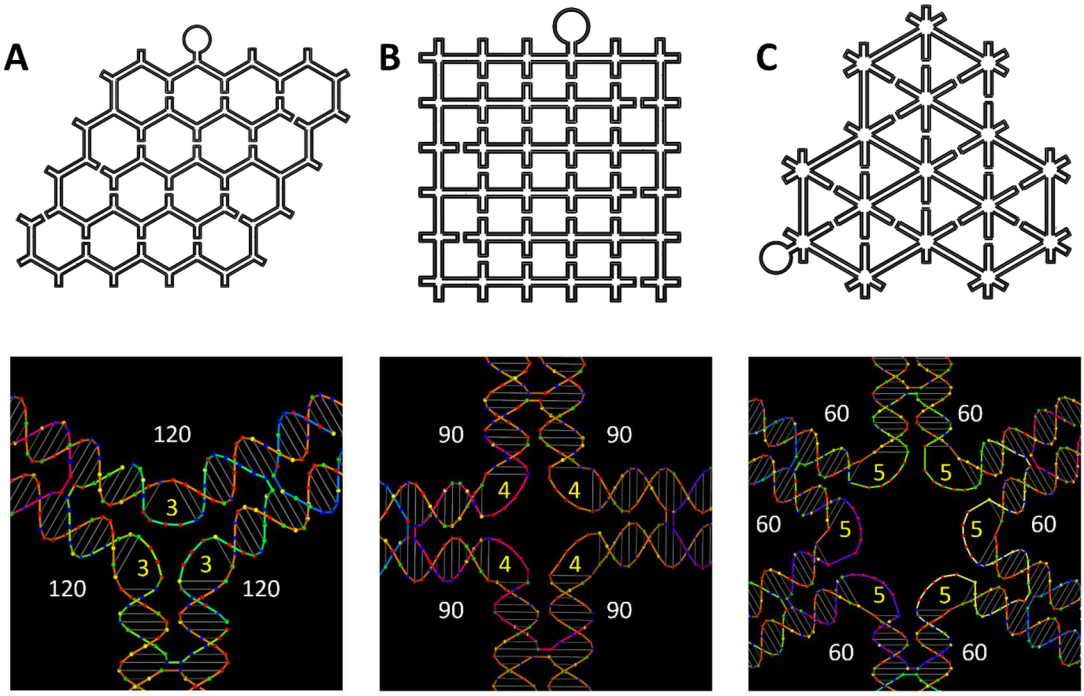


Fig. S11. Angle adjustment in the three Platonic tilings. (A-C) T3, T4, and T5 were selected to form the angle 120º, 90º, and 60º, respectively. There are no unpaired nucleotides in the scaffold strand.

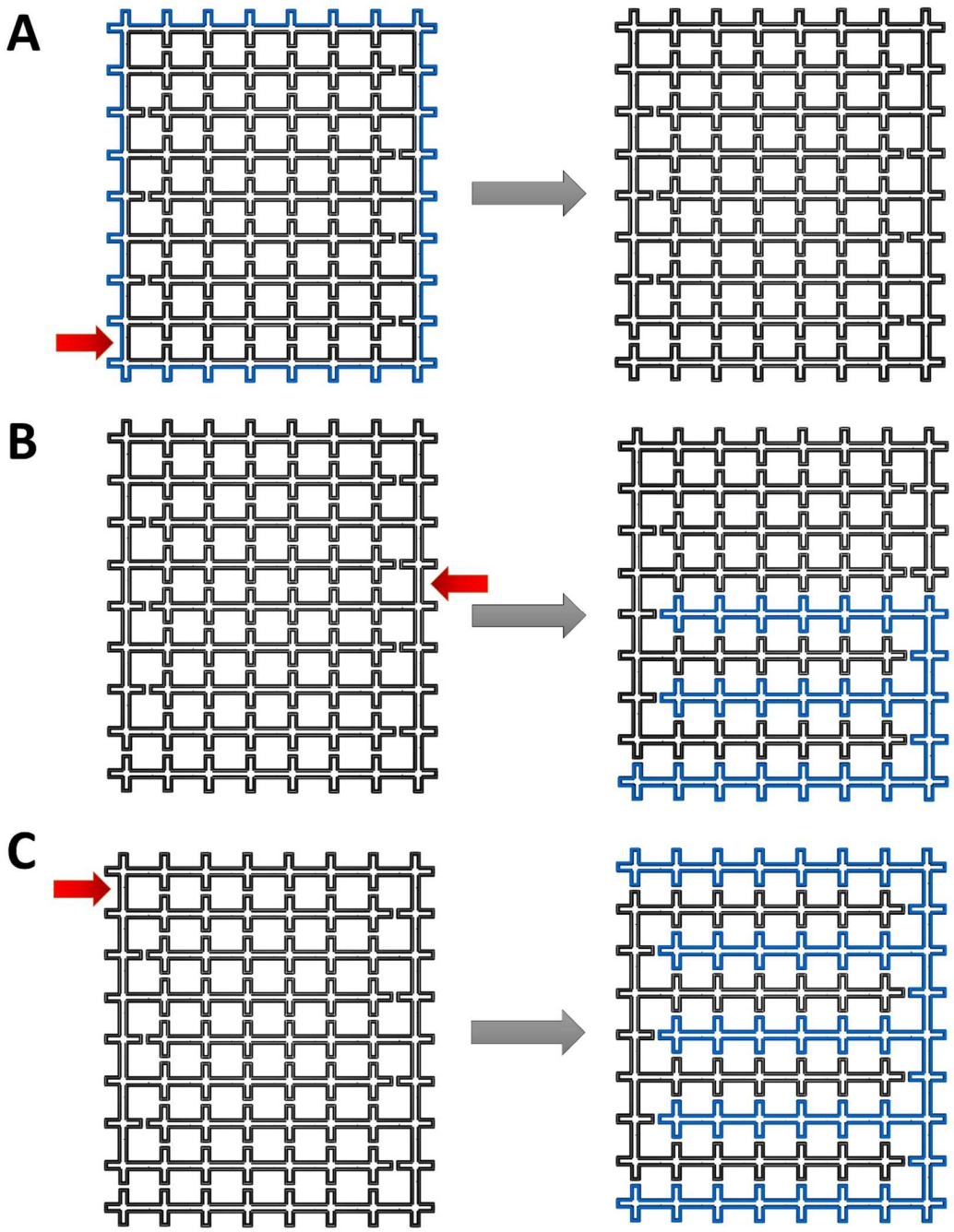


Fig. S12. Bridge method for the square tiling with two scaffolds. (**A**) The last step of bridging is to connect the inner loop with the outside loop by creating a double crossover (DX) (red arrow). (**B**) Adding another DX at the point of the red arrow results in two scaffolds of different lengths. (**C**) The position of the second DX can be changed to create different lengths of the two scaffolds (black and blue loops). In **C**, the contacts between the two differently scaffolded structures are maximized, which is expected to improve the formation yield of the final structure.

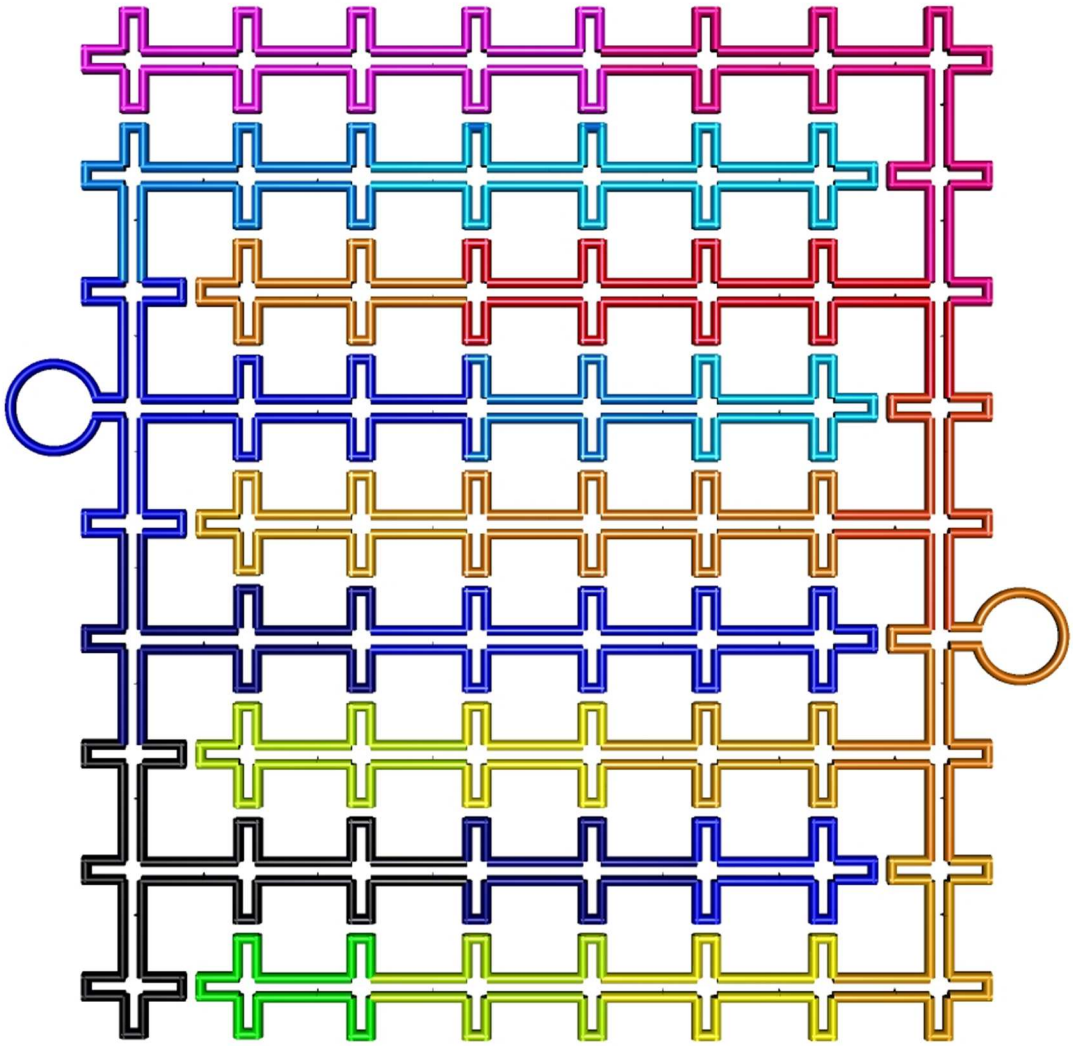


Fig. S13. Scaffold-folding path of the two-scaffold square tiling. The changing colors provide a visual guide showing how the scaffold travels.

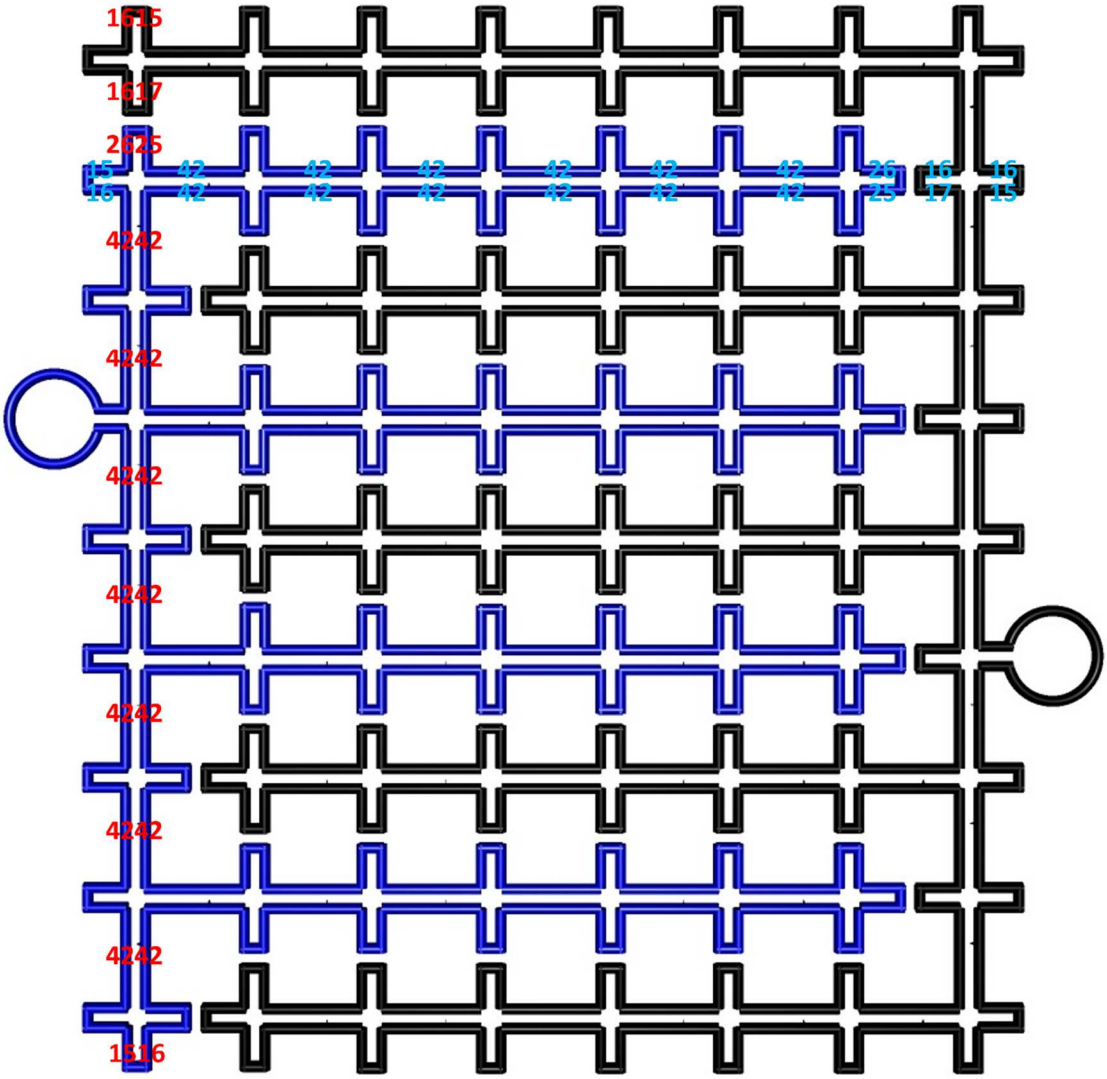


Fig. S14. Number of bases for the two-scaffold square tiling scaffold. The black and blue loops indicate the M13mp18 and PhiX174 scaffolds, respectively. The numbers refer to the distances (number of nucleotides) between the junction points.

S1.2.2 Hierarchical assembly method for the 3 × 3 origami array

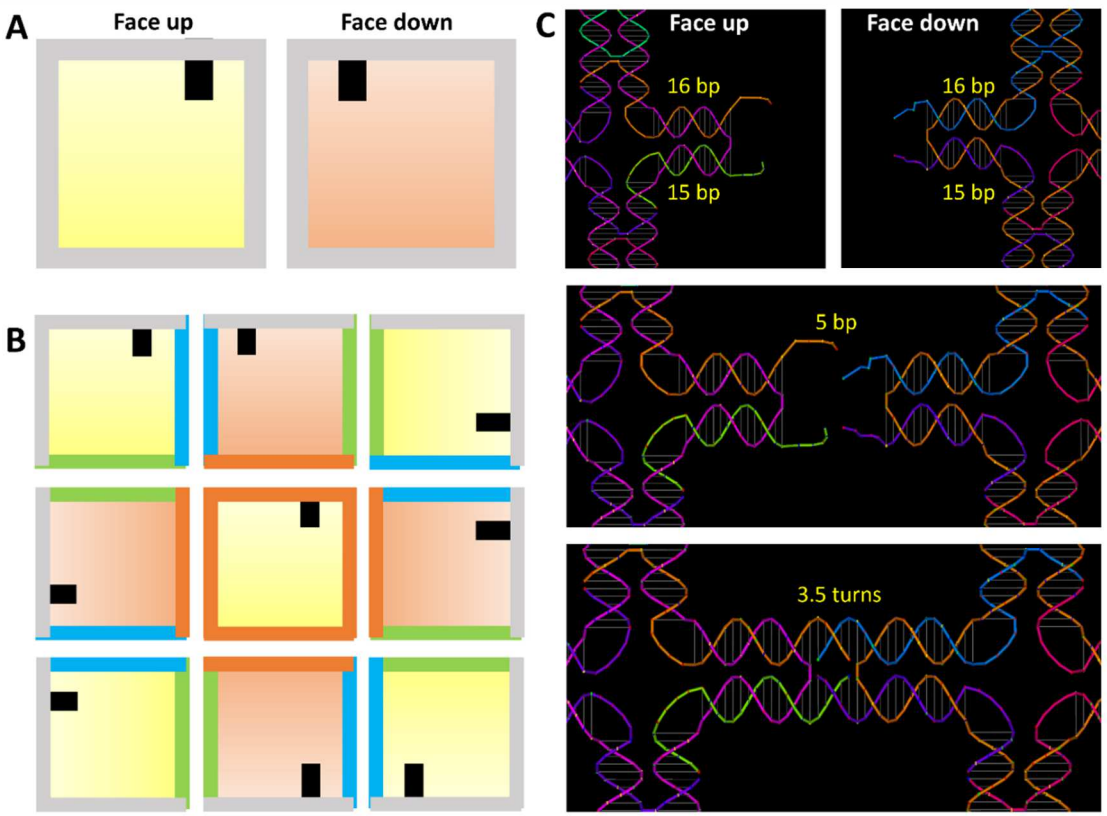


Fig. S15. Hierarchical assembly of the 3×3 square origami array. (**A**) The unit square origami. The yellow and orange colors indicate the two faces of the square, and the inside black marks indicate the chirality of the square and help to orientate the four edges. The gray sides represent the possible sticky ends. (**B**) Three types of origami in the stoichiometry ratio of 1:4:4 form the 3 × 3 origami array so that one is located in the center, four on the sides, and four on the corners. Although each unit is chiral, the overall assembly has a 4-fold rotational symmetry on the exposed edges. The one origami in the center has the same series of sticky ends in its four edges (colored as dark orange). The origami located on the sides of the array each have one edge (orange) paired with the center origami, and the other two edges (green and blue) are connected with two corner origami. The last edge (gray) is tailed with poly T instead of sticky ends. The origami located on the corners each has two neighboring edges (green and blue) paired with the side origami, and the other two edges are tailed with poly T (gray). The matching sticky ends on their edges help to bring the nine origami units together to form the 3 × 3 origami array. (**C**) One example shows how the sticky ends on the edges of the origami units are paired to each other. The sequences of the pairs of five-bp-long sticky ends were designed as extended single-stranded overhangs at each position of the protruding arms. After hybridization, the newly formed edges between two adjacent origami units have 3.5 DNA helical turns in length, which ensures the neighboring origami units in the array are alternately facing in opposite directions. This corrugated design is expected to reduce the overall curvature of the final assembly.

Additional notes: The single scaffold square tiling origami (with 5 × 5 square cavities) has 6 protruding arms on each side of the square, which are designed as the sticky ends for hierarchical assembly. By simply appending the staple strands on the edges with unique sticky ends (to replace the poly T tails) while maintaining the sequences of all the internal helpers, we can design a set of origami with different sticky ends on their four edges. By symmetrically creating the series of sticky ends between origami, three unique DNA origami units in a 1:4:4 molar ratio, instead of nine different units, can be used to create the 3 × 3 array. This greatly reduces the range of sequence design for the sticky ends. Every series of sticky ends along the edges of the each origami unit contain six pairs of 5-base pair (bp)-long ssDNAs that were precisely assigned with specific sequences that allow corrugated contacts between designed neighboring origami with the least mismatching. AFM images confirmed the formation of the 3 × 3 square origami array (equivalent to a 17 × 17 array of 4 × 4 tiles) with ~330 nm in both dimensions (Fig. 2E). The seams between the neighboring individual origami are visible under AFM, as the lengths of these edges are slightly shorter than those in the rest of the structure (3.5 turns instead of 5 turns).

S1.3 Quasicrystalline structures and curved origami design

S1.3.1 Scaffold routing and angle adjustment

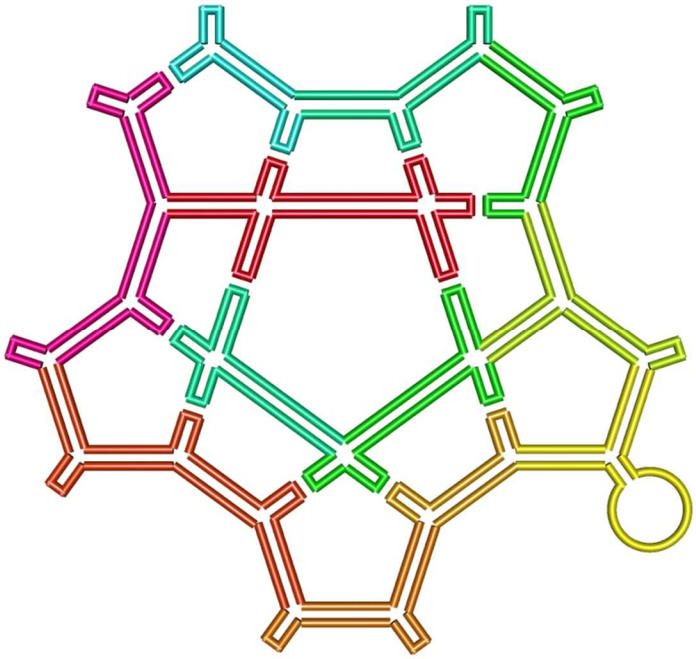


Fig. S16. Scaffold-folding path of the star shape. The changing colors provide a visual guide showing how the scaffold travels.

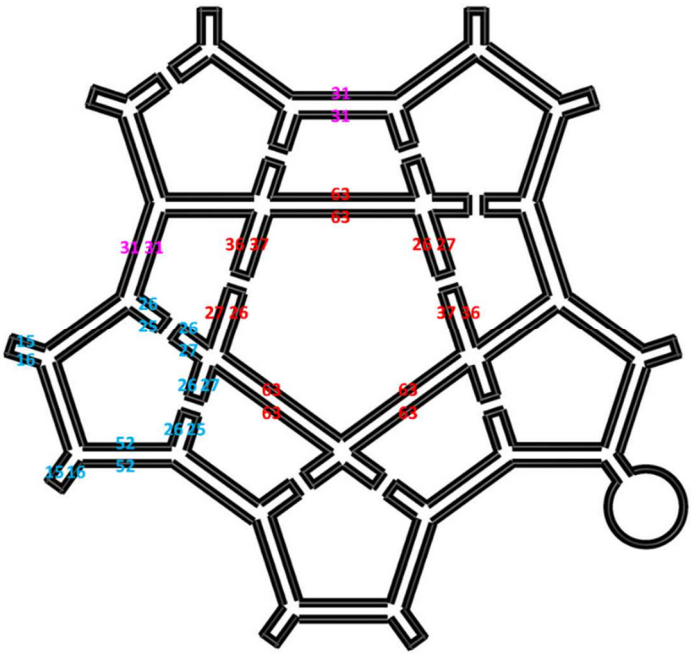


Fig. S17. Number of bases for star-shape scaffold. The numbers refer to the distances (number of nucleotides) between the junction points.

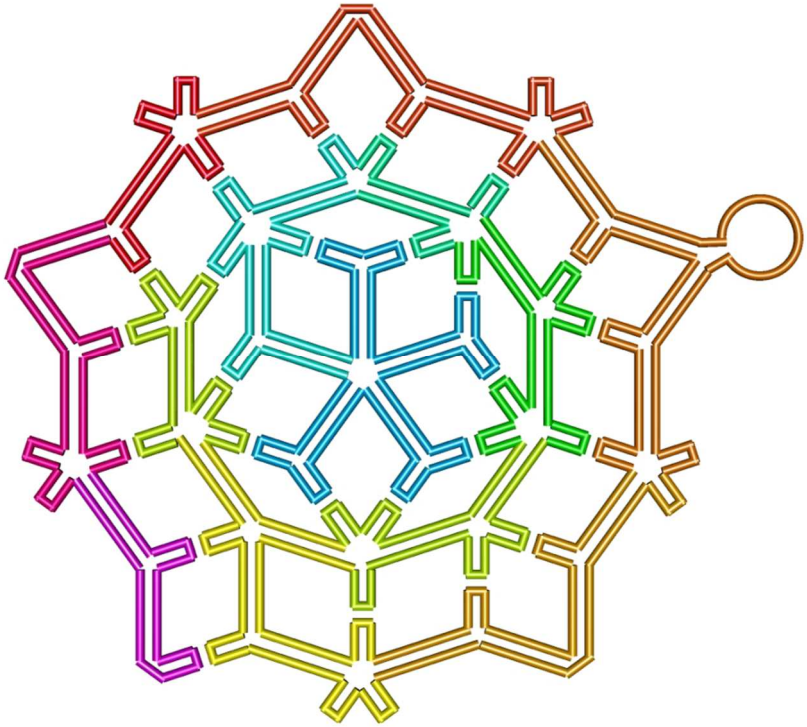


Fig. S18. Scaffold-folding path of the 2D Penrose tiling. The changing colors provide a visual guide showing how the scaffold travels.

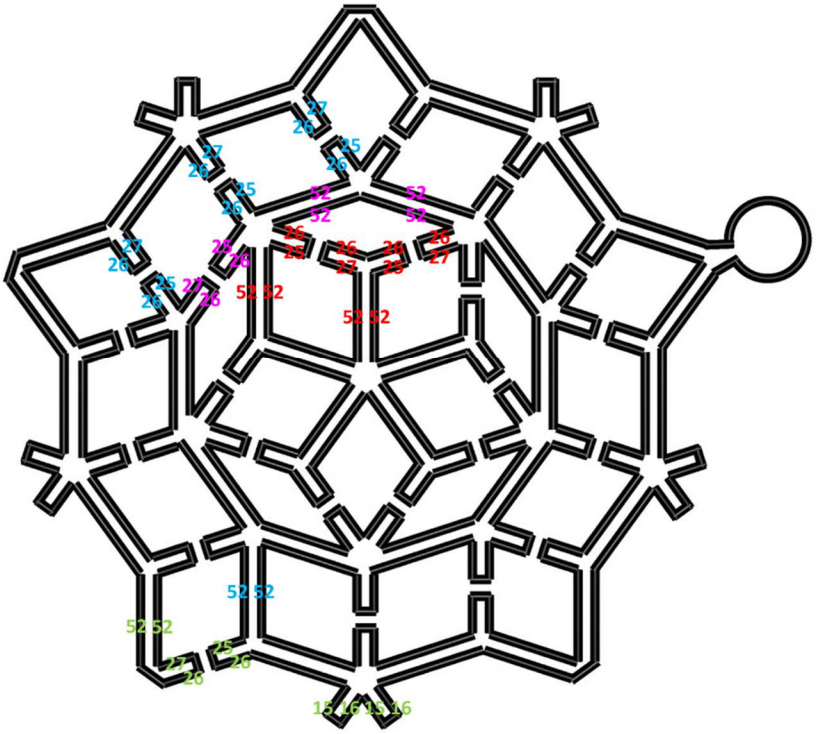


Fig. S19. Number of bases for the 2D Penrose tiling scaffold. The numbers refer to the distances (number of nucleotides) between the junction points.

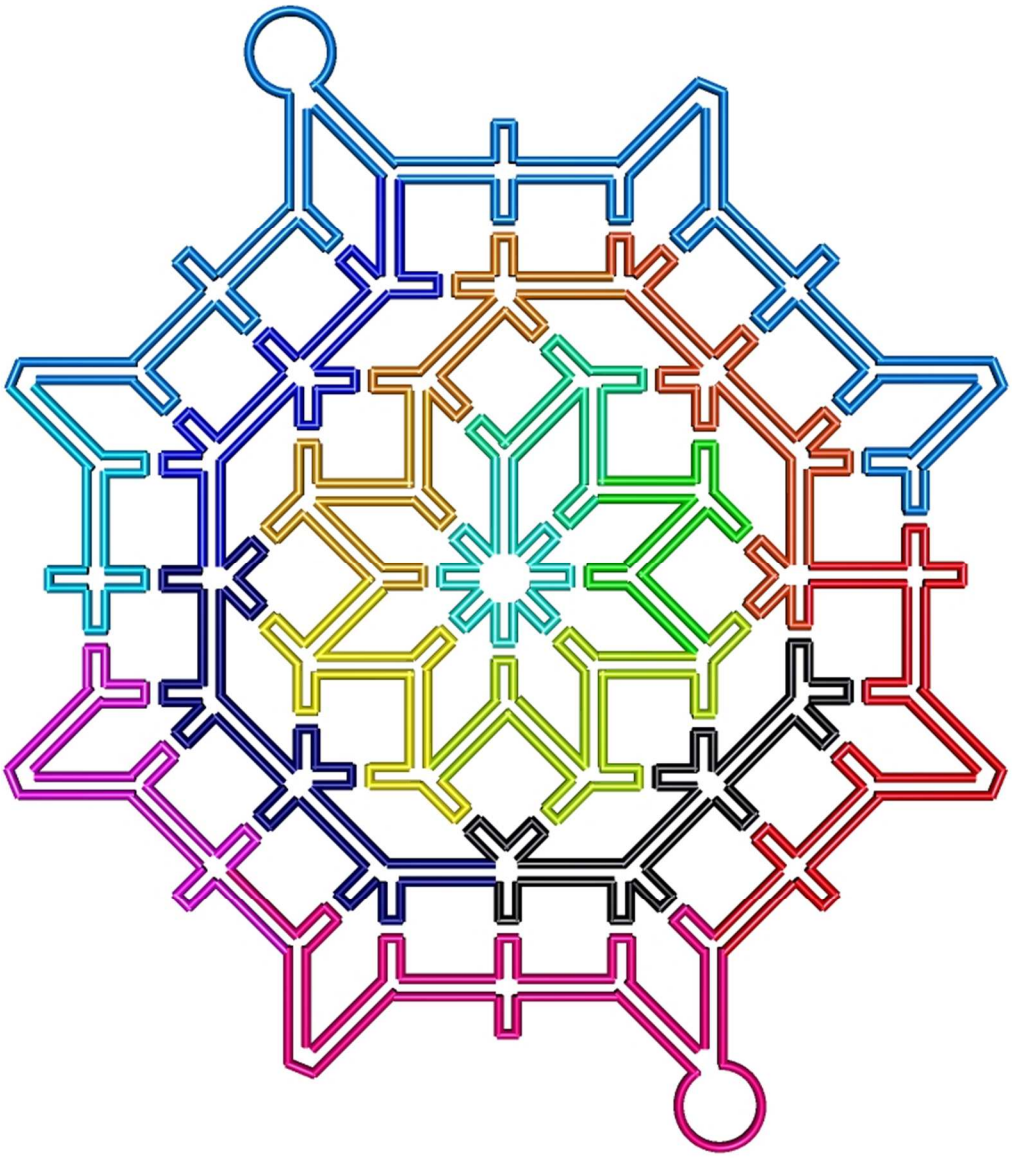


Fig. S20. Scaffold-folding path of the 8-fold quasicrystalline 2D pattern. The changing colors provide a visual guide showing how the scaffold travels. There are two independent scaffolds in this design, one in black and blue (PhiX174 scaffold), and one in varying colors (M13mp18 scaffold).

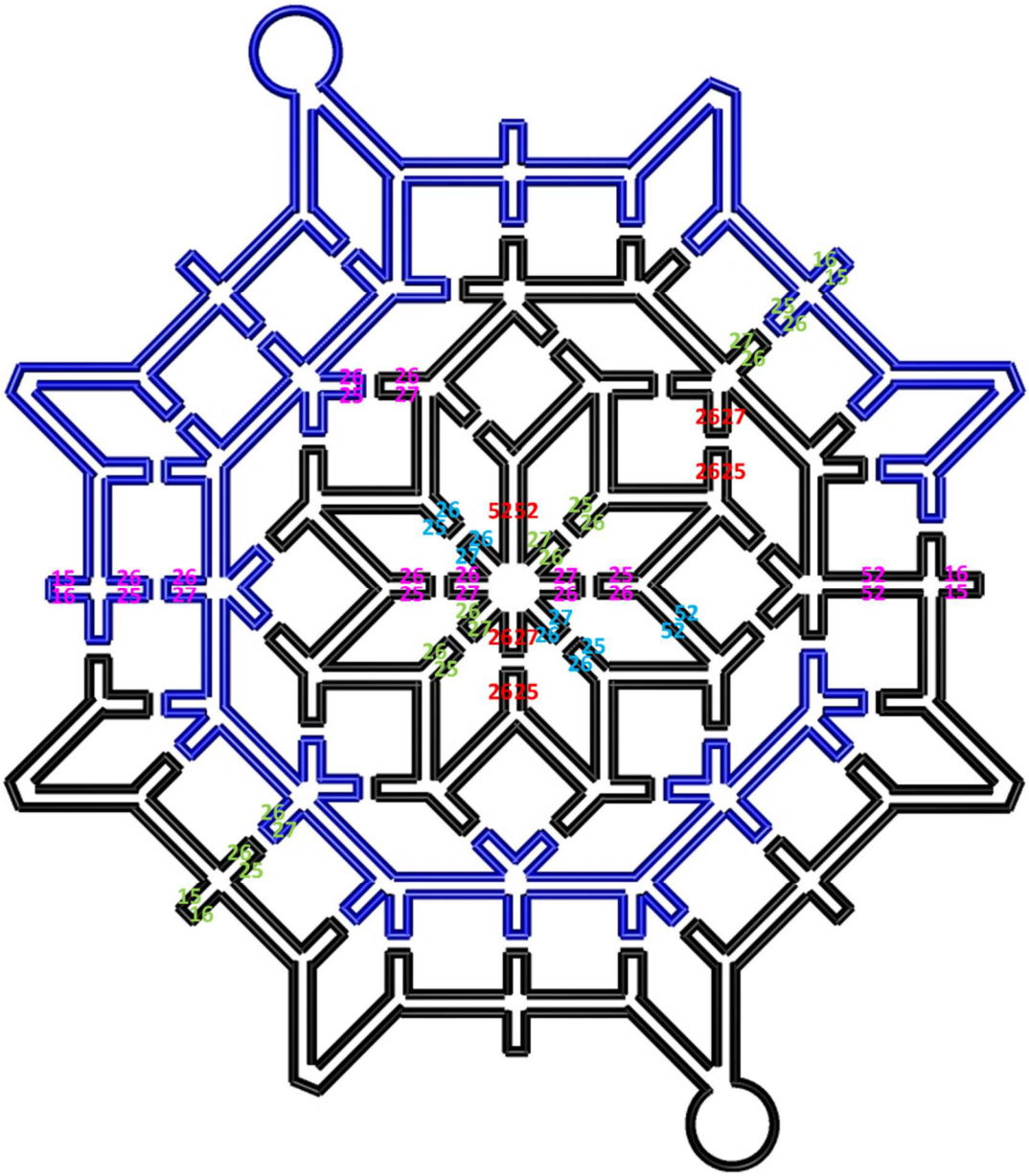


Fig. S21. Number of bases for the 8-fold quasicrystalline 2D pattern scaffold. The black and blue loops represent the M13mp18 and PhiX174 scaffolds, respectively. The numbers refer to the distances (number of nucleotides) between the junction points.

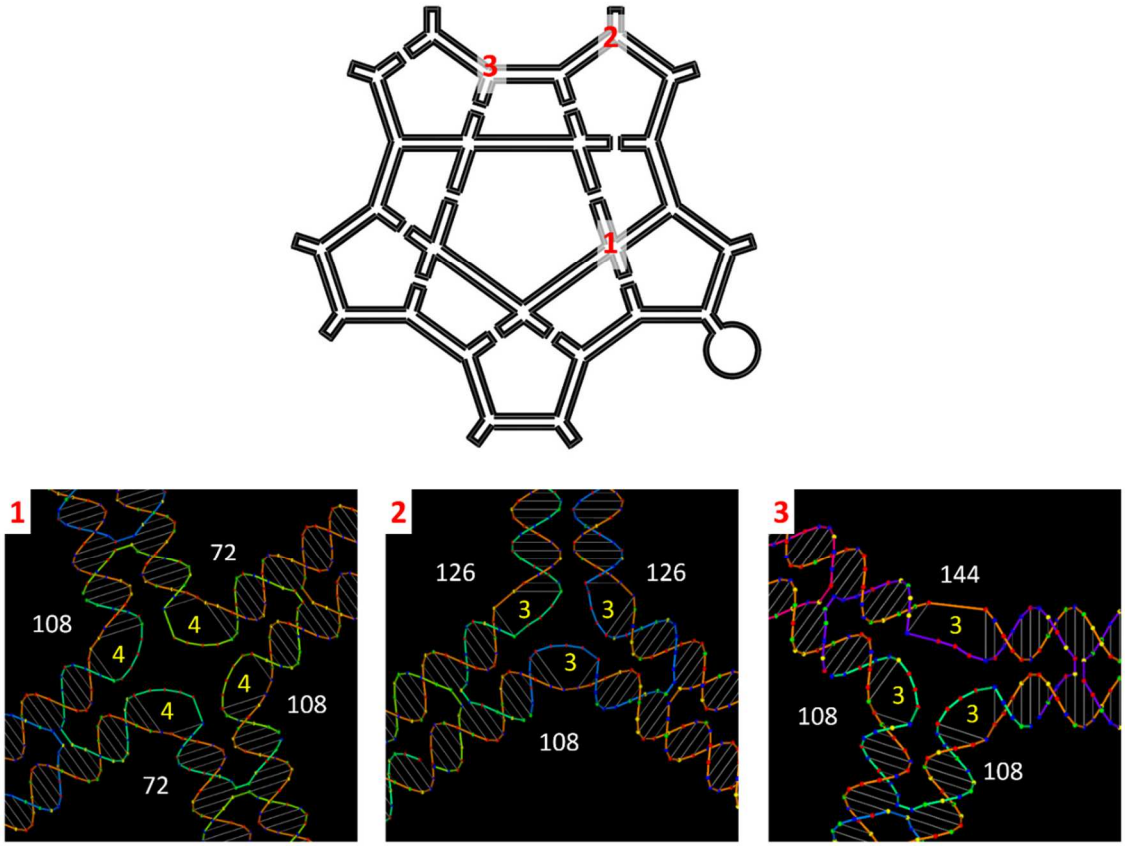


Fig. S22. Angle adjustment in the star shape. Three types of angles are employed to form the star-shaped origami. The angles are labeled in white, and the inserted Tn loops are marked in yellow. The star shape is made up of three different patterns.

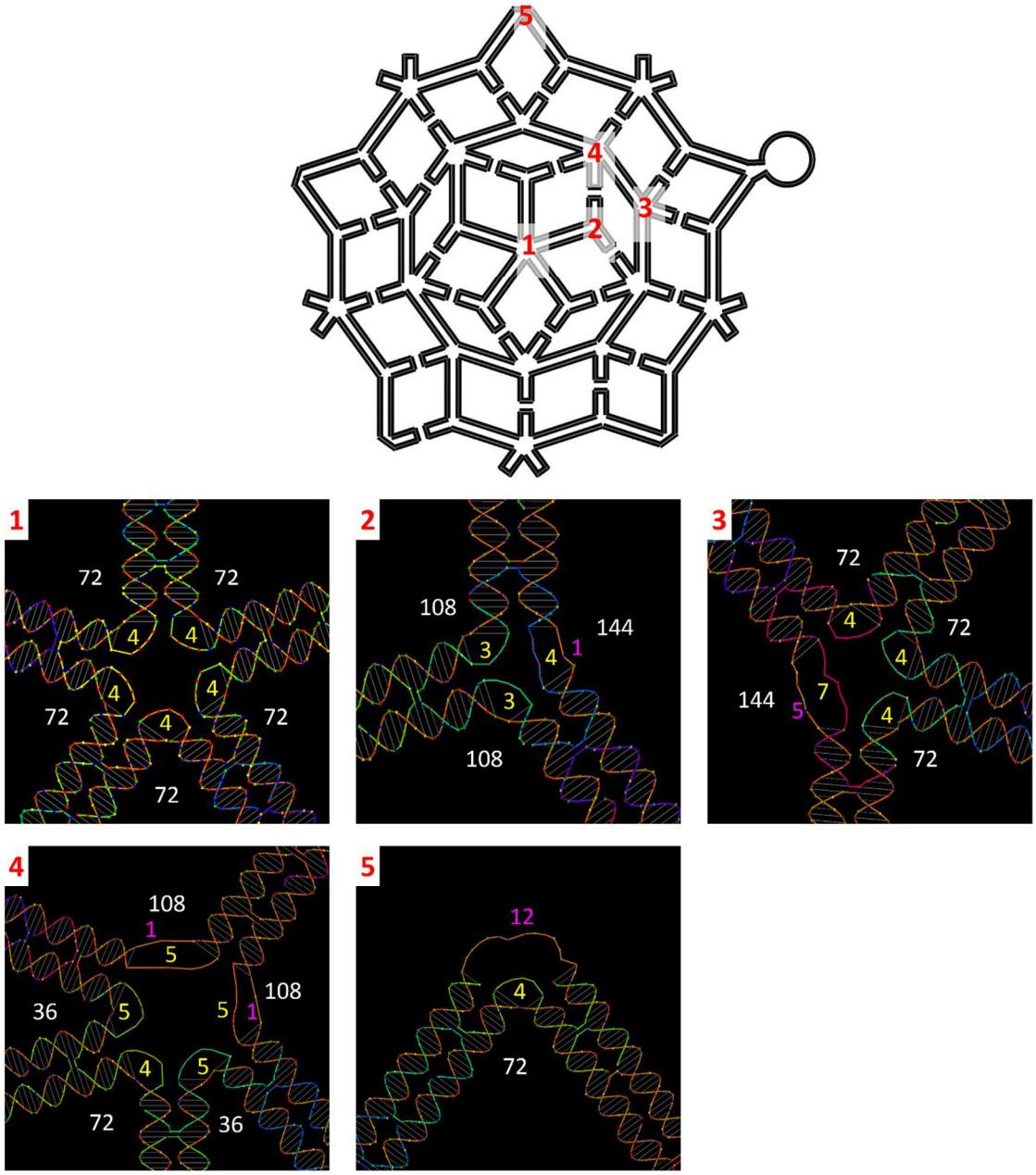


Fig. S23. Angle adjustment in the 2D Penrose tiling. The desired angles are labeled in white, and the Tn loops inserted in the staples are marked in yellow. The numbers of unpaired nucleotides in the scaffold are marked in pink. The Penrose tiling comprises five different patterns.

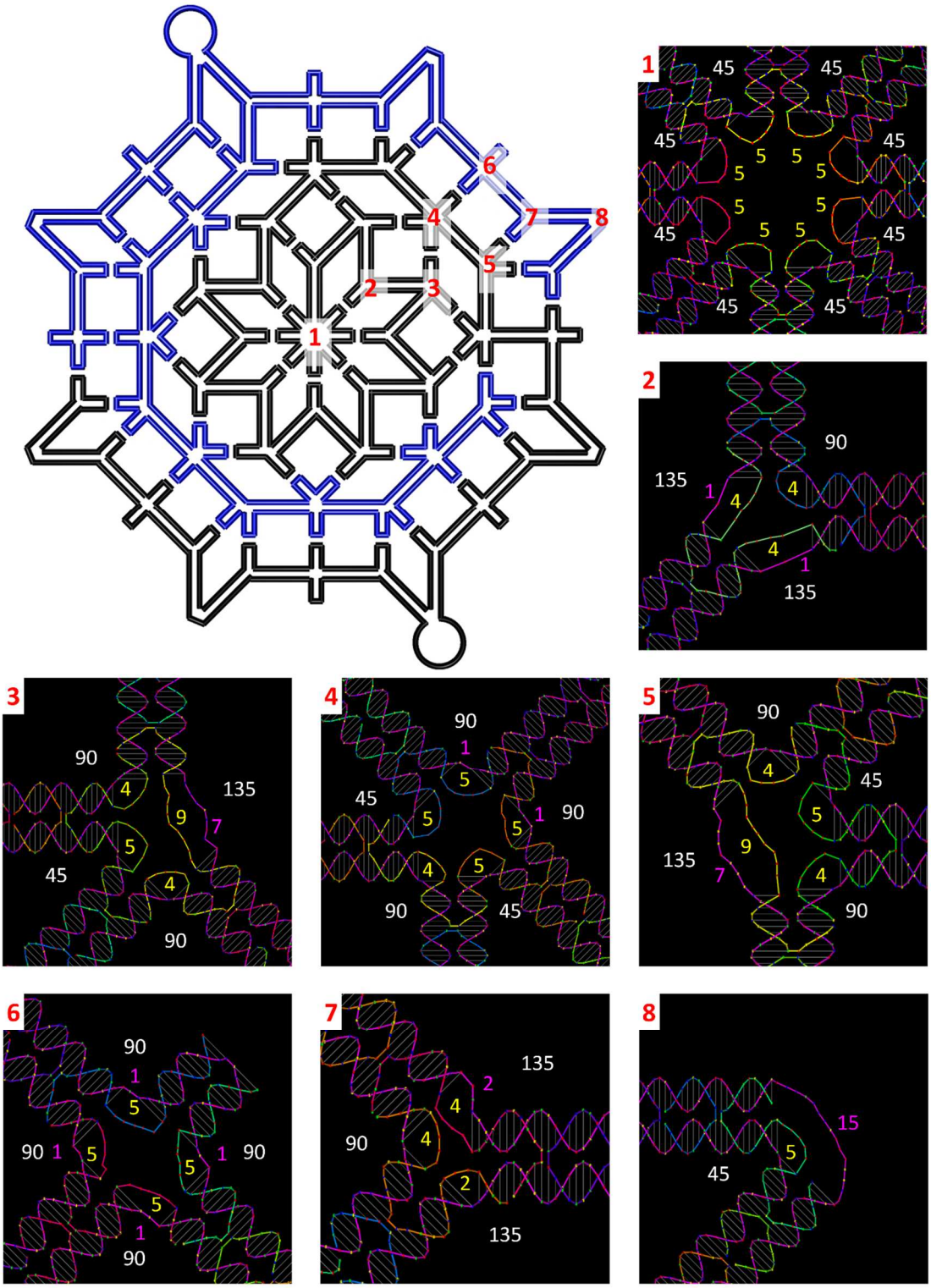


Fig. S24. Angle adjustment in the 8-fold quasicrystalline 2D pattern. The pattern includes eight different scenarios.

S1.3.2 Curvature design

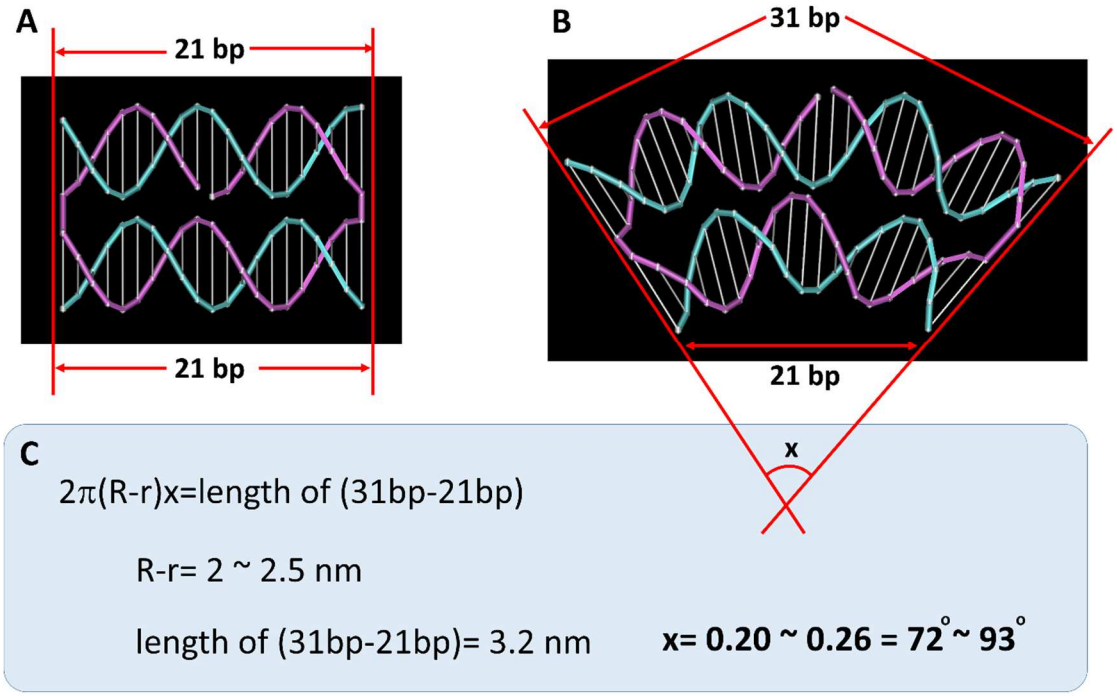


Fig. S25. Design of curvature unit: inserting base pairs. (**A**) Two antiparallel DNA double helixes of 21 base pairs (bp) are connected by double crossovers on both ends. (**B**) A curved unit was created by inserting 10 more bps on one of the helices. The angle of the arcs formed from bending the two helixes is labeled as x. (**C**) Estimates the angle x. The two DNA double helixes are treated as concentric arcs. The lengths of these two arcs are estimated from the lengths of straight 21-bp and 31-bp double helixes of B-form DNA (10.5 bp/3.4 nm), respectively. The distance between the centers of the two arcs is estimated as 2 to 2.5 nm. From this simplified calculation, the angle x is estimated to be in the range of 72º to 93º.

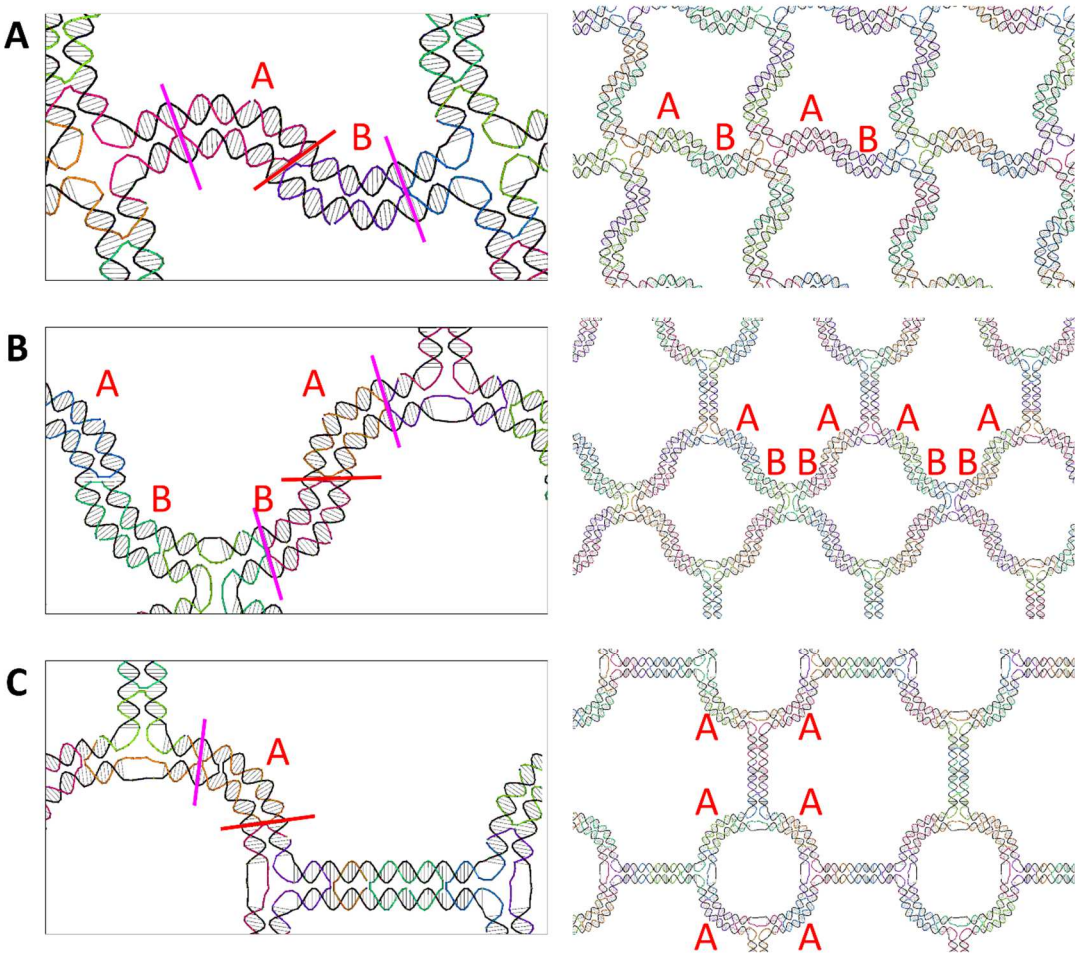


Fig. S26. Three curved structure design details. The curvature unit is located between the pink and red lines. **A.** The waving grid is constructed by connecting two opposite curvature units on each repeating edge (ABAB pattern). **B.** The fishnet pattern consists of two opposite curvature units and the alternate ordering of these two units on adjacent edges (ABBA pattern). **C.** The circle array contains only one type of curvature unit that is repeated in each unit cell (AAAA pattern).

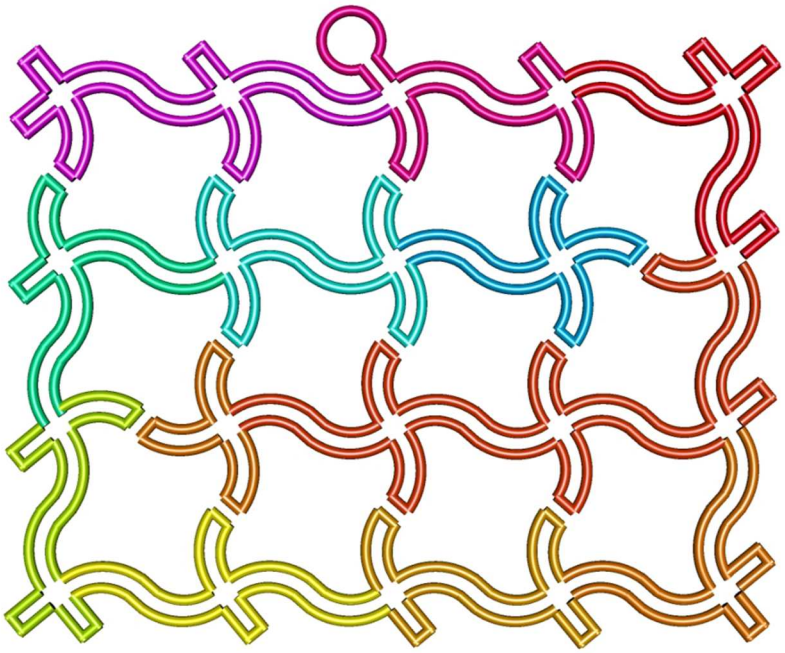


Fig. S27. Scaffold-folding path of the waving grid. The changing colors provide a visual guide showing how the scaffold travels.

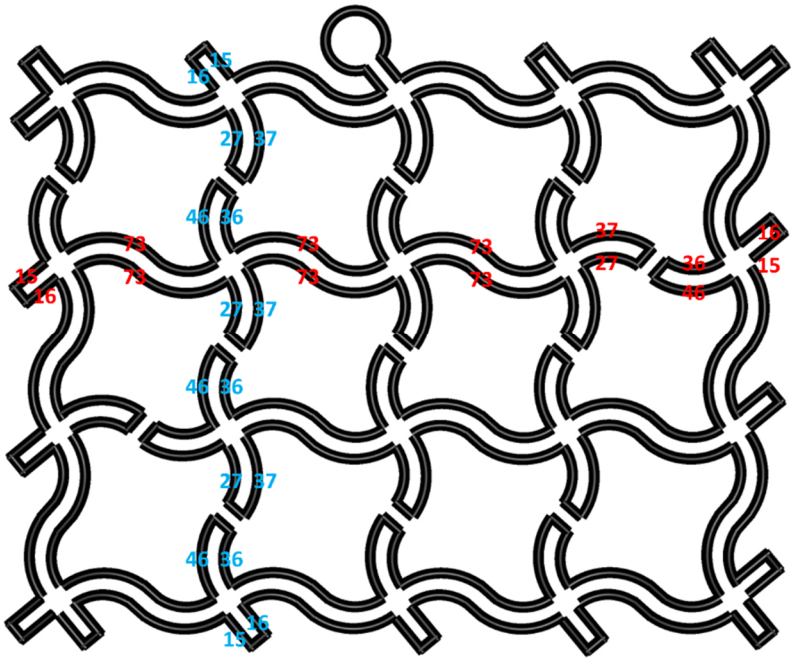


Fig. S28. Number of bases for the waving grid scaffold. The numbers refer to the distances (number of nucleotides) between the junction points.

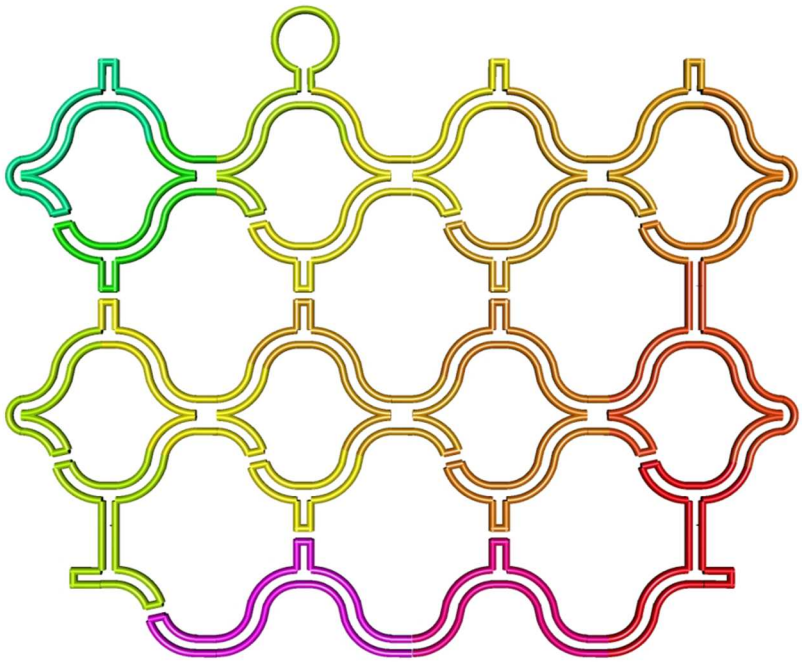


Fig. S29. Scaffold-folding path of the fishnet. The changing colors provide a visual guide showing how the scaffold travels.

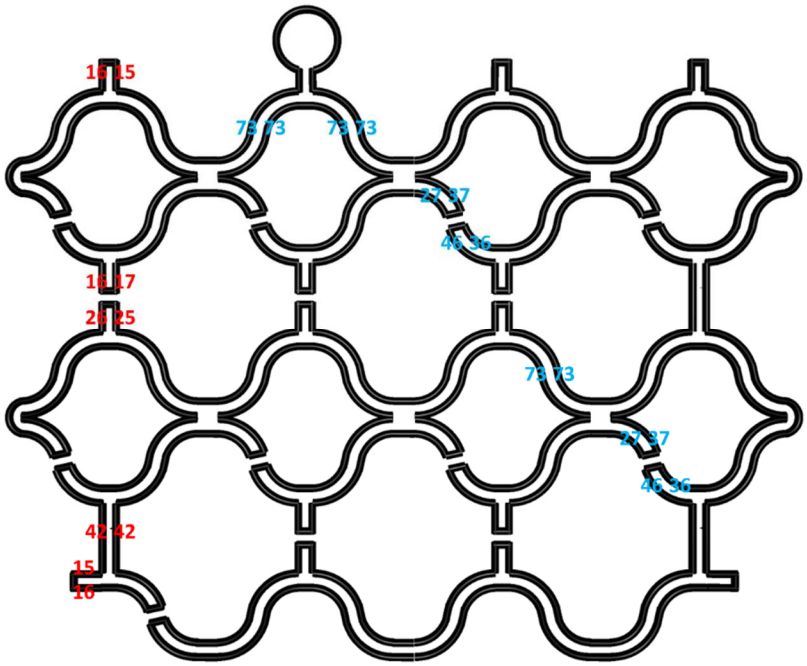


Fig. S30. Number of bases for the fishnet scaffold. The numbers refer to the distances (number of nucleotides) between the junction points.

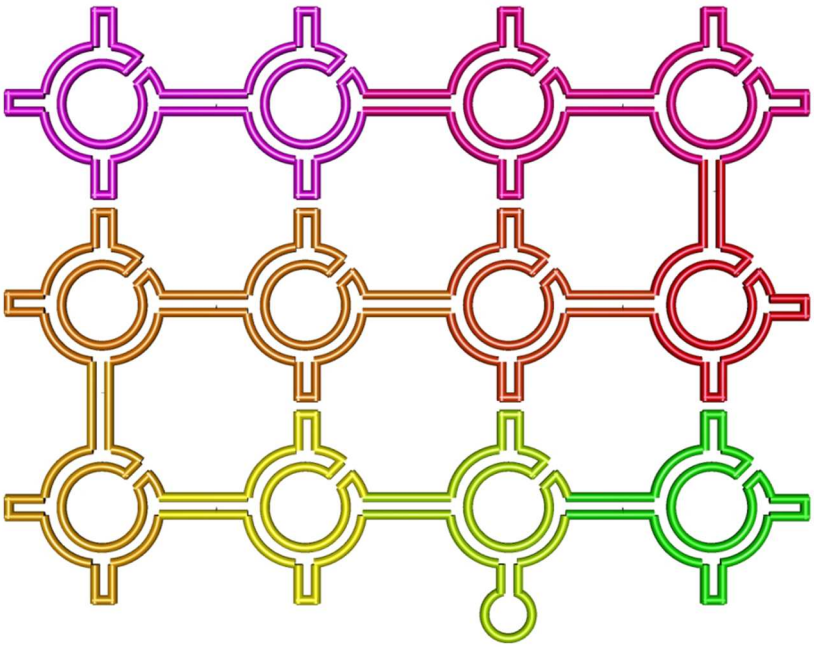


Fig. S31. Scaffold-folding path of the circle array. The changing colors provide a visual guide showing how the scaffold travels.

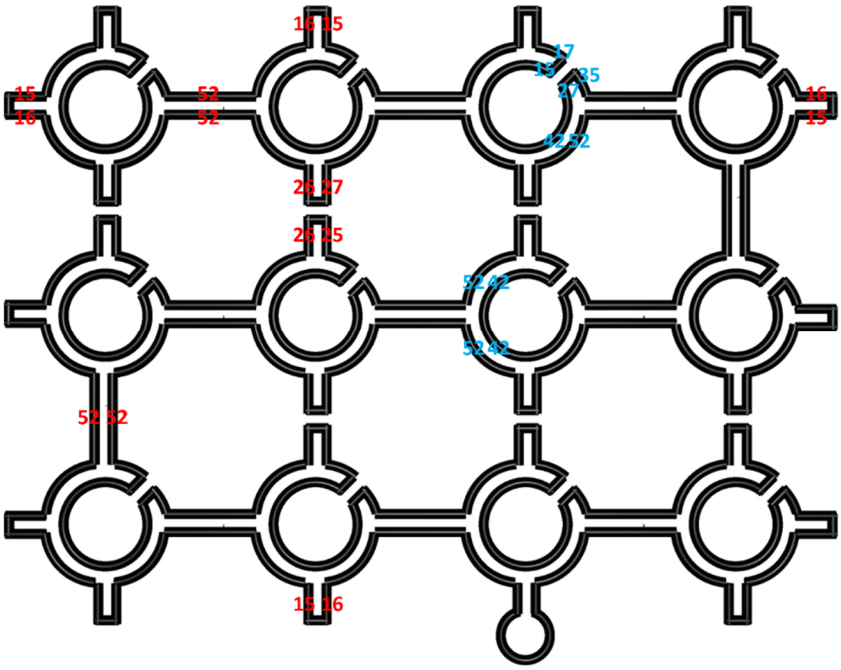


Fig. S32. Number of bases for the circle array scaffold. The numbers refer to the distances (number of nucleotides) between the junction points.

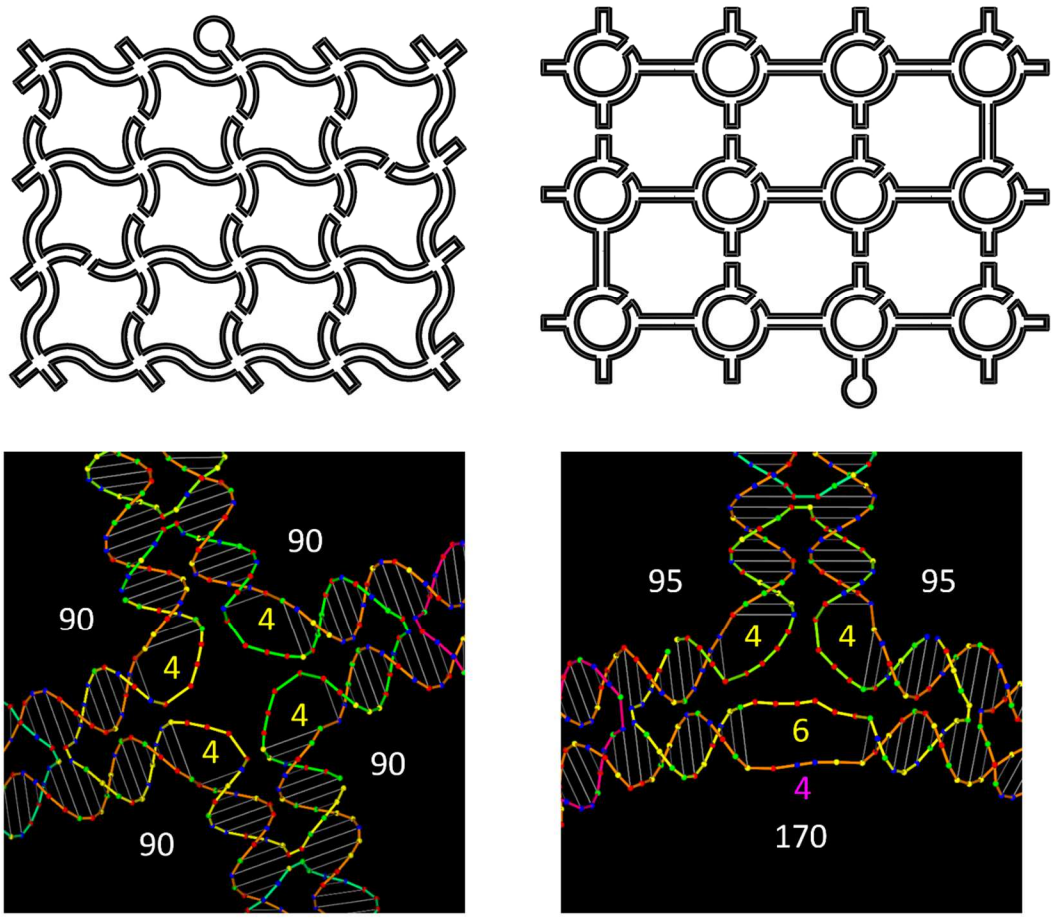


Fig. S33. Angle adjustment in the waving grid and circle array. Each vertex is equivalent in these two patterns.

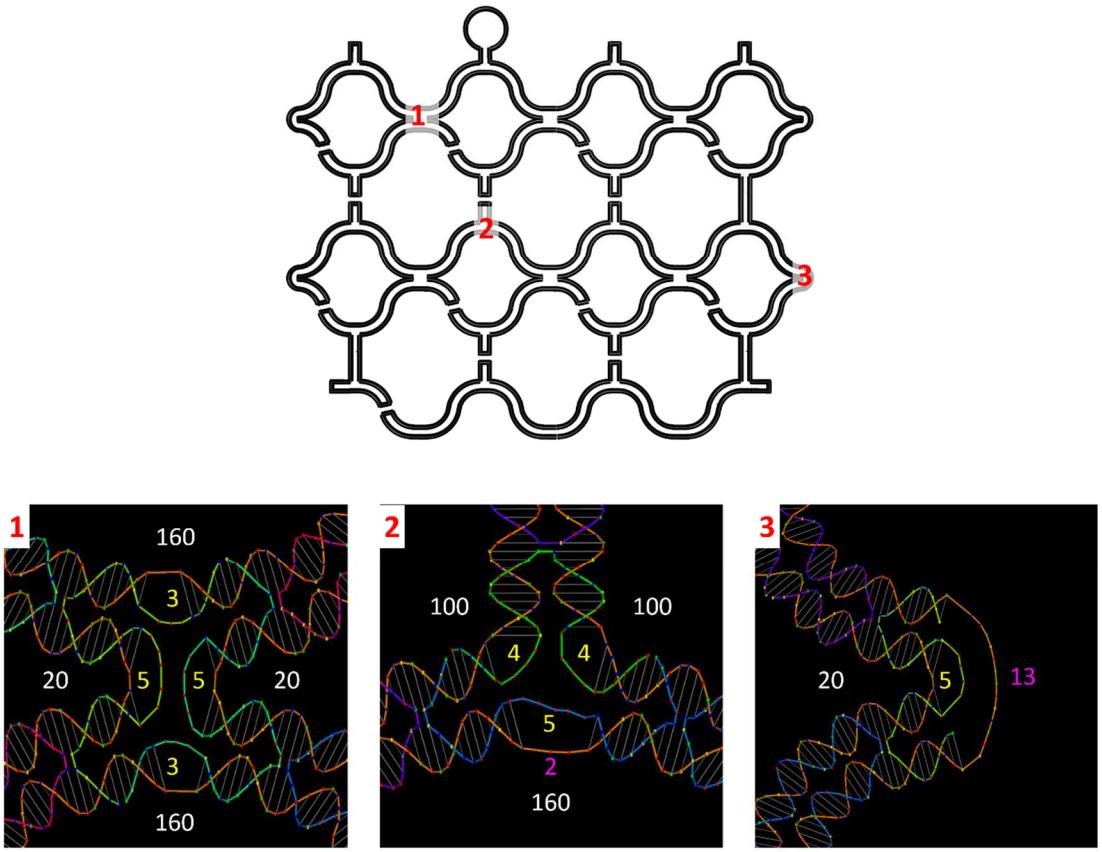


Fig. S34. Angle adjustment in the fishnet. The fishnet is made up of three different patterns.

S1.4 The flower-and-bird design

S1.4.1 Scaffold routing and angle adjustment

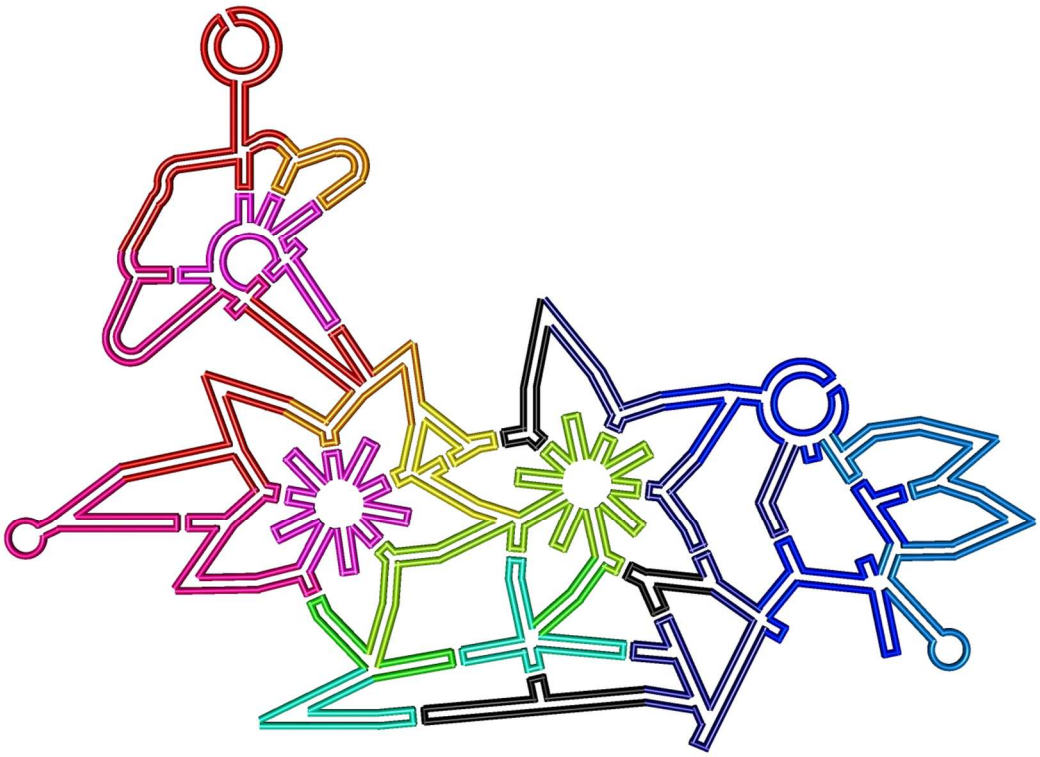


Fig. S35. Scaffold-folding path of the flower and bird. The loop on the left (colored in red, pink, yellow, green, and cyan) is the M13mp18 scaffold. The loop on the right (colored in black and blue) is the PhiX174 scaffold.

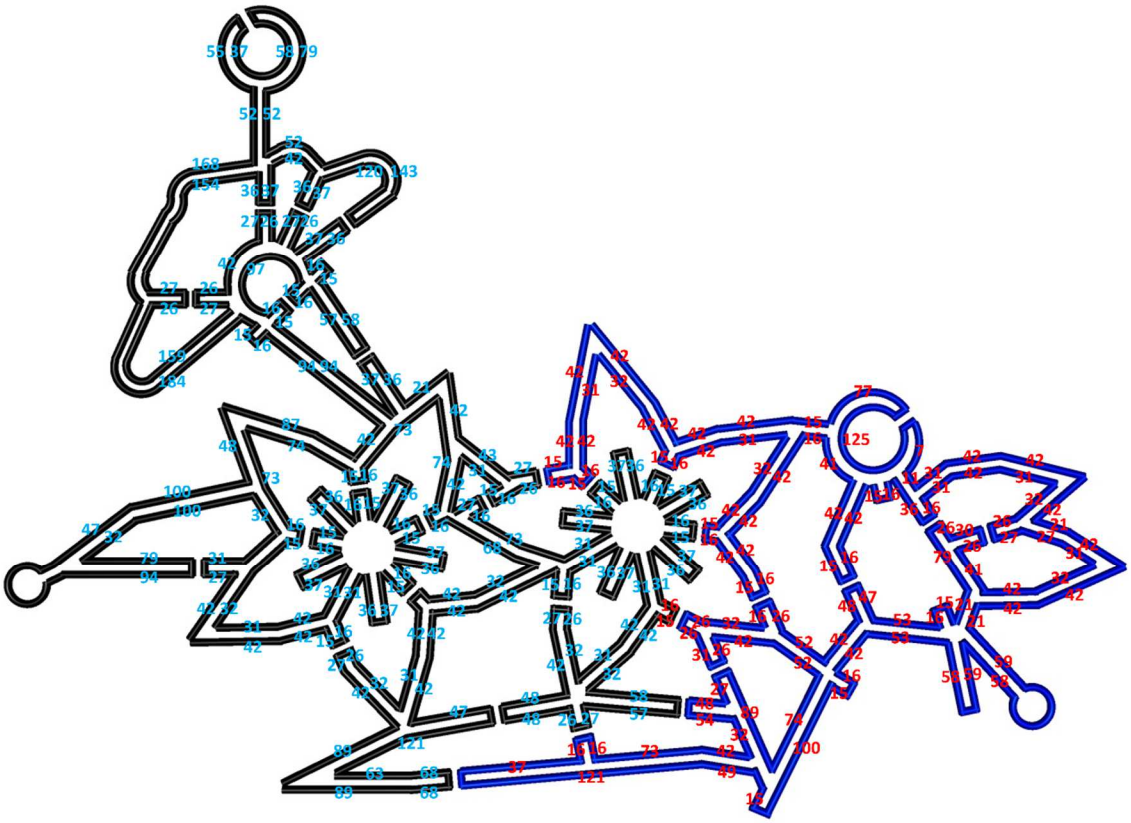
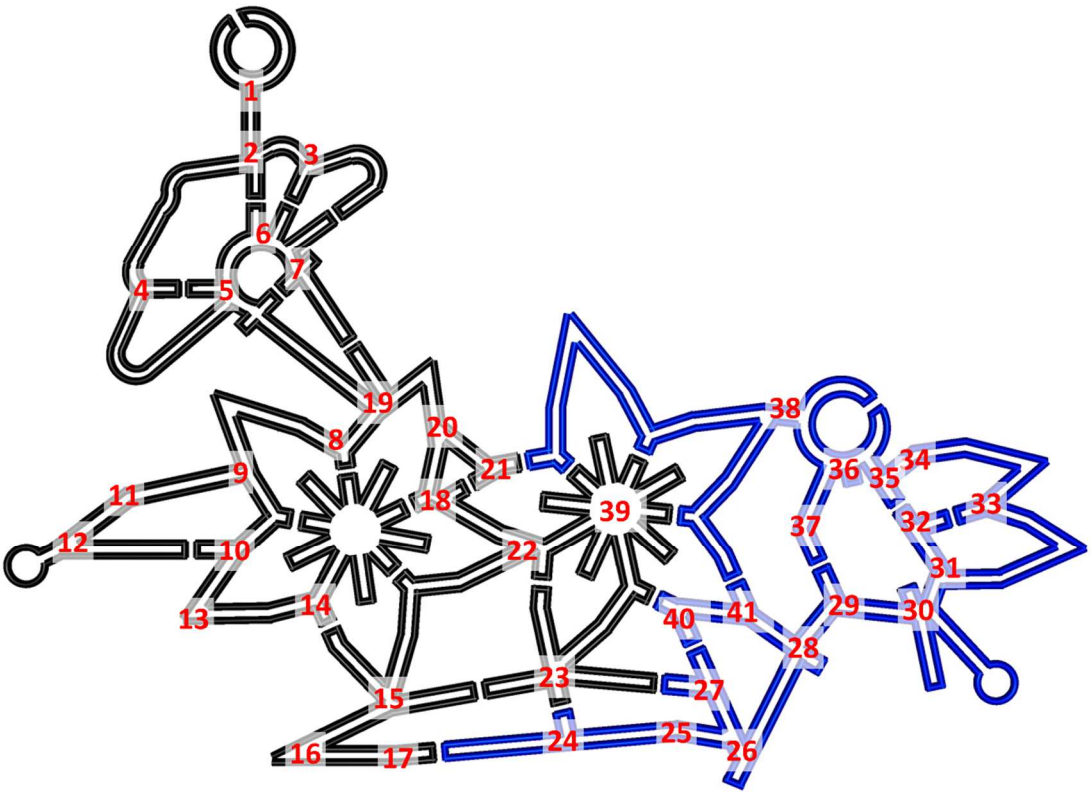
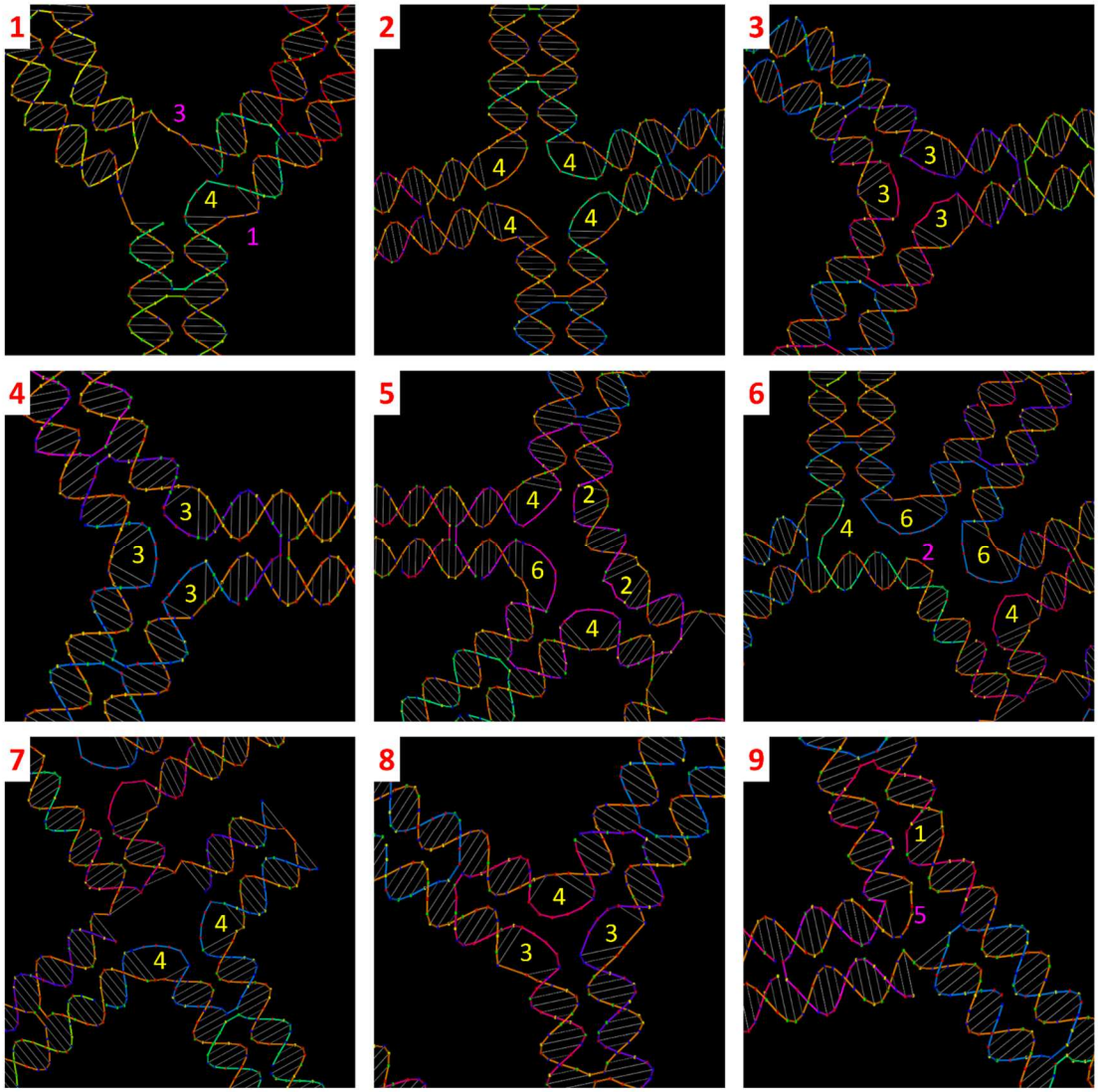
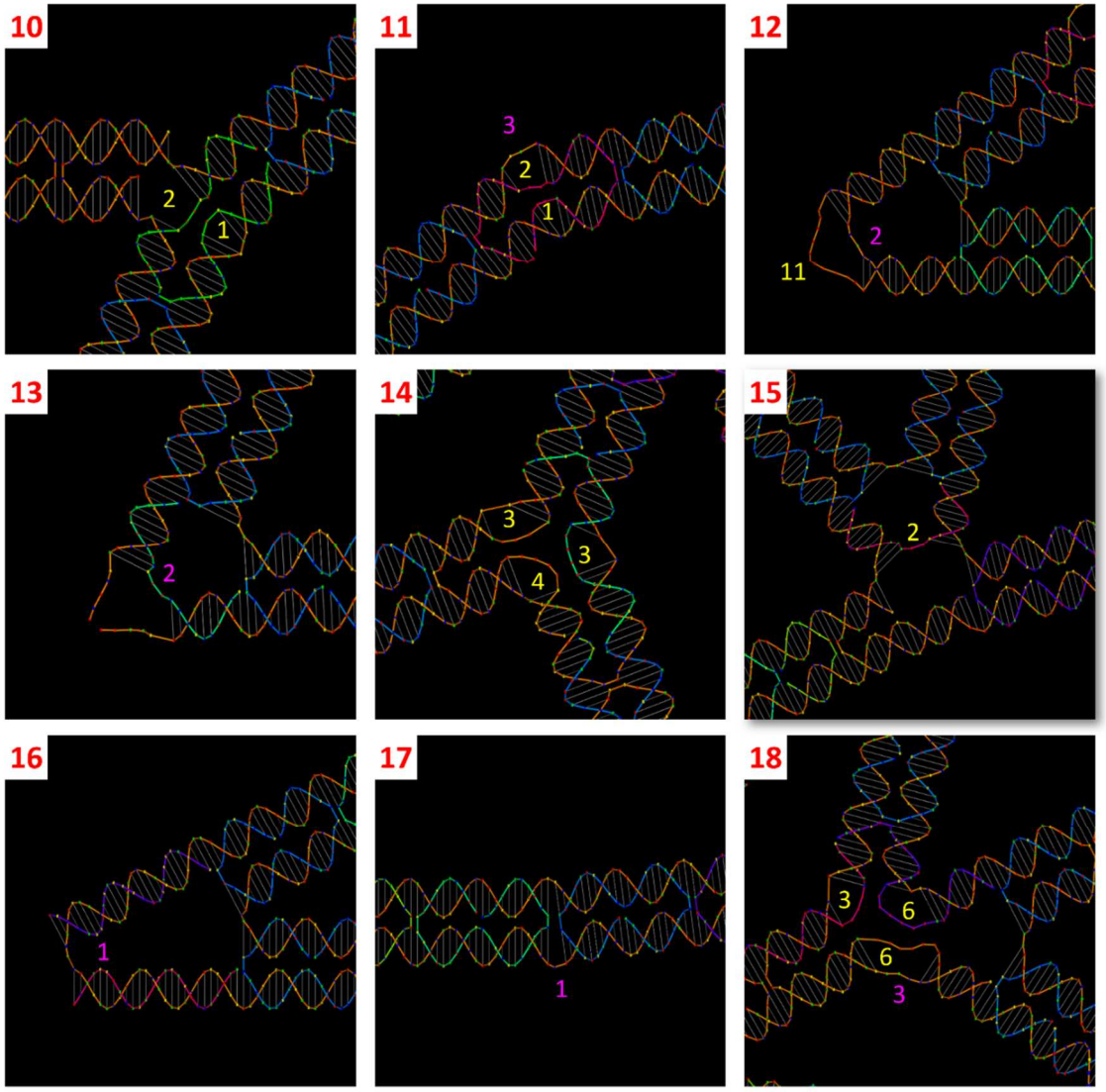
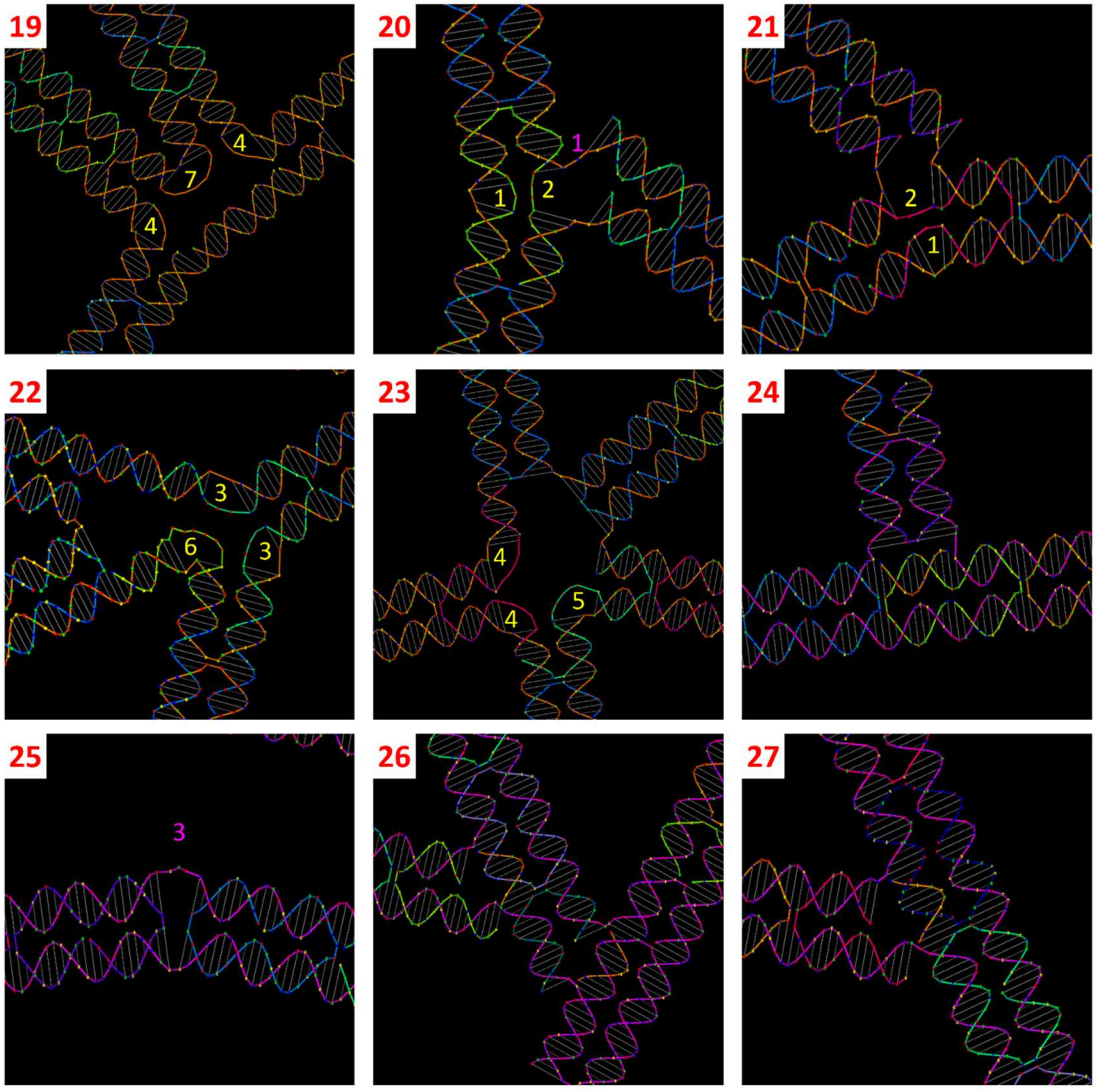


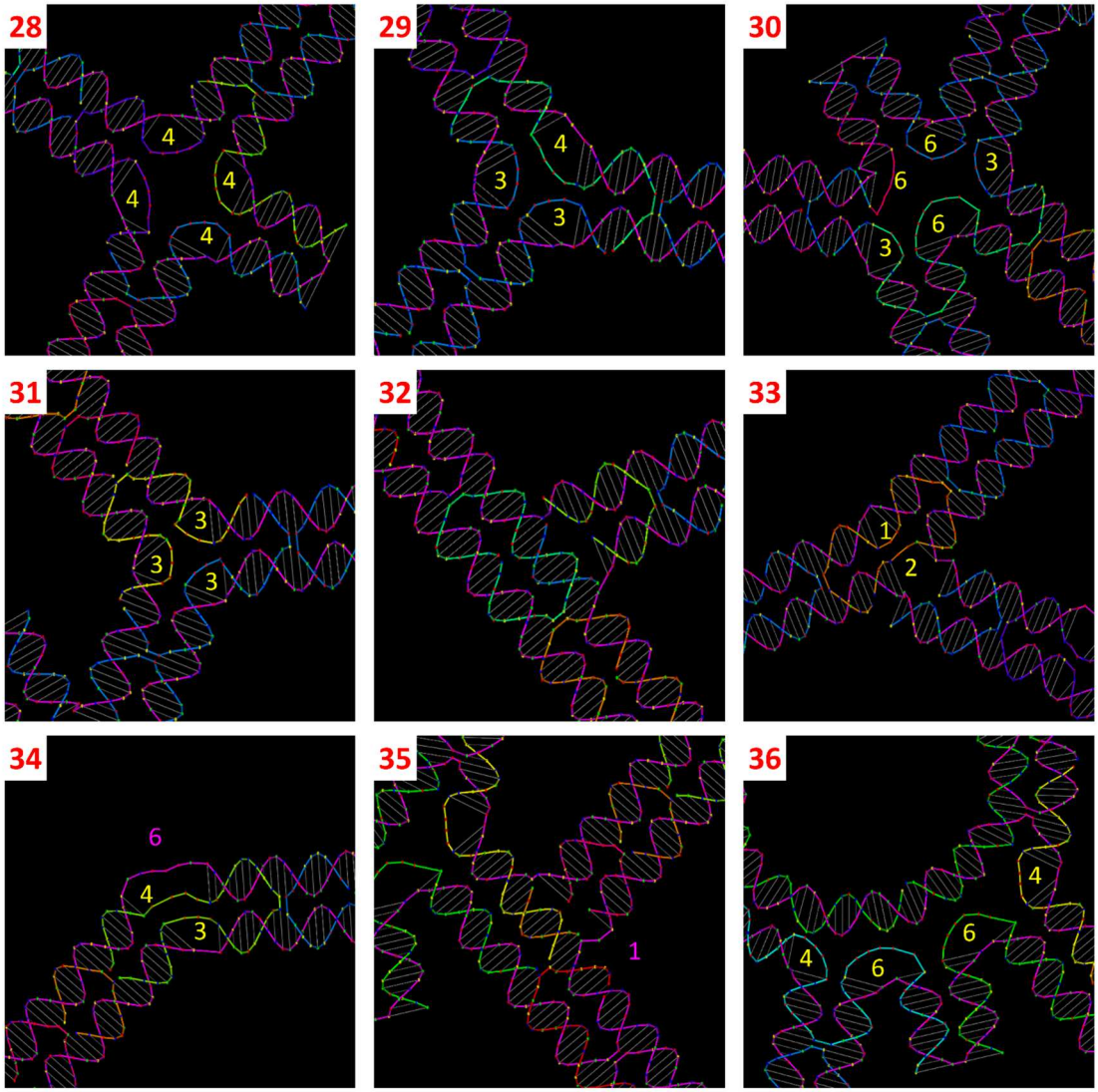
Fig. S36. Number of bases for the flower-and-bird scaffold. The loops in blue and black are the M13mp18 and PhiX174 scaffolds, respectively. The numbers refer to the distances (number of nucleotides) between the junction points. Each edge is individually adjusted to match the length required for the designed pattern.


1
2
3
6
7
4
5
19
20
8
21
9
11
18
39
12
10
22
13
14
15
23
16
17
24
38
36
35
34
37
32
33
31
29
30
40
41
28
27
25
26


1
3
4
1
2
4
4
4
4
3
3
3
3
4
3
3
3
5
4
2
6
2
4
6
4
6
2
6
4
7
4
4
8
4
3
3
9
1
5


10
2
1
11
3
2
1
12
2
11
13
2
14
3
3
4
15
2
16
1
17
1
18
3
6
6
3


19
4
7
4
20
1
1
2
21
2
1
22
3
6
3
23
4
4
5
24
25
3
26
27


28
4
4
4
4
29
4
3
3
30
6
3
6
6
3
31
3
3
3
32
33
1
2
34
6
4
3
35
1
36
4
6
4
6

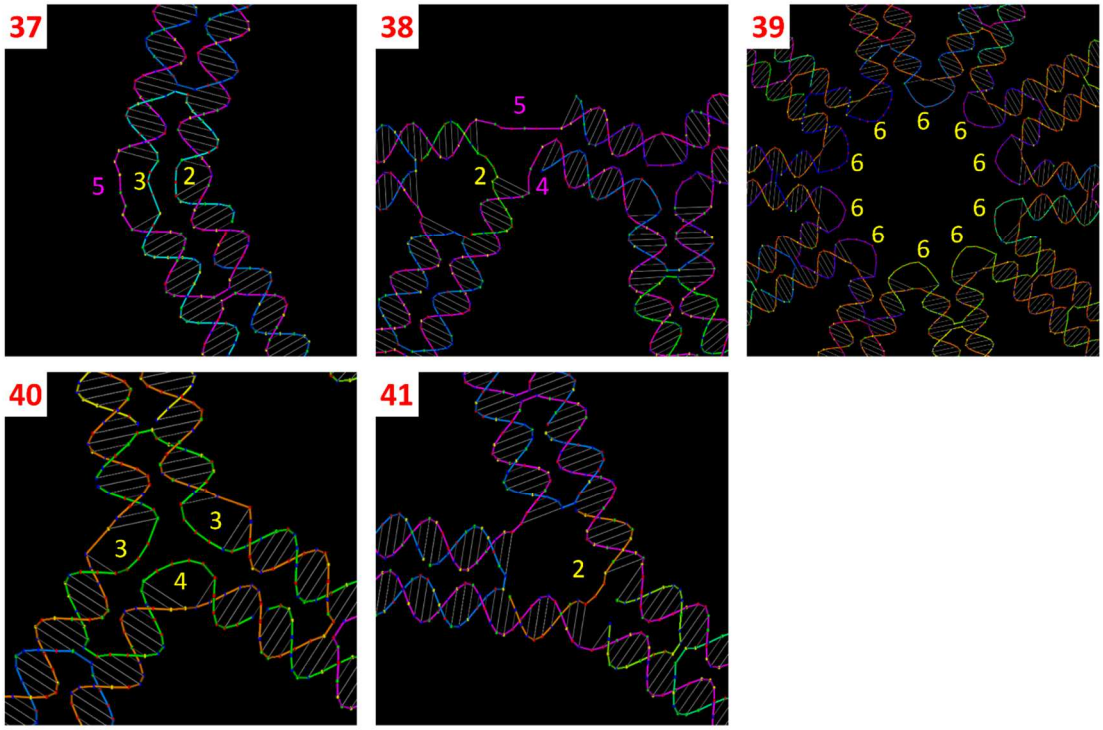


Fig. S37. Angle adjustment in the flower-and-bird. This arrangement involves 41 uniquely designed patterns.

S1.4.2 Summary of the general rules for angle adjustment

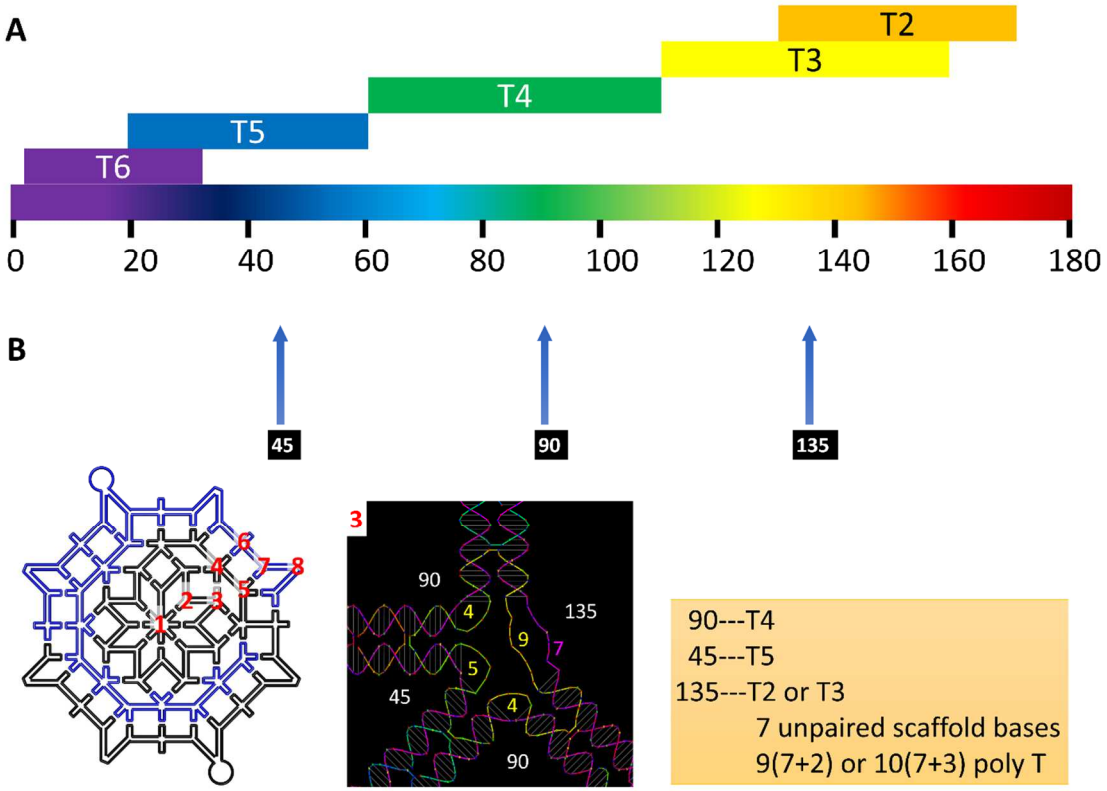


Fig. S38. Summary of the general rules for creating angles at any vertex. The number of T that should be inserted in the staple strand is determined by the angle between two adjacent arms and the length of the unpaired DNA scaffold loop opposite to the T loop. (**A**) If there is no unpaired nucleotides on the scaffold, the relationship between the number of T and the angle is shown in the ruler, which shows a rough reverse proportional relationship (i.e., the shorter the Tn loop, the larger the angle it affords). (**B**). Take the 8-fold quasicrystal pattern as an example. The vertex labeled **3** consists of five angles (one 45°, two 90°, and one 135°). For the 45° and two 90° angles, the adjacent arms are close in space, so that the scaffold strand was left with no gap. The number of Ts inserted can be found according to the ruler in **A**: T5 for the 45° and T4 for the 90°. However, the two arms that form the 135° angle are separated by a certain distance, where seven unpaired bases are left on the scaffold strand to create a gap. Because 135° is related to T2 or T3, the number of T that should be inserted in the opposite staple strand is estimated as 7 plus 2 or 3. Nine is used here.

S1.5 3D wireframe DNA origami

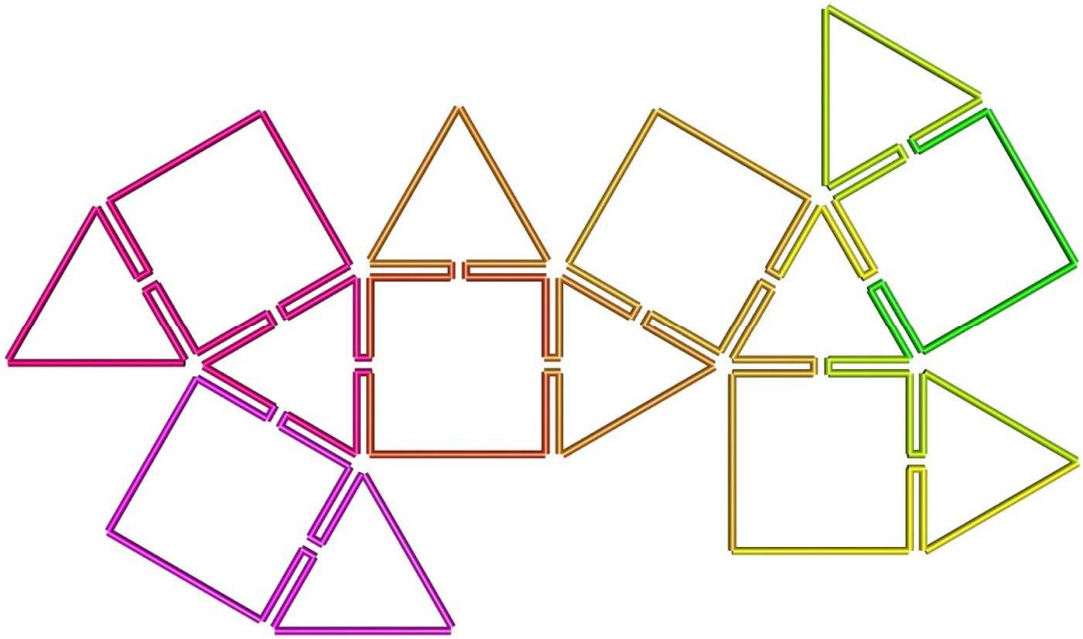


Fig. S39. Scaffold-folding path of the 3D cuboctahedron. The changing colors provide a visual guide showing how the scaffold travels.

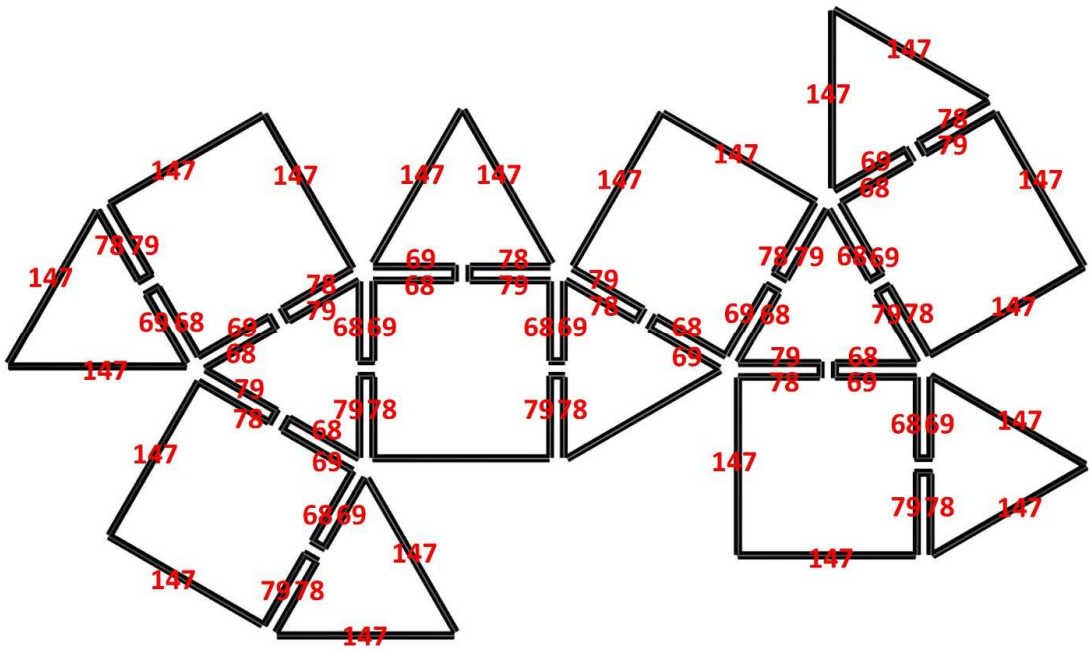


Fig. S40. Number of bases for the 3D cuboctahedron.

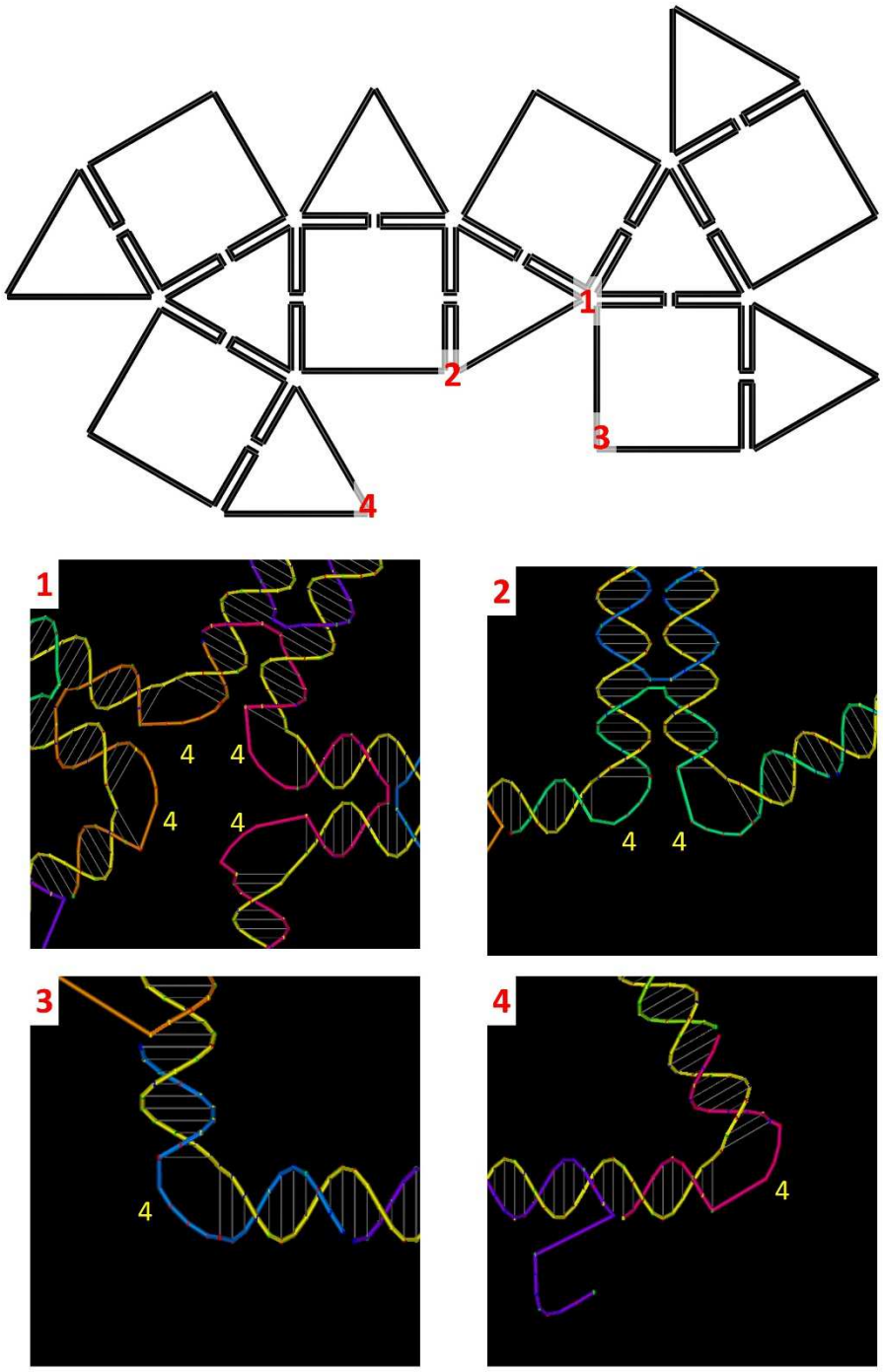


Fig. S41. Angle adjustment in the 3D cuboctahedron. There are four different scenarios.

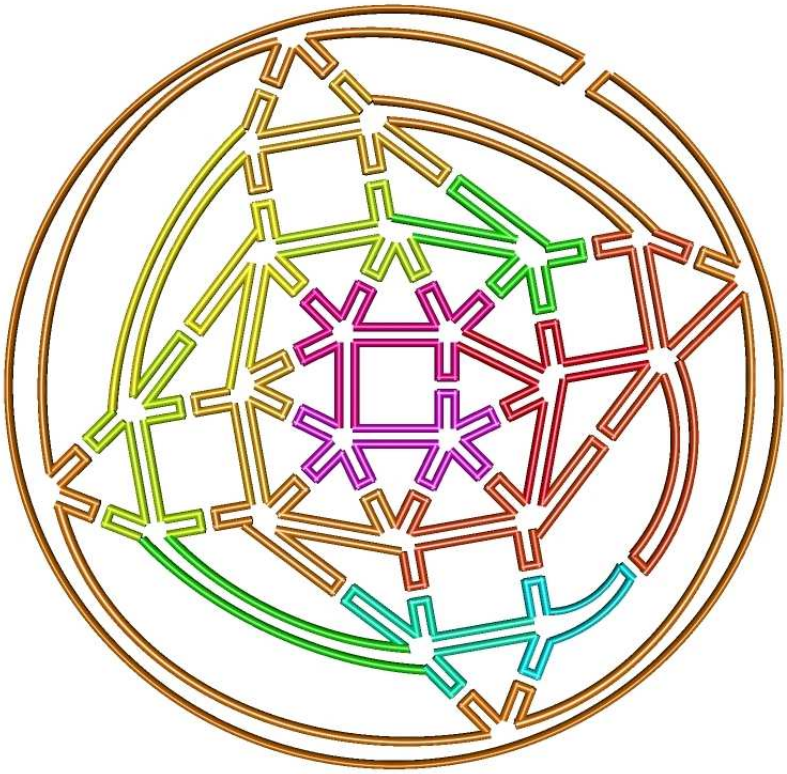


Fig. S42. Scaffold-folding path of the 3D snub cube. The changing colors provide a visual guide showing how the scaffold travels.

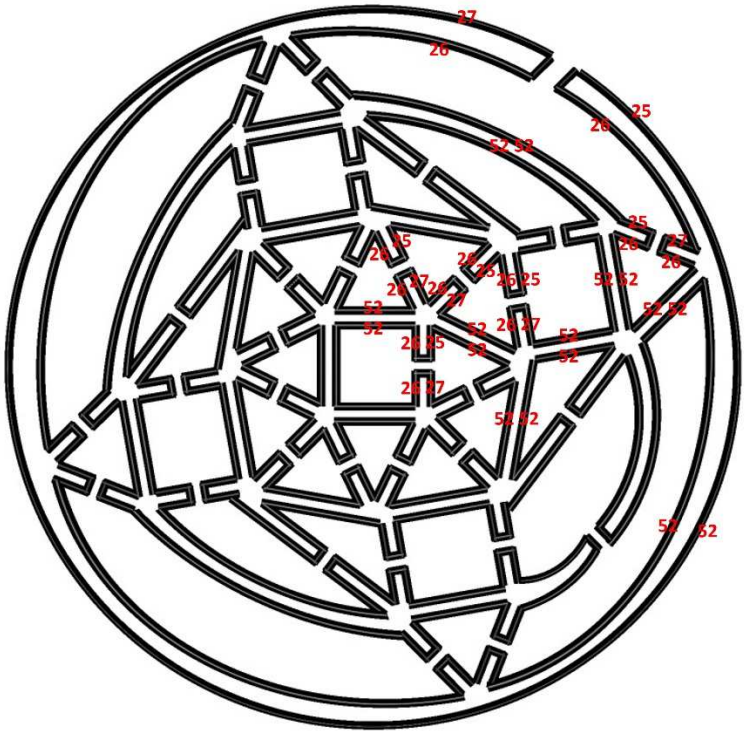


Fig. S43. Number of bases for the 3D snub cube.

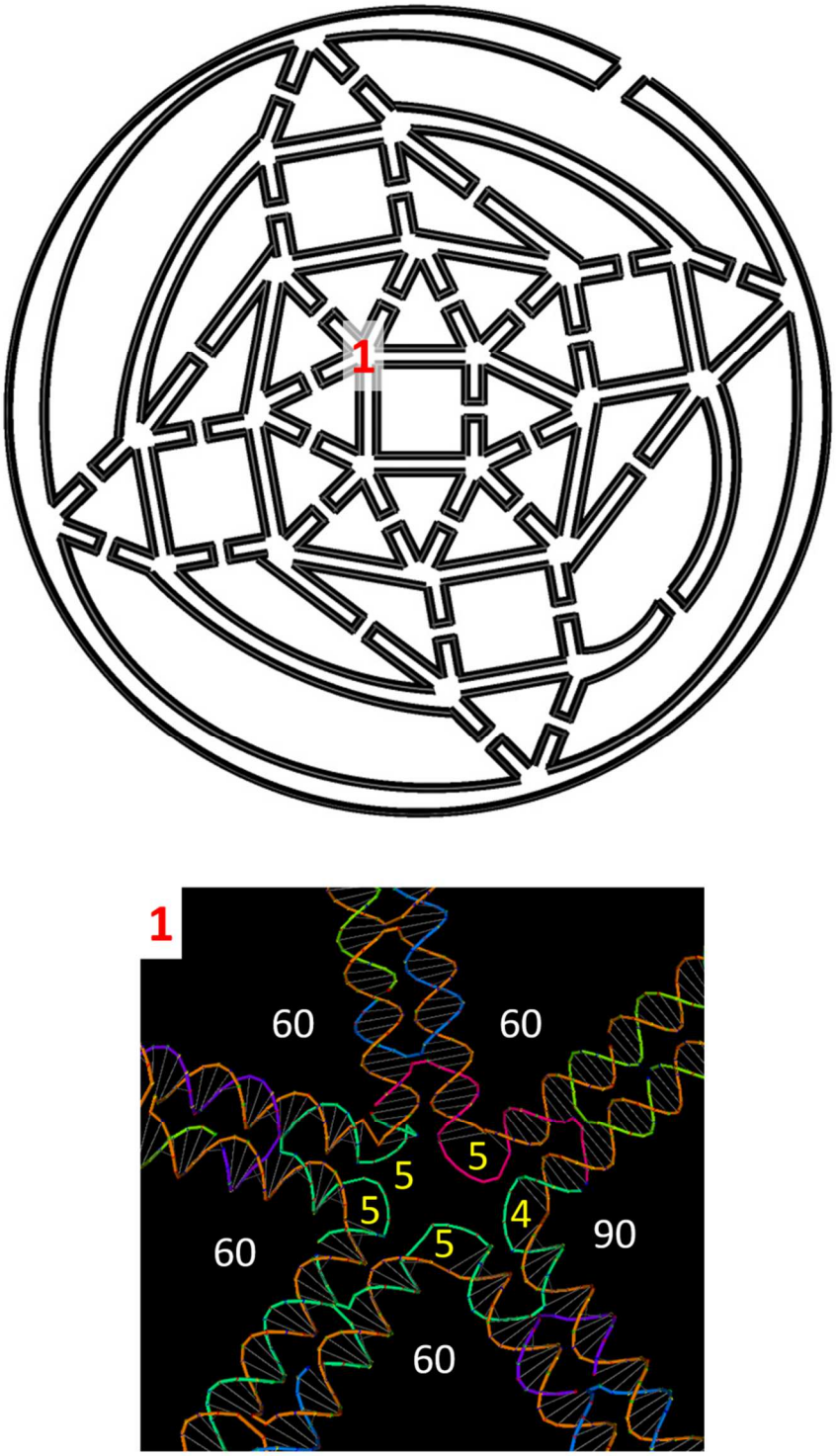


Fig. S44. Angle adjustment in the 3D snub cube.

Section 2. Yield analysis

The yields of DNA frame origami were estimated by gel image analyses of native agarose gel electrophoresis. The samples were prestained with SYBR gold nucleic acids gel stain (see details in materials and methods). The yield of one structure was estimated by comparing the fluorescence intensity of the target band and the entire lane. For a better estimation, background intensity was subtracted from both the measured intensity of the target band and the entire lane. The fluorescence intensity of the entire lane consists of the band of the well-formed target structure and any upper bands representing the unwanted aggregates (oligomers). The intensity of the extra staple strands in the bottom of each lane was not counted in the entire lane. ImageJ software (NIH) was used to analyze the gel images. Figure S39 demonstrates one example of this assay. The red rectangle outlines the area that refers to the intensity of the target band. The green rectangle contains the entire lane, and the blue rectangle contains the background. The yield of DNA frame origami was calculated as:

$$Percentage\ Yield = \frac{Intensity\ of\ target\ band}{Intensity\ of\ entire\ lane} \times 100\%$$

The background was subtracted as:

$$Intensity\ of\ target\ band = (Intensity\ mean\ value\ of\ target\ band - Intensity\ mean\ value\ of\ backgroud) \times\ area\ of\ target\ band$$

$$Intensity\ of\ entire\ lane = (Intensity\ mean\ value\ of\ entire\ lane - Intensity\ mean\ value\ of\ backgroud) \times\ area\ of\ entire\ lane$$

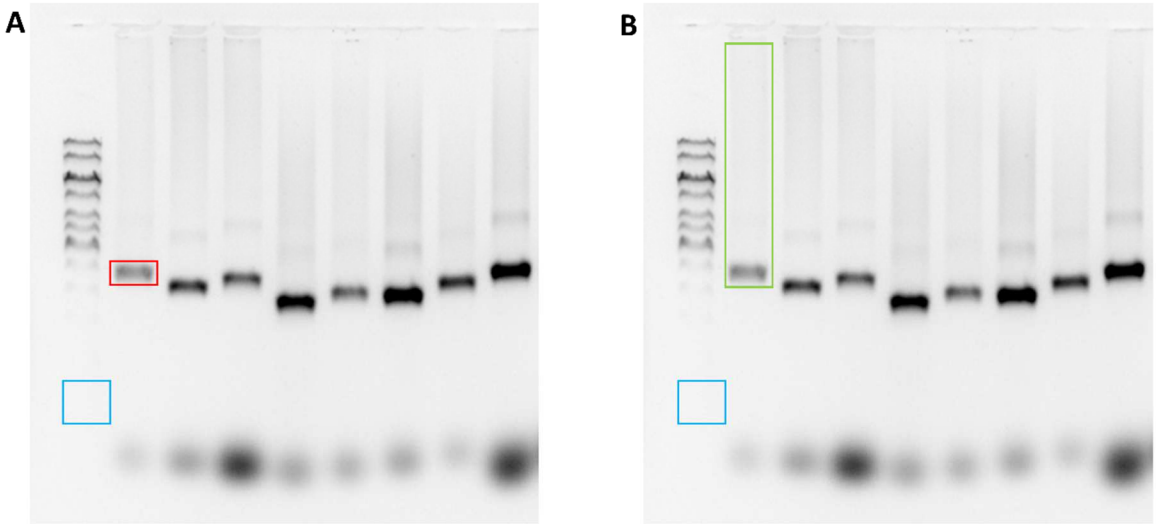


Fig. S45. An example of assembly yield quantification based on fluorescence intensity in the gel image. (**A**) The red rectangle outlines the target band. (**B**) The green rectangle encompasses an entire lane, and the blue rectangles indicate the background in both A and B.

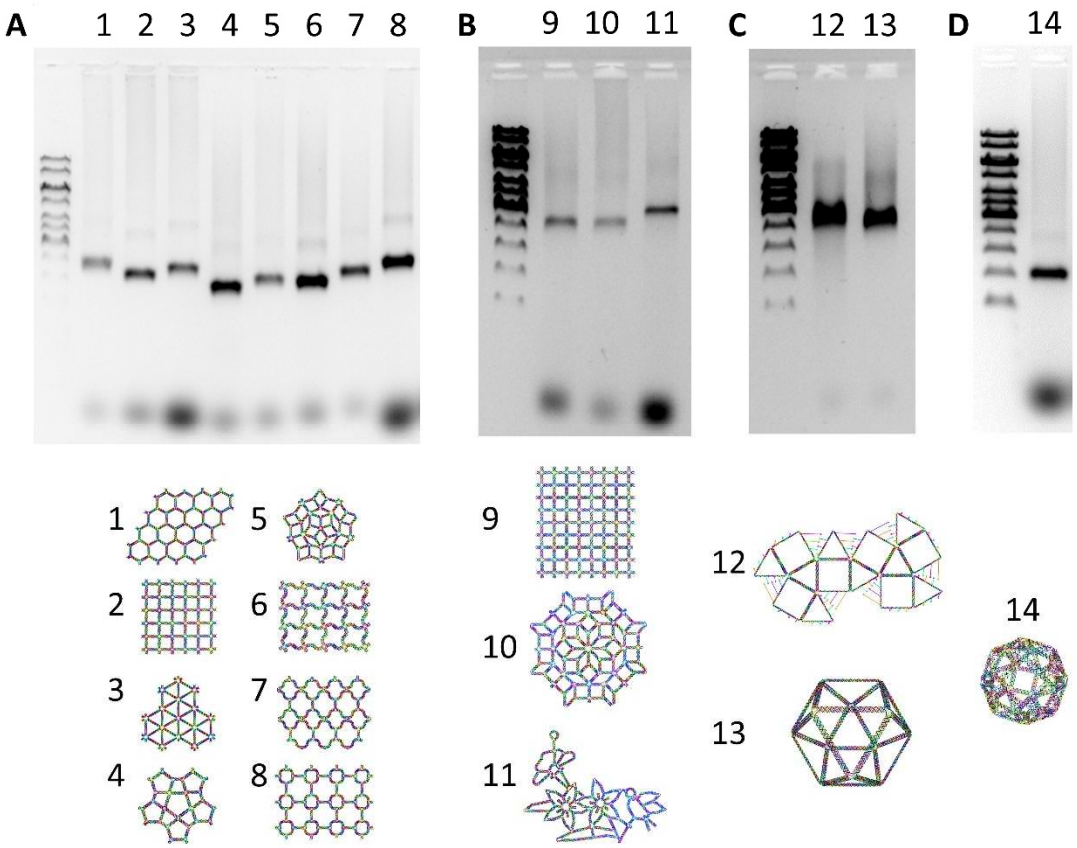
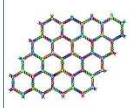
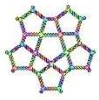
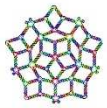
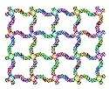
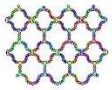
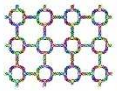


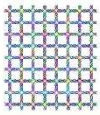
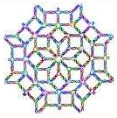
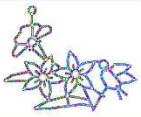
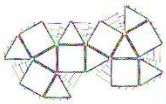
Fig. S46. Gel images for the native agarose gel electrophoresis. Samples were annealed individually and subjected to native agarose gel electrophoresis (see experimental details in materials and methods). The target bands are shown as the main bands in the middle of the gels. The big blots at the bottom of each lane contain the extra staple strands. A 1-kb DNA double-stranded ladder was added in the first lane of each gel. (**A)** The 2D wireframe origami with one scaffold. The corresponding patterns are labeled for each lane. The molar ratios of the scaffold to the staple strands were 1:2 or 1:5. (**B**) The 2D frame origami consisting of two scaffolds (from left to right): 9 × 8 square Platonic tiling, 8-fold quasicrystalline pattern and flower-and-bird pattern. (**C**) The 2D open net and 3D closed structures of the cuboctahedron. (**D**) The 3D snub cube structure.

Additional notes: We obtained the yields of each frame origami from the gel images shown in figs. S42 and 43 and summarized in Table S1. The 2D structures with only one scaffold generally had a decent assembly yield ranging from 71% to 87%. The upper faint bands and smeared upper bands represent the dimer structures and the other

unwanted aggregates, respectively. The 2D frame origami consisting of two scaffolds had lower yields than those with one scaffold (62-73%). They formed more aggregates than the structure with one scaffold. One possible reason for the lower yield is that the staple strands designed to link two scaffolds could also bring three or more scaffolds together. Additionally, the two-scaffold structures employed a larger number of staple strands (around 300) than the one-scaffold origami (normally ~200 helper strands or less). This could increase the possible mismatch. The higher-order assembly of frame-origami to form the 3 × 3 array gave the lowest assembly yield (23%) estimated from its AFM images (fig. S53). This is anticipated because the hierarchical assembly of origami, which has a large molecular weight and is living in a complex system with thousands of different bases, is more difficult than just folding origami itself and generally presents a low yield.

Table S1. Estimated yield of frame origami from agarose gels (except 3 × 3 array) and AFM images (only 3 × 3 array).

	2D Structures							
	One scaffold							
								
Yield	86%	76%	74%	76%	76%	71%	87%	81%

2D Structures				**3D Structures**		
Two scaffolds			3x3 array	2D open	3D closed	Snub cube
						
64%	62%	73%	23%	85%	70%	80%

Section 3. Curvature and chirality

In the previous tile self-assembly work, the overall curvature accumulated from every n-by-4 junction has been discussed since 2003 (H. Yan *et al.*, *Science* 301, 1882-1884, 2003). It keeps drawing researcher's interests, trying to answer certain questions: For example, how to tune the bending degree of the tile array to form tubes, and how to decrease the curvature to obtain more extended 2D structures (*Y. Ke, Y. Liu, J. Zhang, H. Yan, J. Am. Chem. Soc. 128, 4414-4421, 2006*). One effective strategy used to overcome the inherent bending or twist of each individual tile is the corrugated design. Integer-and-a-half turn B-form DNA was required between two adjacent junctions on the two connecting tiles to ensure the neighboring tiles are facing opposite directions to cancel the intrinsic curvatures.

In our wireframe origami design, we used integer turns as the length of each edge to facilitate routing of the scaffold strand. Local curvatures in the 2D wireframe origami structures were anticipated to exist. However, from our own observations, the bending of the overall structure was more obvious in the simpler cases of the wireframe origami that we first demonstrated rather than the more complicated patterns designed later. As we created more complex structures by our wireframe origami method, bending away from the 2D n × 4 junction may not accumulate as much as the highly symmetric patterns. With the presence of single-stranded loops, the scaffold strand, and unpaired bases in the staple in an asymmetric pattern, the bending or twisting tension would not build up in any one direction. The curvatures from different junctions can be neutralized by each other in certain structures.

For example, the simple star-shaped 2D wireframe origami could not be imaged clearly under AFM without adding $NiCl_2$, which may help the structure attach more firmly to the mica (figs. S44D-F). The star-shaped structure has a 5-fold symmetry, and we also designed the junctions in this pattern with the same symmetry. So, a possible curvature in this shape may bend the whole structure into a domed surface. Under AFM, we did observe that this star-shape structure presented blurred edges without $NiCl_2$ (see fig. S47A-C), indicating that the center part of this shape was attached to the mica surface, but the edges curved away from the substrate.

Another important aspect about curvature in the wireframe origami is that we can tune the rigidity of each junction by adjusting the number of unpaired nucleic acids in the junction. Because we need to consider the angle adjustment in the junction (as discussed in supplemental section 1), the number of unpaired bases left in the scaffold and the length of the poly T loops inserted in the staples should be chosen carefully. We summarized the general rule in fig. S38 that contains some overlapping regions to create the same angle. This means that one may choose to use a different number of poly T to satisfy one specific degree angle. The different choices will affect the rigidity of this junction. The star-shaped structure is an example of wireframe origami with all rigid junctions. The angle with 108º, 108º, and 144º (angle number 2 in fig. S22) was assigned with three T3 without any unpaired scaffold. Three same length poly Ts exist with 3-fold symmetry, which best matches the 3-fold symmetry junction (with three 120º angles). So, the tension in the 144º angle (with a +24º deviation from 120º) is the largest when compared with the other two 108º angles (with a -12º deviation from 120º). While the same angle appeared in Penrose tiling (angle number 2 in fig. S23), it was arranged with

two T3 for the 108º angles and one T4 with one unpaired nucleotide on the scaffold for the 144º angle. The extra T and base on the scaffold would release the tension in the 144º angle in this junction. A flatter structure was observed under AFM for the Penrose structure as it has the more flexible junctions than the star-shaped structure (fig. S48).

In our wireframe DNA origami, we two designed chiral patterns: the waving grid and the flower-and-bird pattern. When deposited on the mica surface, two conformations with different chiralities were observed under AFM (for waving grid see fig. S54, and for flower-and-bird see fig. S57). We counted the number of each conformation for both structures. They all exist in a preferred conformation when deposited. The waving grid had 59% of one conformation (fig. S54 A) and 41% of the other (fig. S54 B). The flower-and-bird pattern had 64% of one conformation (fig. S57 A) and 36% of the other (fig. S57 B). This phenomenon can be explained by the possible curvature existing in both structures. The face bending upward would be easier to attach to the mica surface compared to the other face.

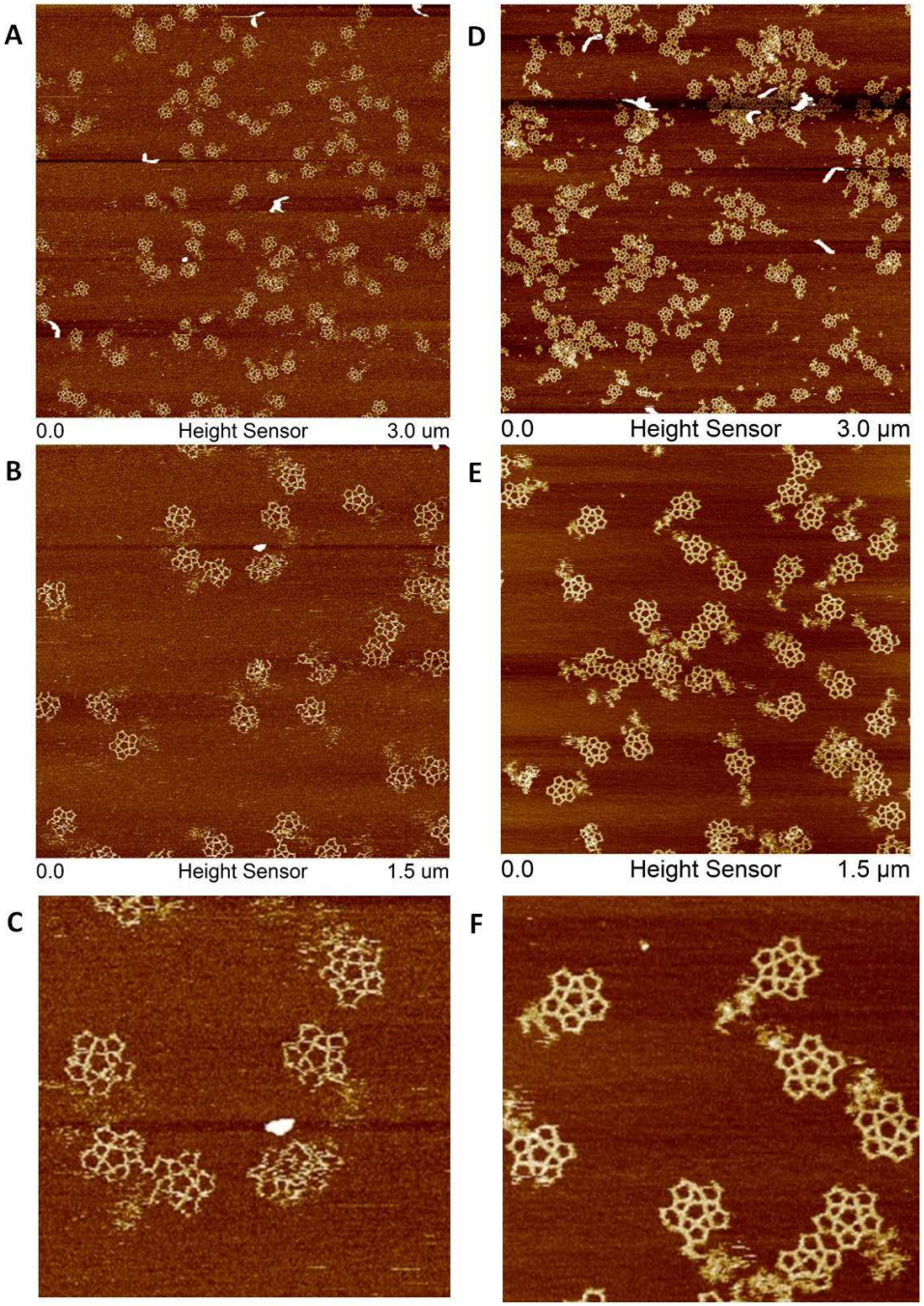


Fig. S47. The effect of $NiCl_2$ on AFM imaging for the star shape. The sample was annealed following the standard protocol described in materials and methods. We deposited 2 µL samples with or without $NiCl_2$ onto freshly cleaved mica. After waiting for 30 s for adsorption onto the mica surface, 80 µL 1× TAE-Mg^{2+} buffer was added to the samples, and an additional 40 µL of the same buffer was deposited on the AFM tips. (**A-C**) AFM images of the star shape without adding $NiCl_2$ on mica surface. (**D-E**) AFM

images of the same sample with addition of 3 μL of 25 mM $NiCl_2$. **C** and **F** are magnified images (660 nm × 660 nm in dimension).

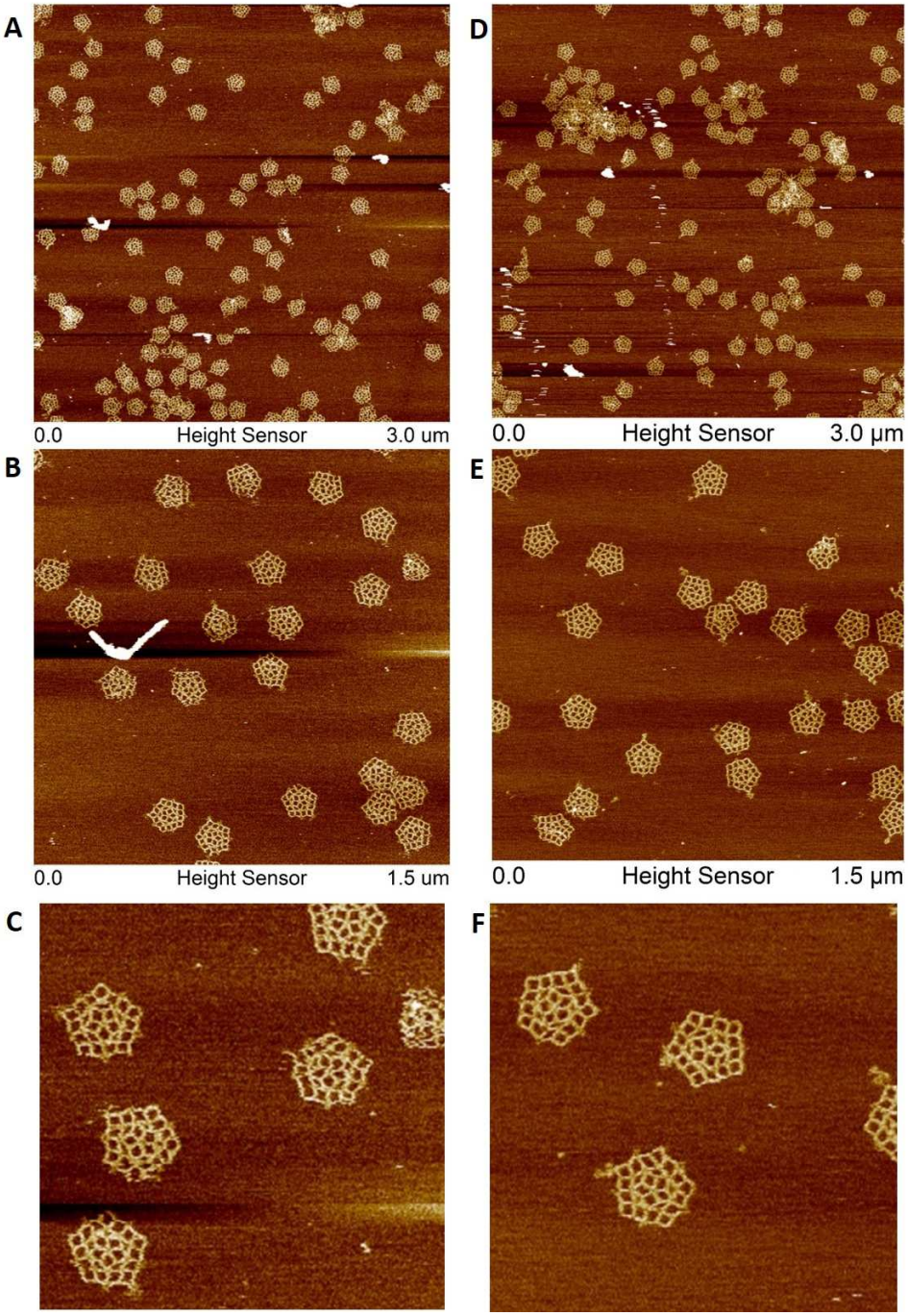
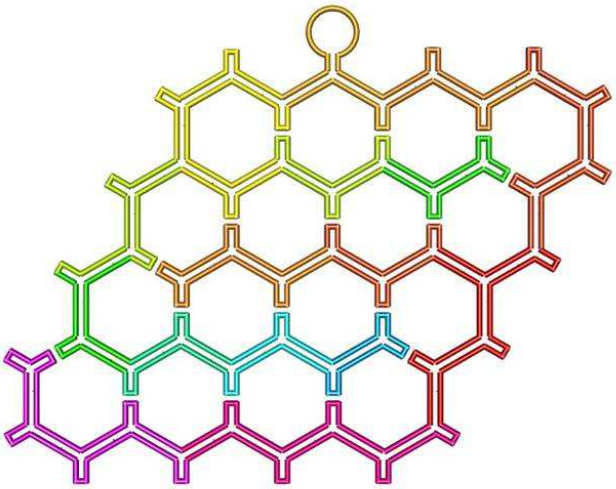


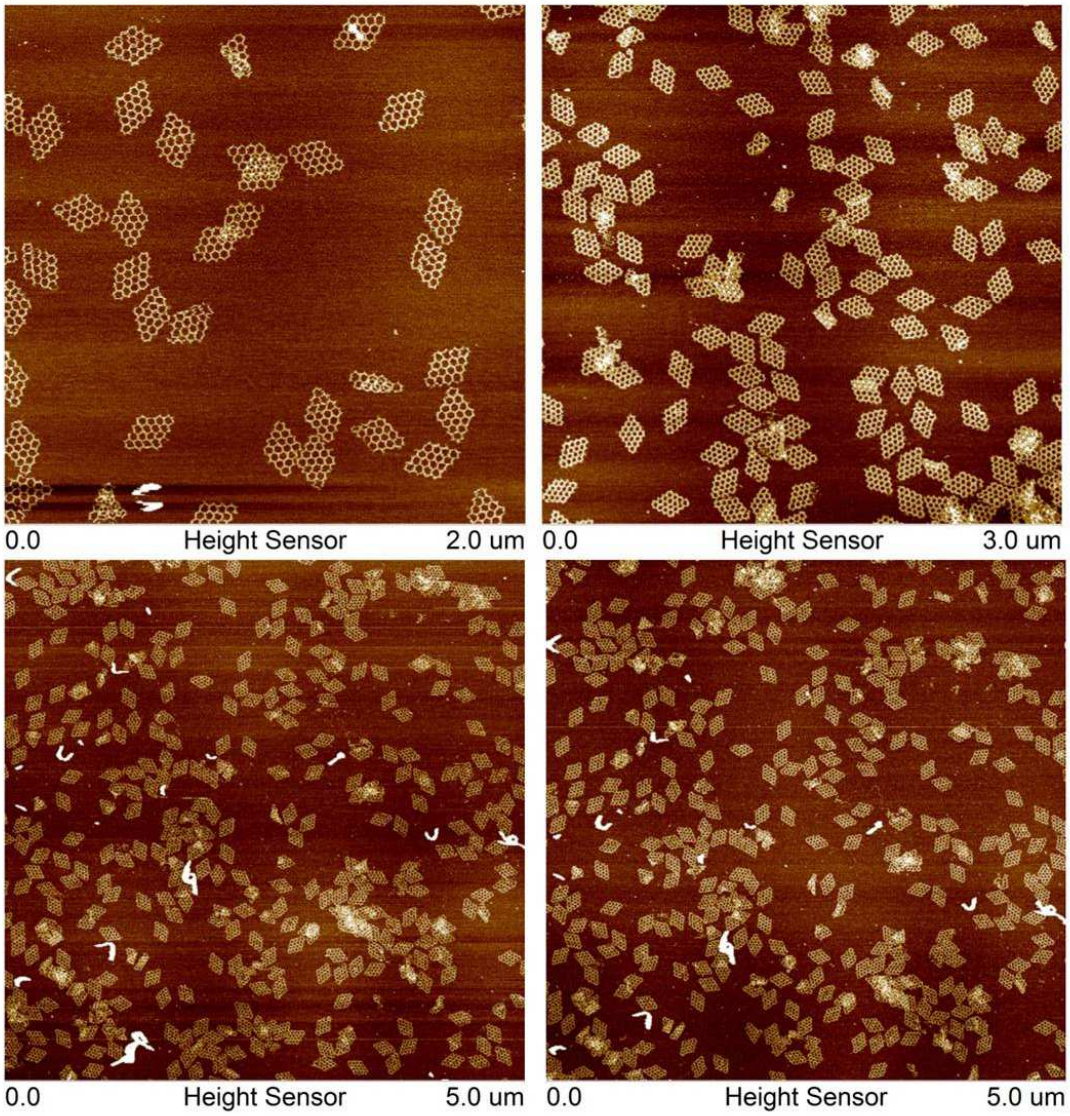
Fig. S48. The effect of $NiCl_2$ on AFM imaging for Penrose tiling. The sample was annealed following the standard protocol described in materials and methods. We deposited 2 μL samples with or without $NiCl_2$ onto freshly cleaved mica. After waiting for 30 s for adsorption onto the mica surface, 80 μL 1× TAE-Mg^{2+} buffer was added to

the samples, and an additional 40 μL of the same buffer was deposited on the AFM tips. (**A-C**) AFM images of the star shape without adding $NiCl_2$ on the mica surface. (**D-E**) AFM images of the same sample with addition of 3 μL $NiCl_2$ of 25 mM. **C** and **F** are magnified images (660 nm × 660 nm in dimension).

Section 4. Additional AFM images of the wireframe DNA origami

Additional AFM images of the wireframe DNA origami are listed in this section. For each designed structure, the scaffold-folding path is displayed first, followed by the wide AFM images and then the magnified AFM images. The protocols for AFM imaging are described in materials and methods.




0.0
Height Sensor
2.0 um
0.0
Height Sensor
3.0 um
0.0
Height Sensor
5.0 um
0.0
Height Sensor
5.0 um

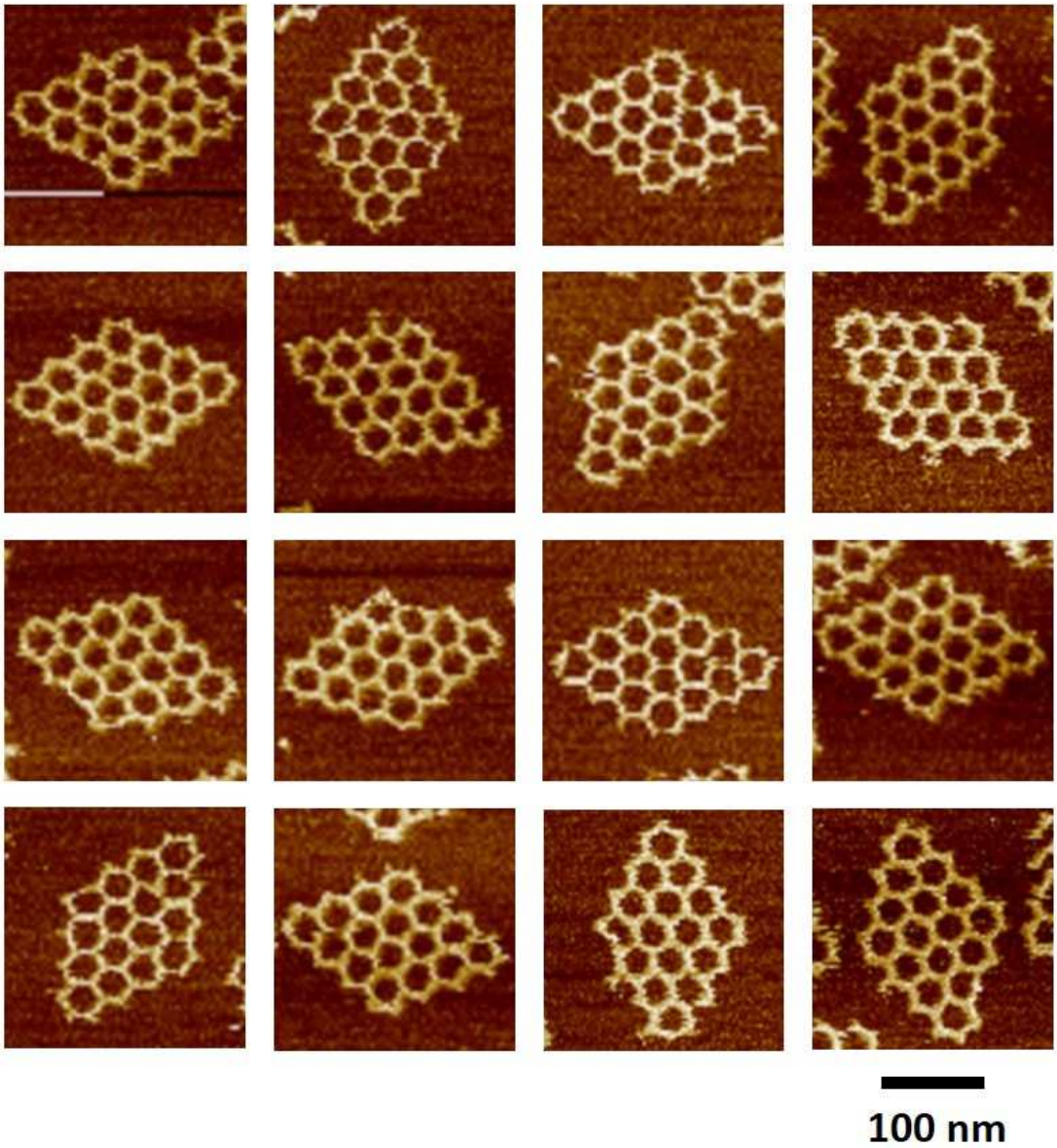
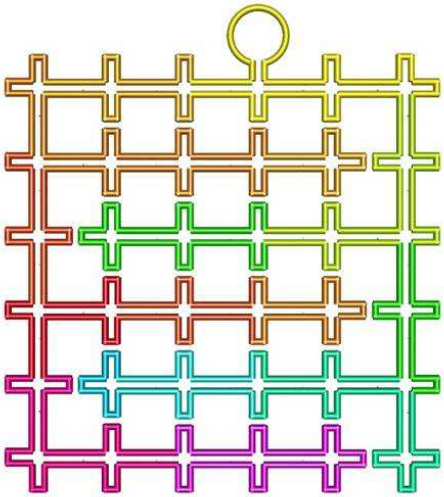
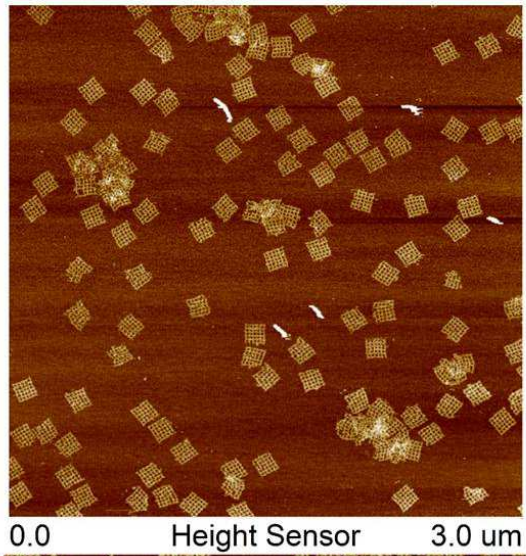
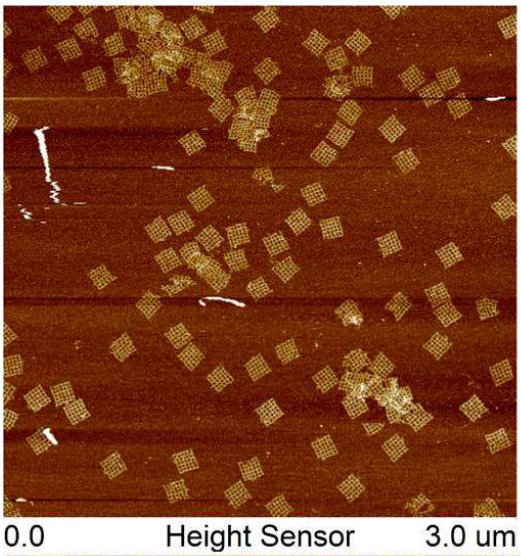
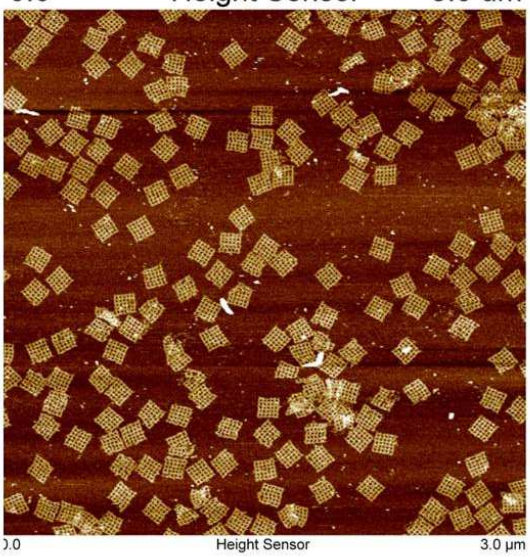


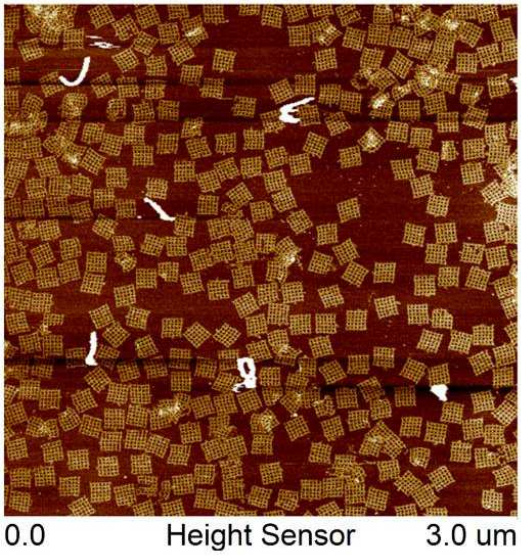
Fig. S49. AFM images of the honeycomb Platonic tiling




0.0
Height Sensor
3.0 um


0.0
Height Sensor
3.0 um


Height Sensor
3.0 µm


0.0
Height Sensor
3.0 um

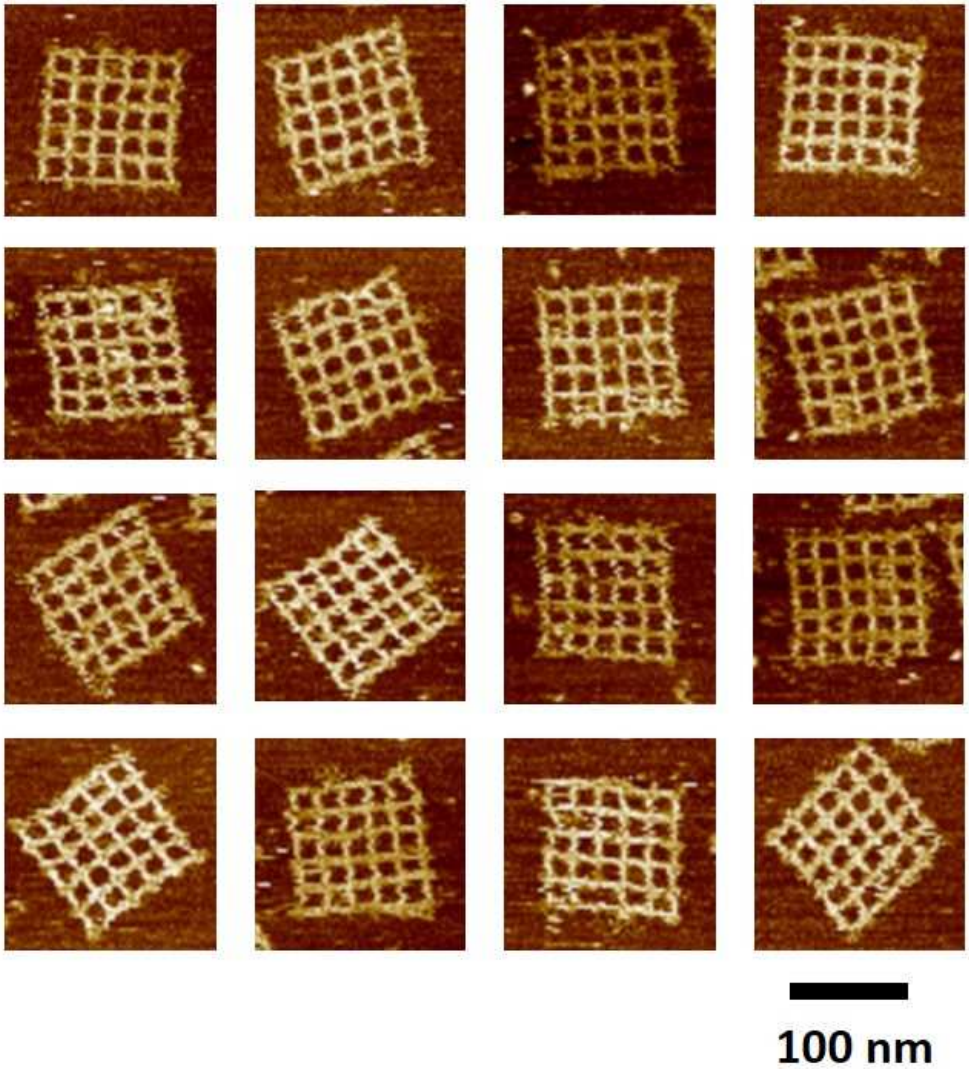
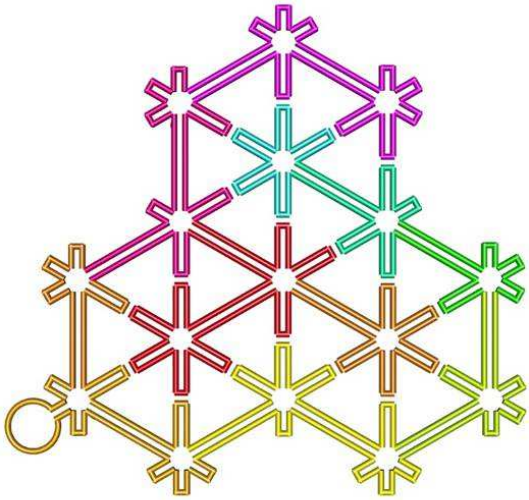
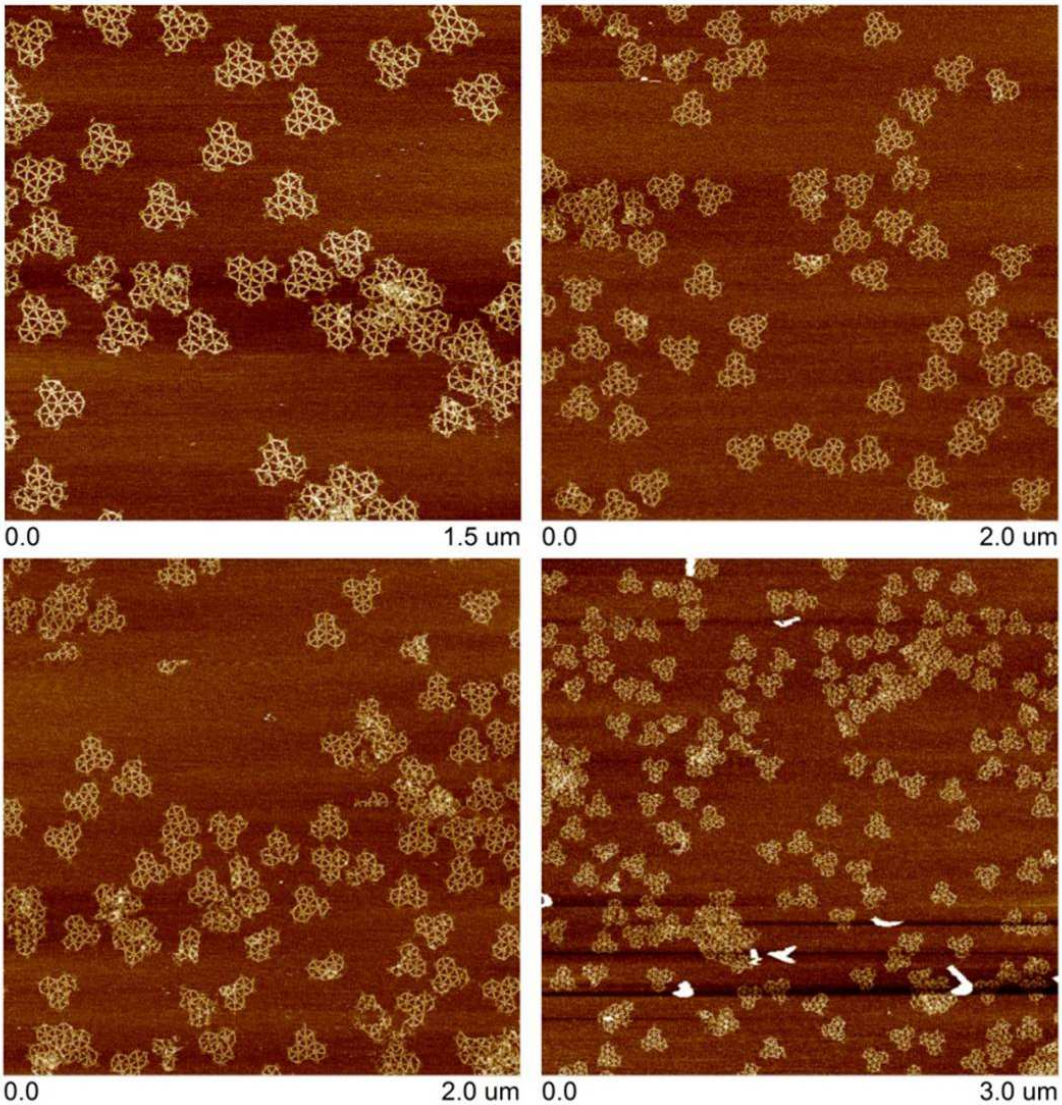


Fig. S50. AFM images of the square Platonic tiling




0.0
1.5 um
0.0
2.0 um
0.0
2.0 um
0.0
3.0 um

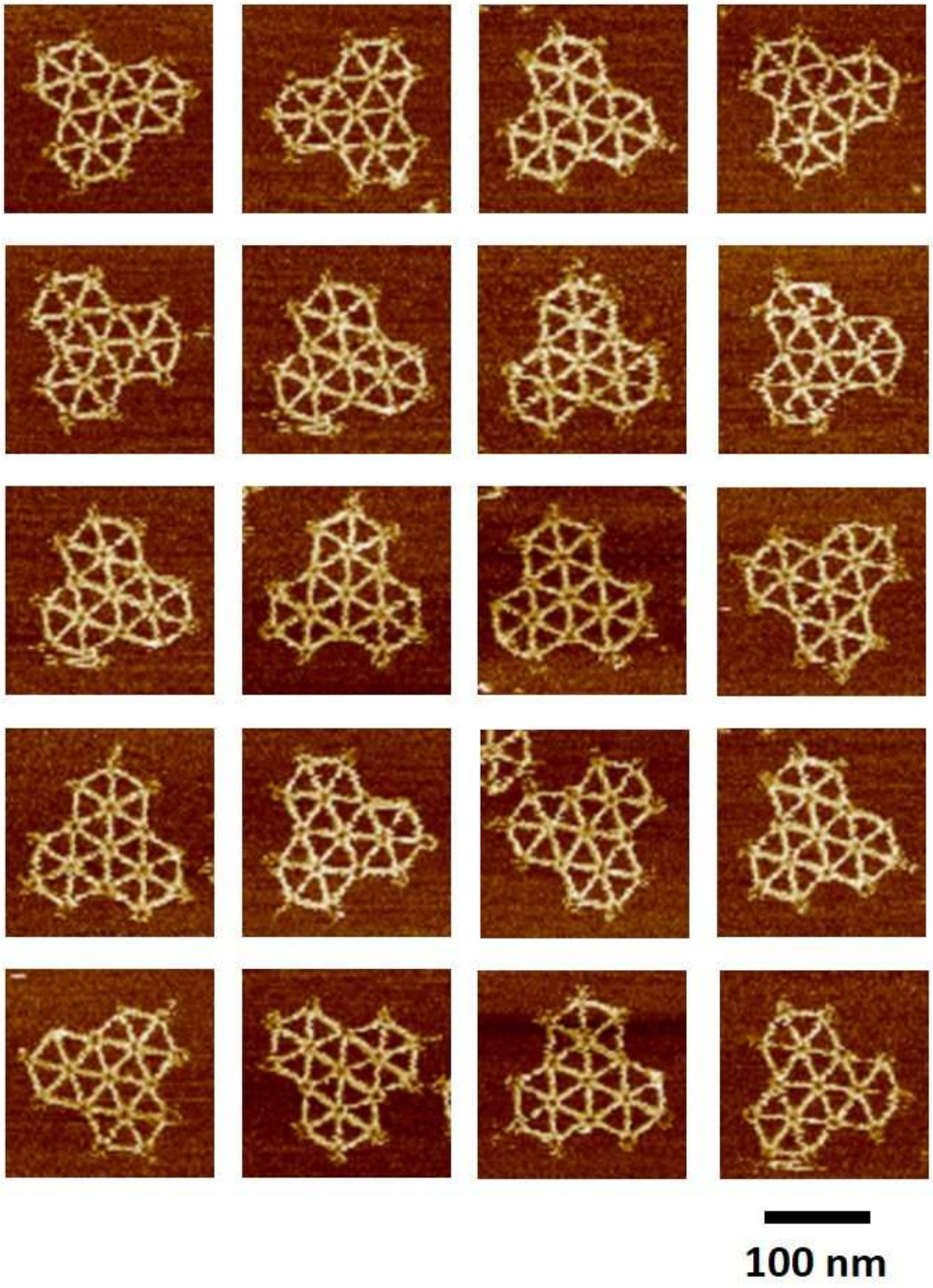
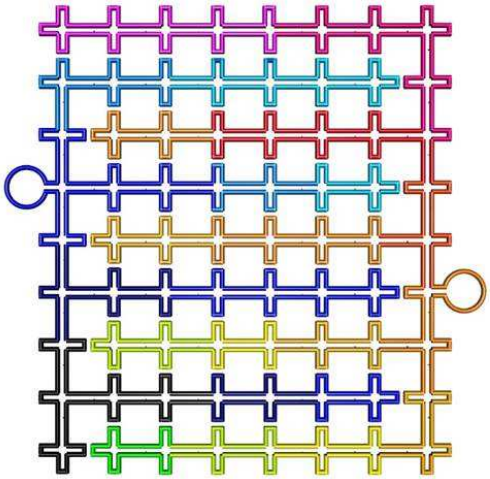
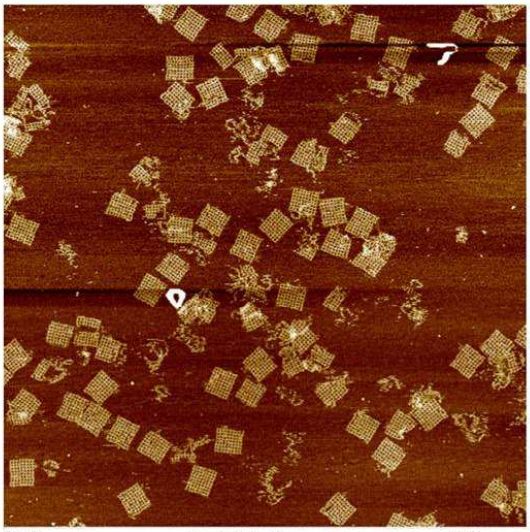


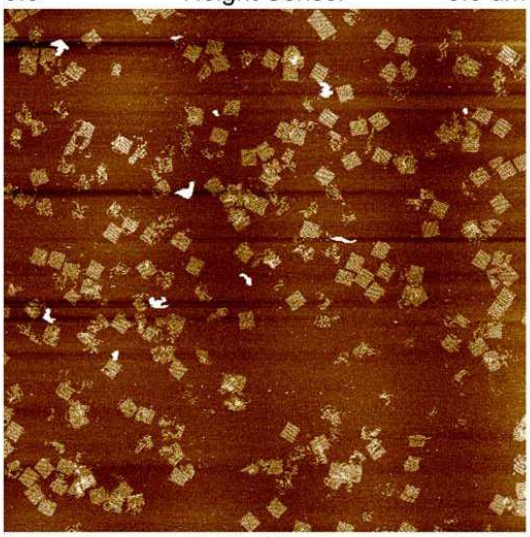
Fig. S51. AFM images of the triangular Platonic tiling



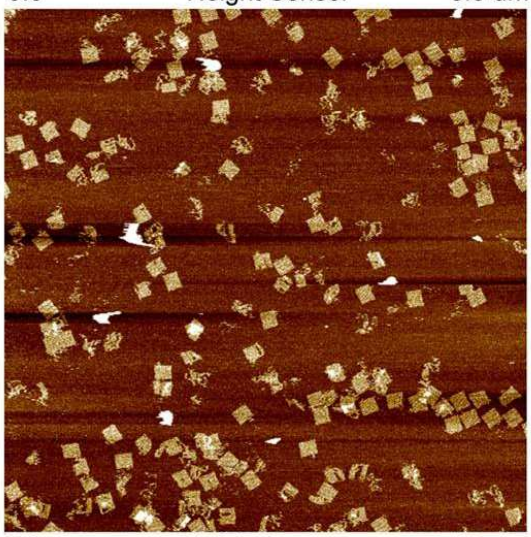


0.0 Height Sensor 3.0 um

0.0 Height Sensor 3.0 um



0.0 Height Sensor 5.0 um



0.0 Height Sensor 5.0 um

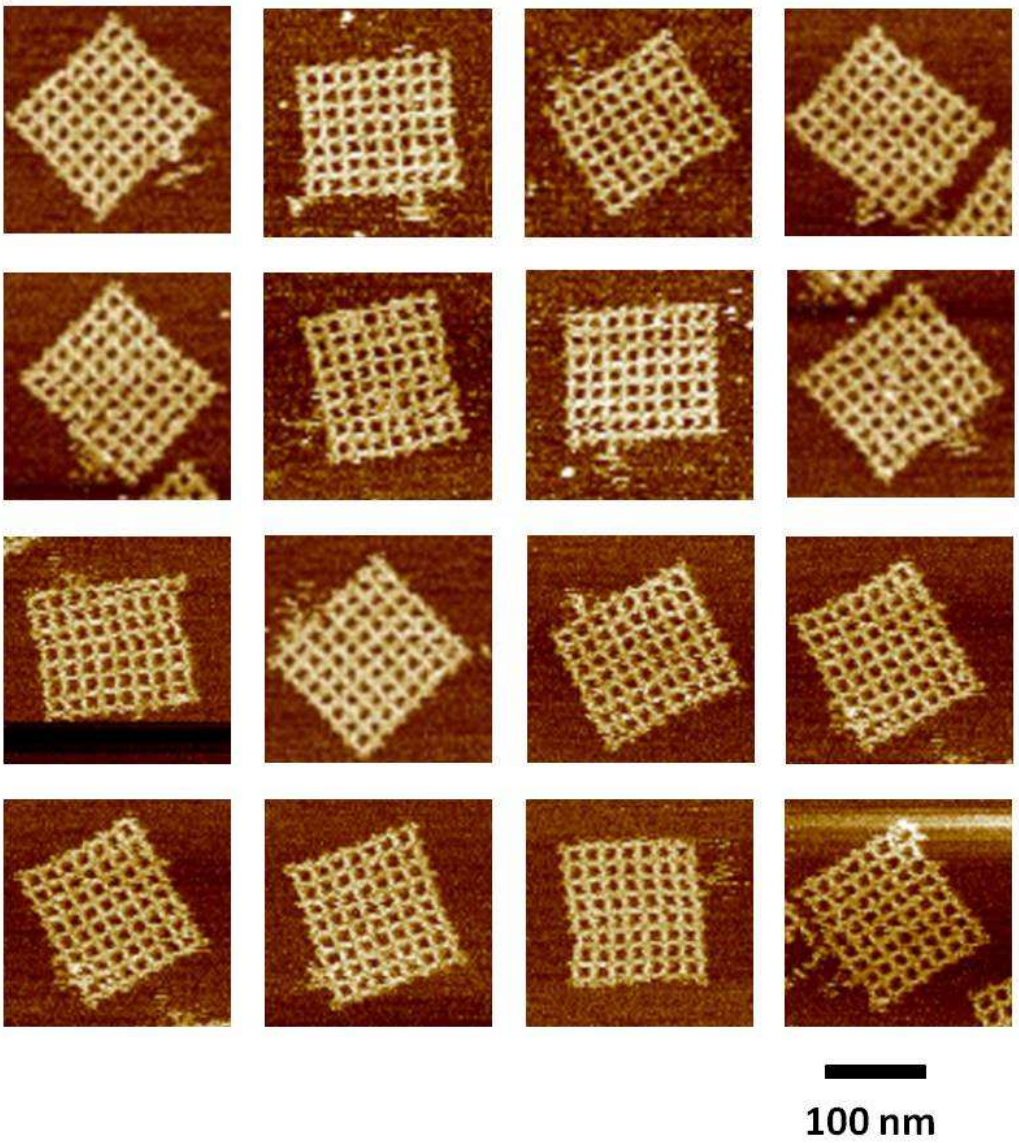
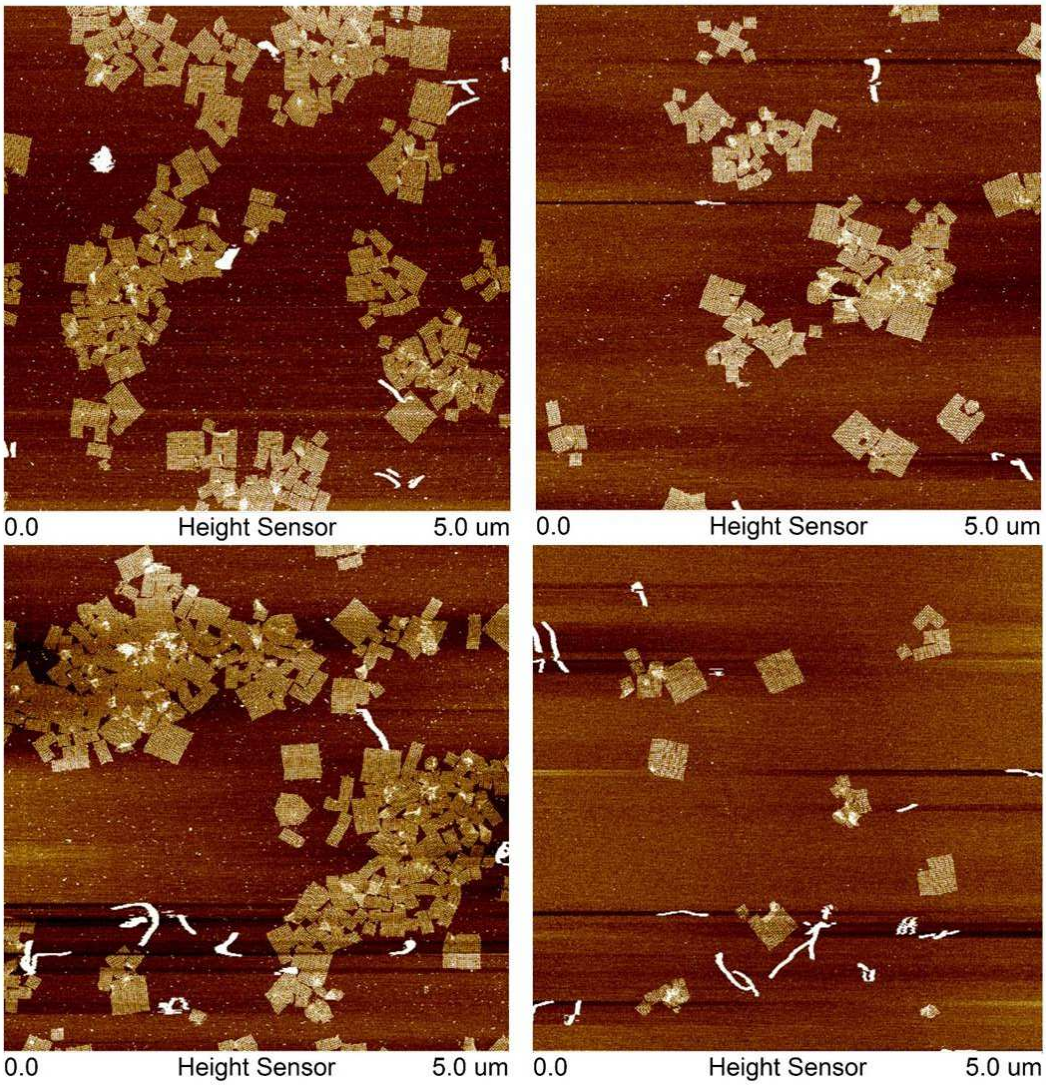


Fig. S52. AFM images of the two-scaffold square tiling


0.0
Height Sensor
5.0 um
0.0
Height Sensor
5.0 um
0.0
Height Sensor
5.0 um
0.0
Height Sensor
5.0 um

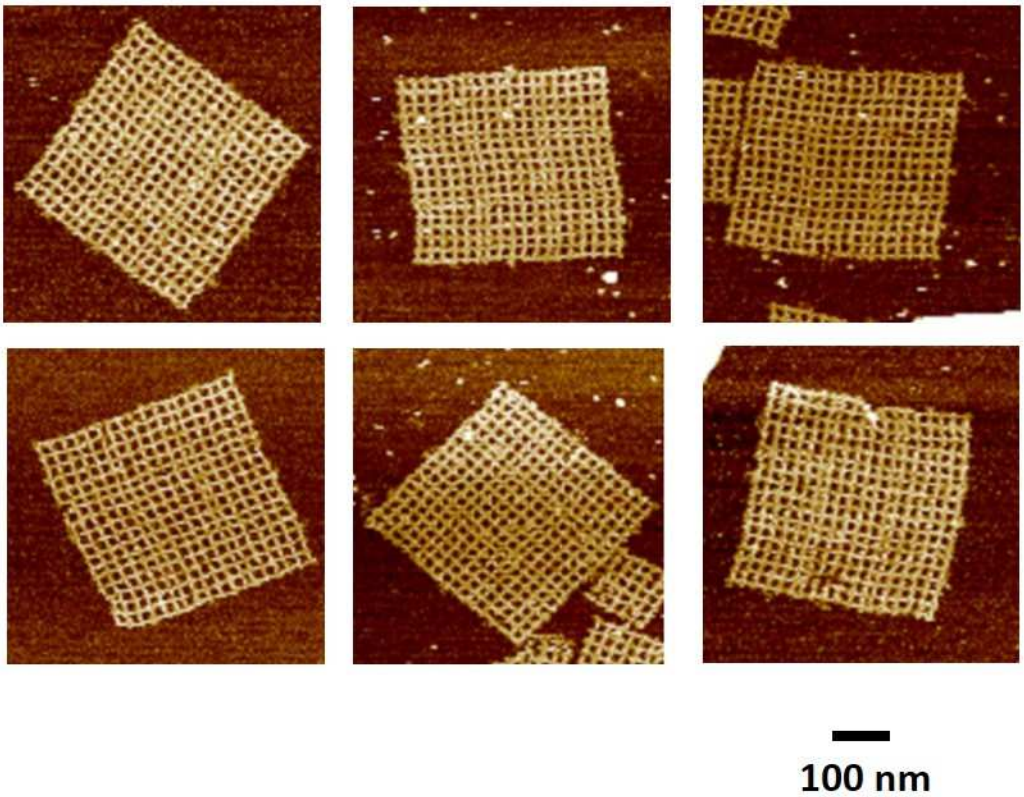
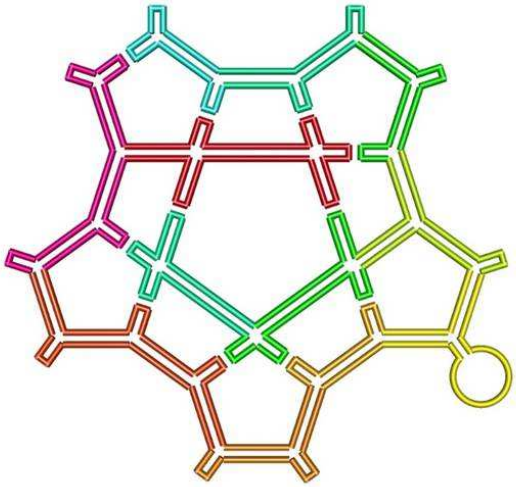
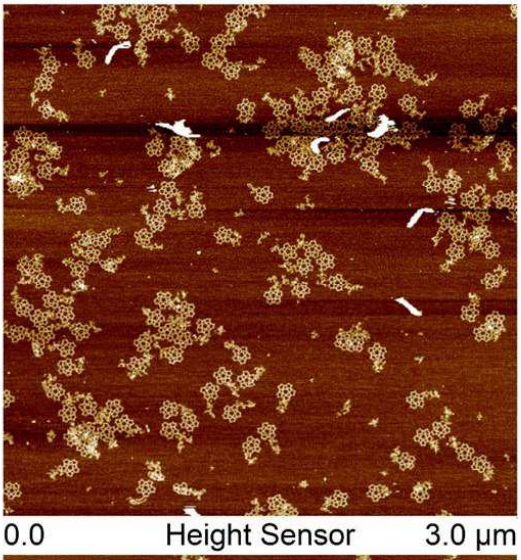
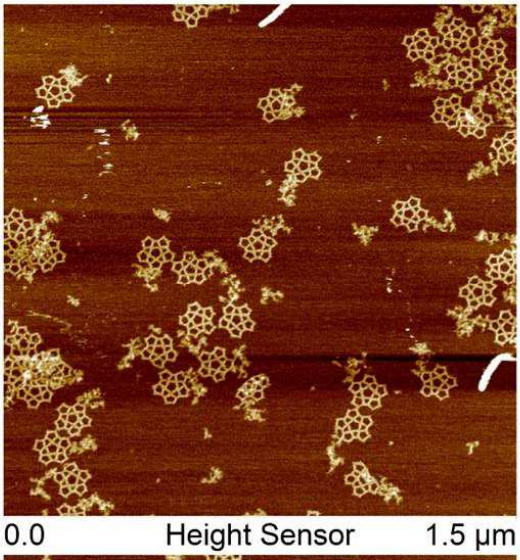
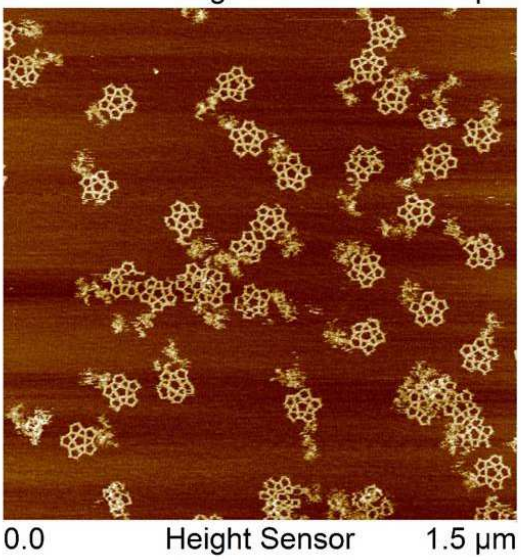


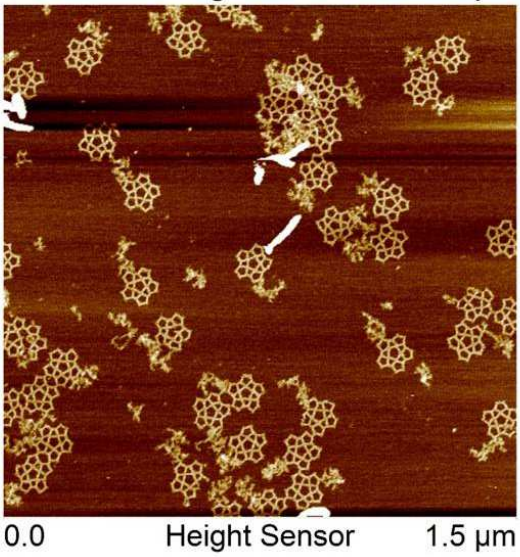
Fig. S53. AFM images of the 3 × 3 square origami array




0.0
Height Sensor
3.0 µm


0.0
Height Sensor
1.5 µm


0.0
Height Sensor
1.5 µm


0.0
Height Sensor
1.5 µm

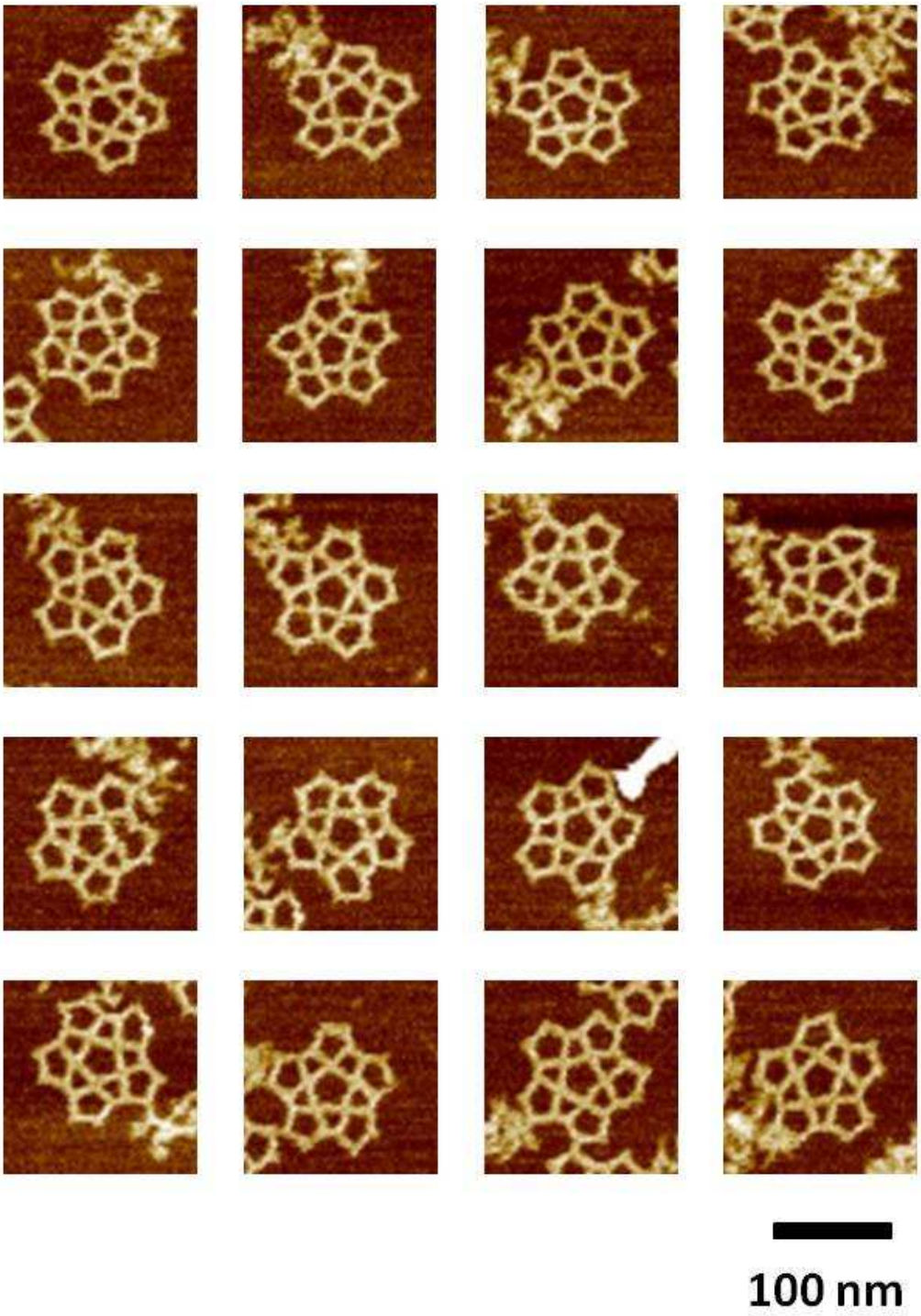
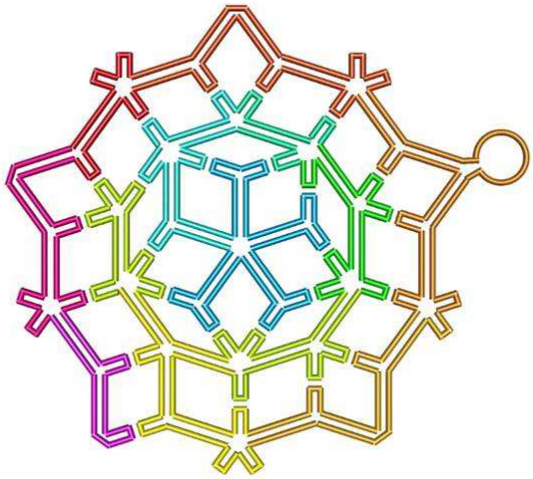
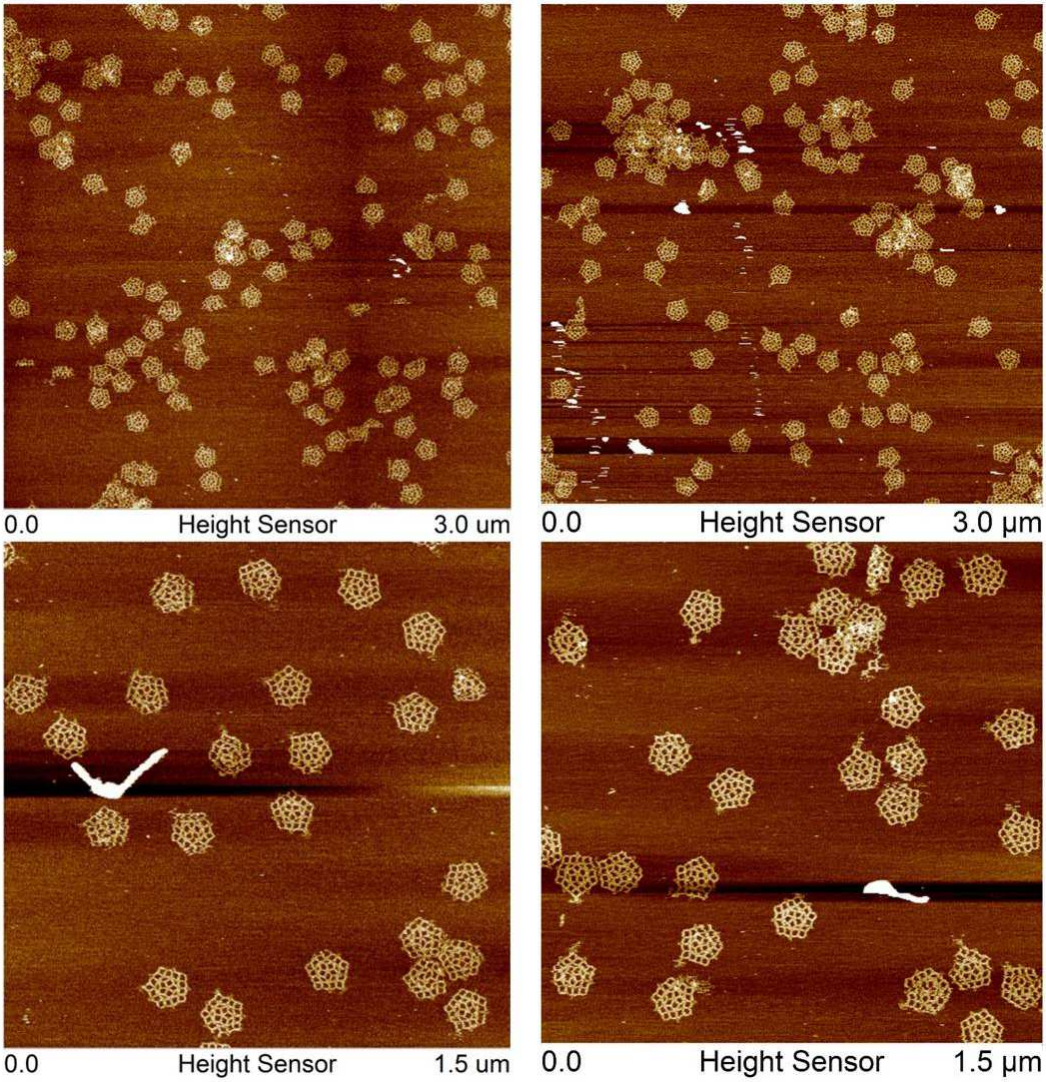


Fig. S54. AFM images of the star shape




0.0
Height Sensor
3.0 um
0.0
Height Sensor
3.0 µm
0.0
Height Sensor
1.5 um
0.0
Height Sensor
1.5 µm

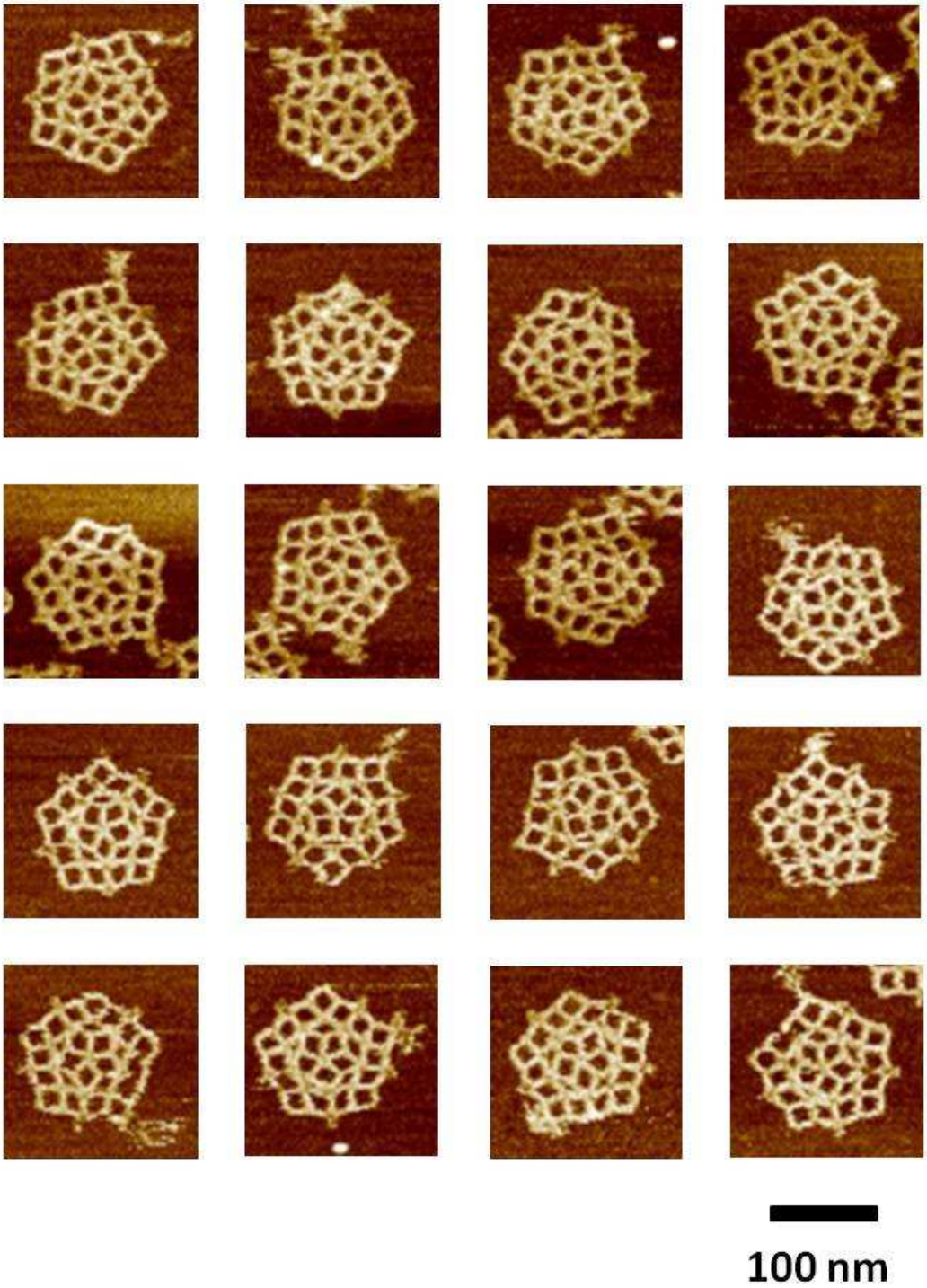
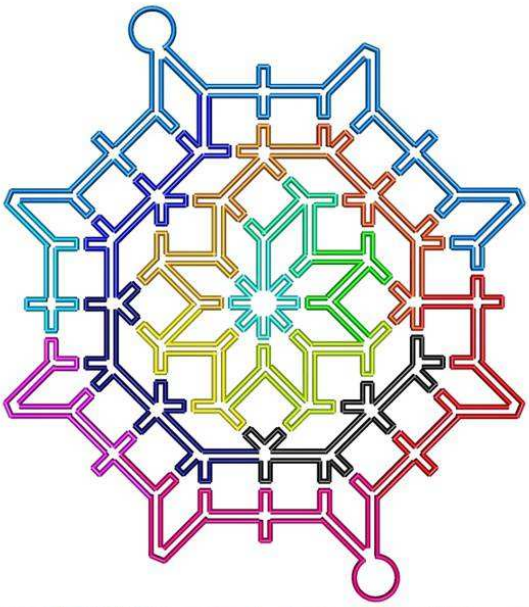
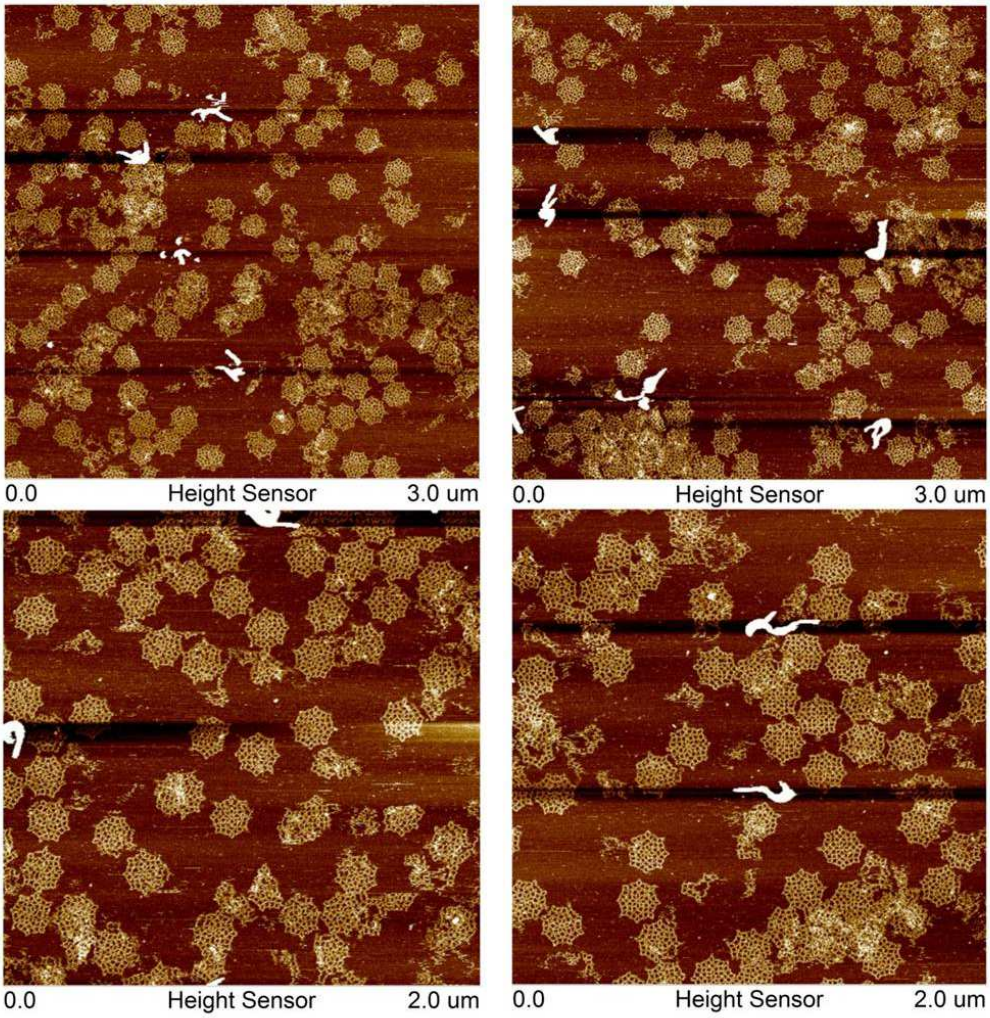


Fig. S55. AFM images of the 2D Penrose tiling




0.0
Height Sensor
3.0 um
0.0
Height Sensor
3.0 um
0.0
Height Sensor
2.0 um
0.0
Height Sensor
2.0 um

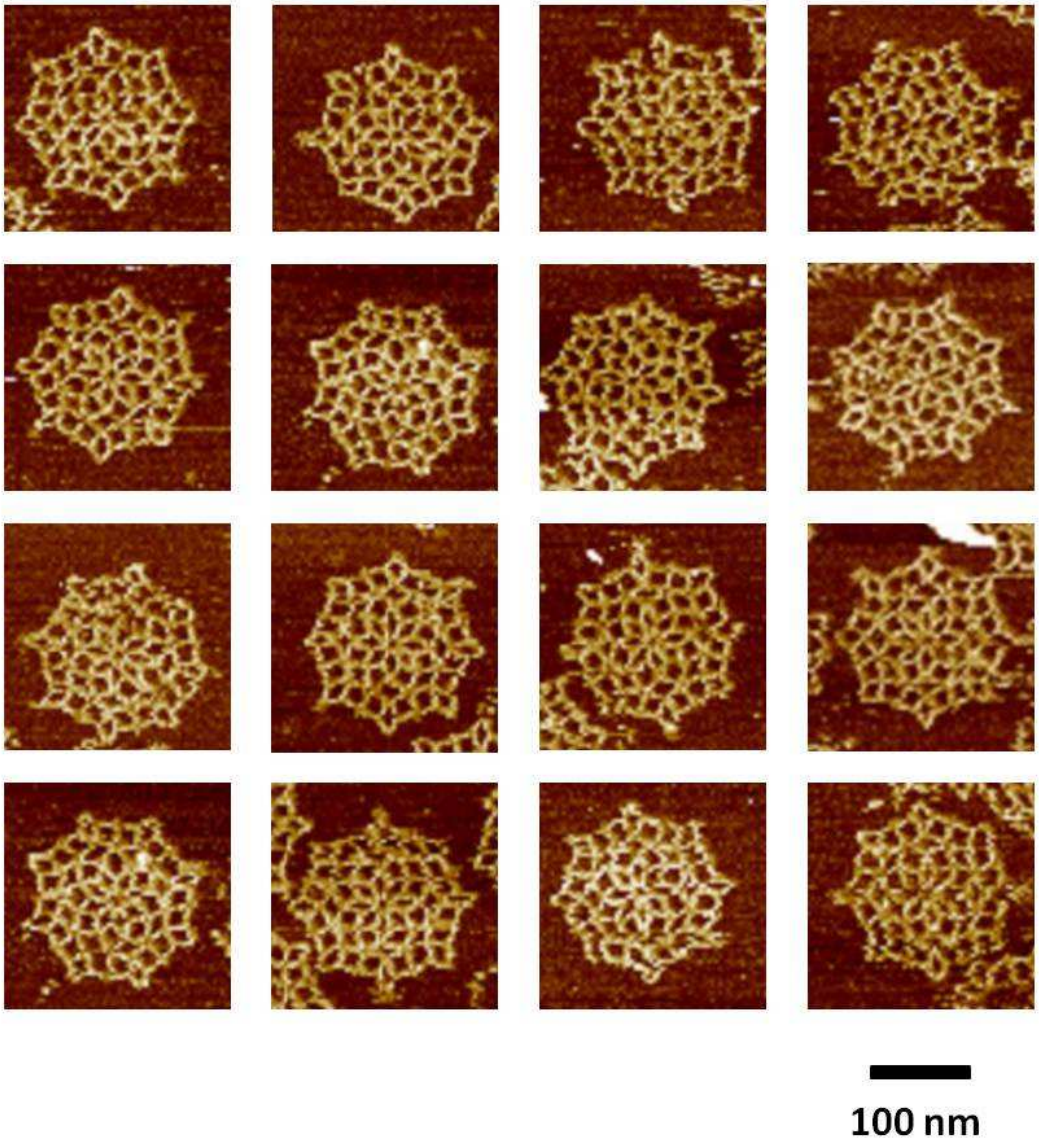
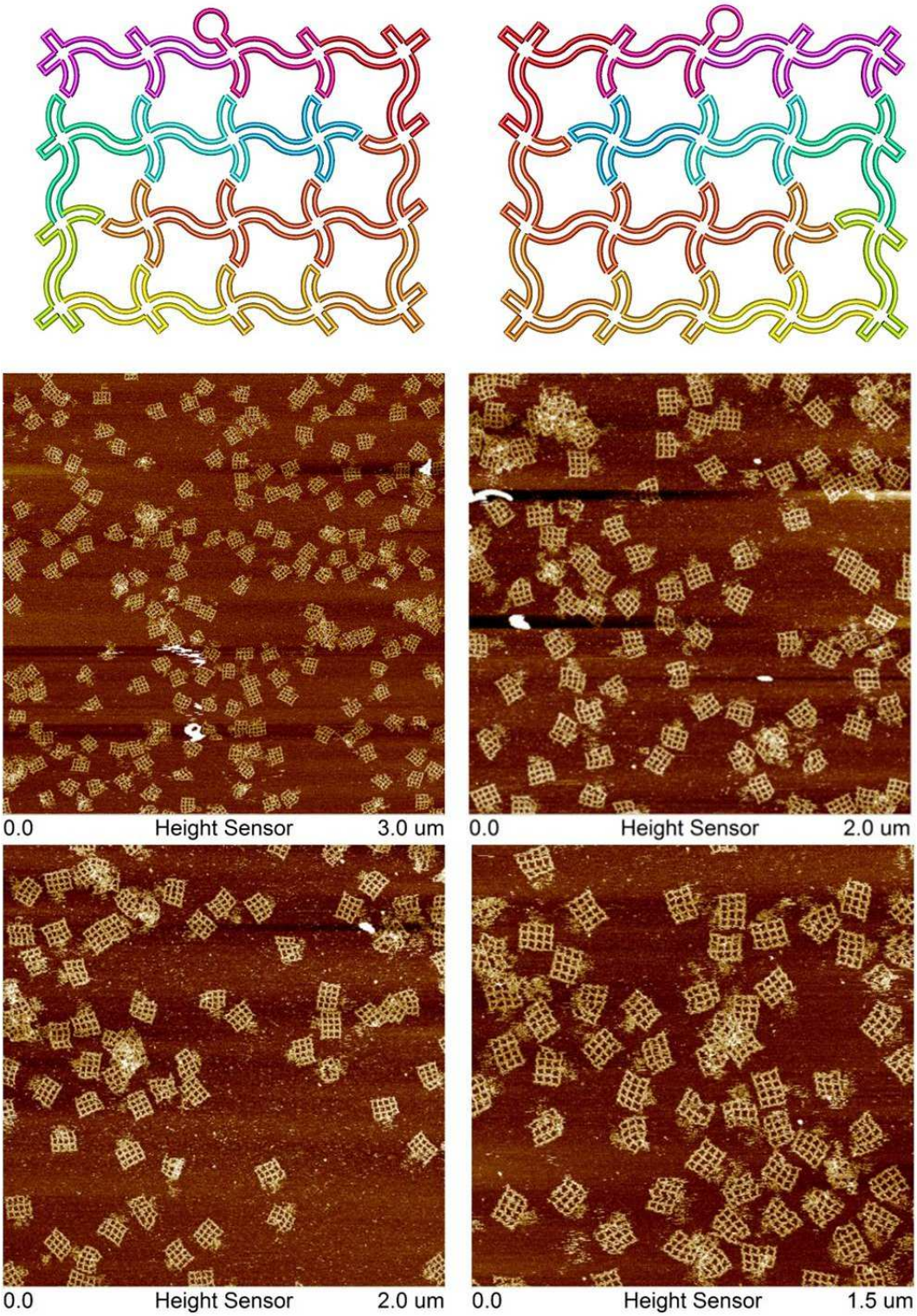


Fig. S56. AFM images of the 2D quasicrystalline pattern with 8-fold symmetry


0.0
Height Sensor
3.0 um
0.0
Height Sensor
2.0 um
0.0
Height Sensor
2.0 um
0.0
Height Sensor
1.5 um

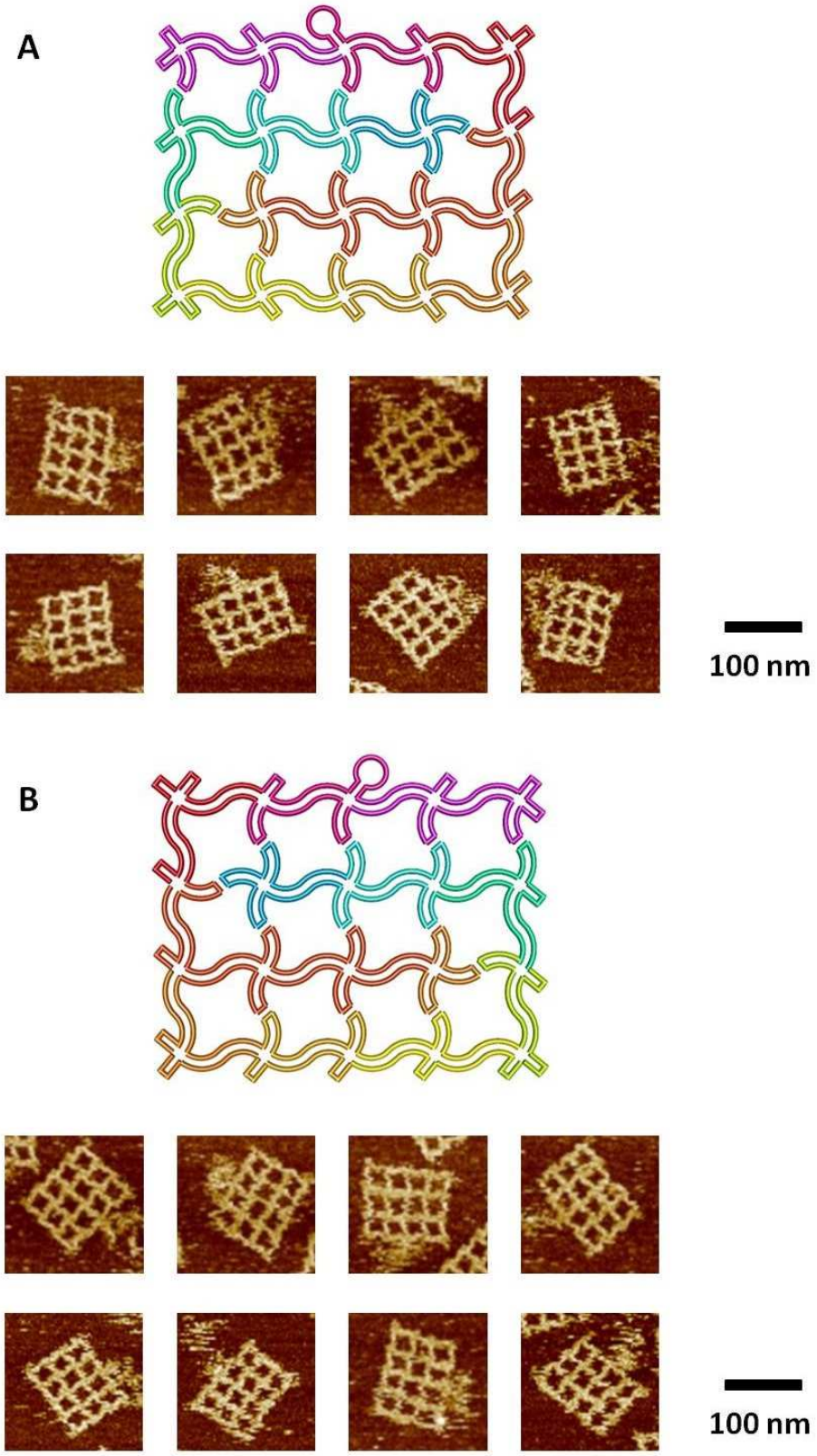
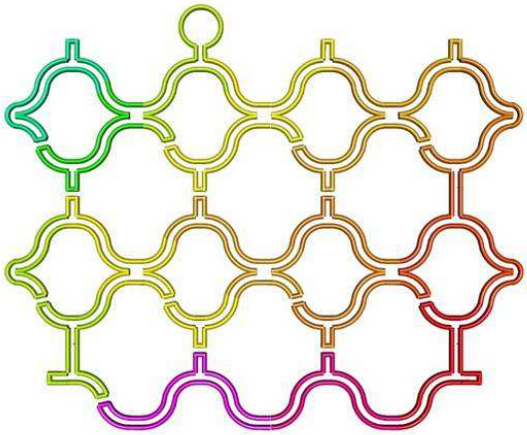
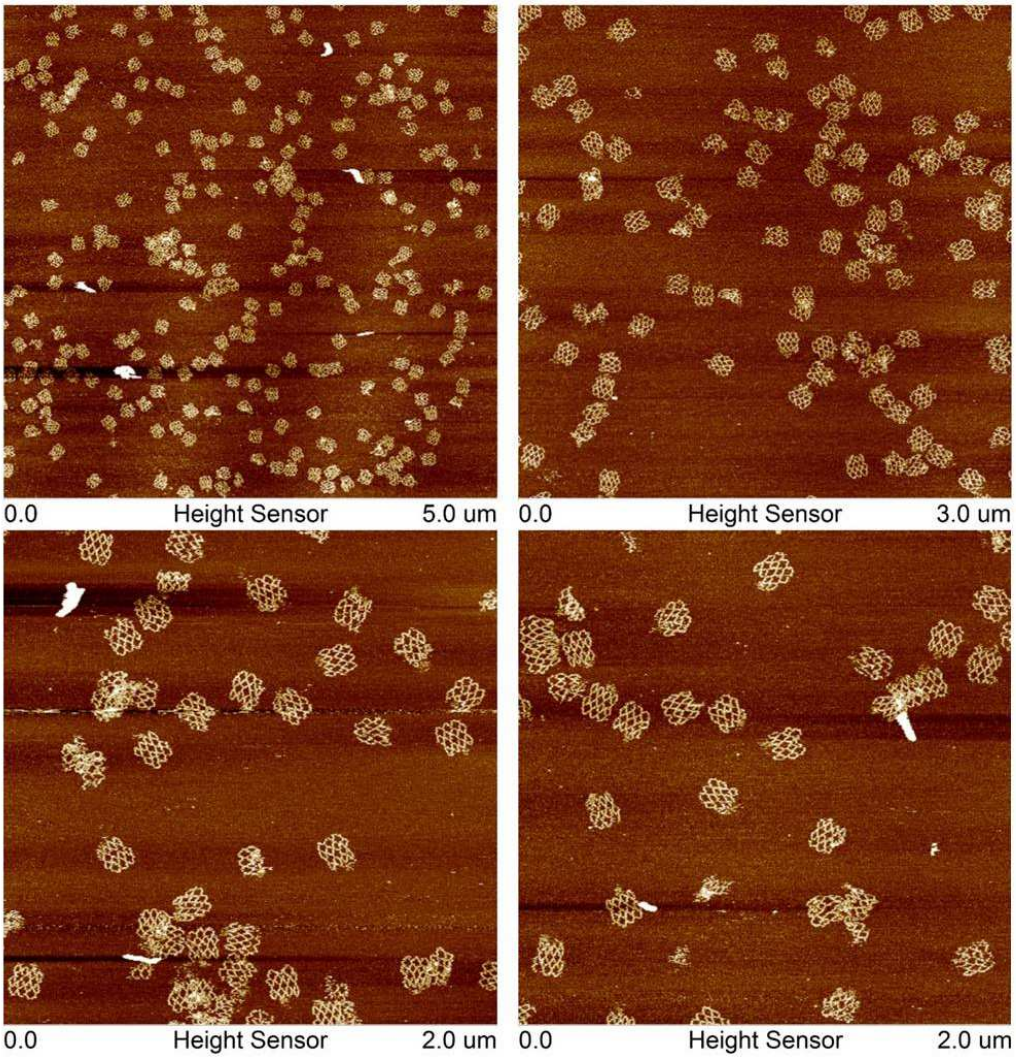


Fig. S57. AFM images of the waving grid




0.0
Height Sensor
5.0 um
0.0
Height Sensor
3.0 um
0.0
Height Sensor
2.0 um
0.0
Height Sensor
2.0 um

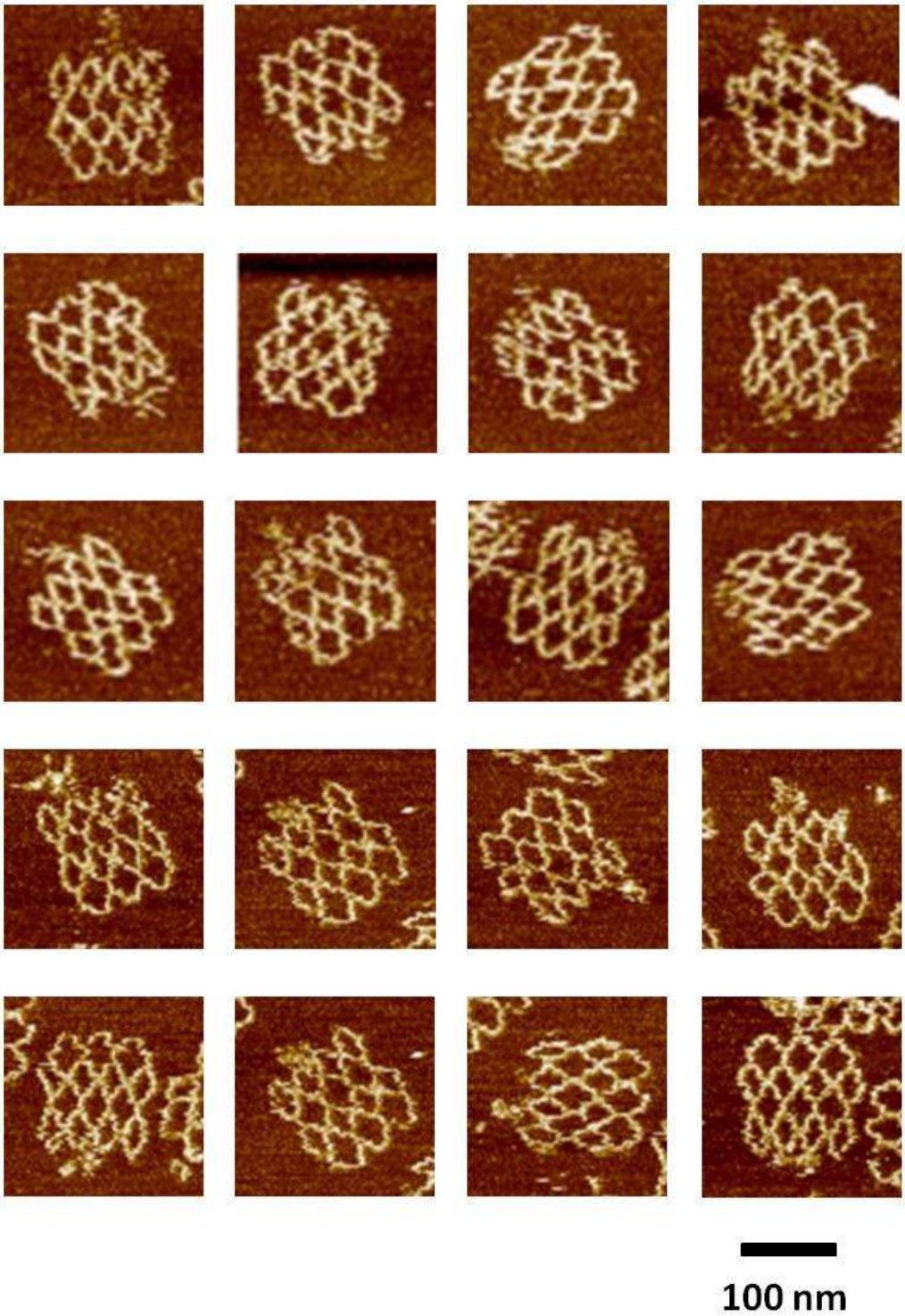
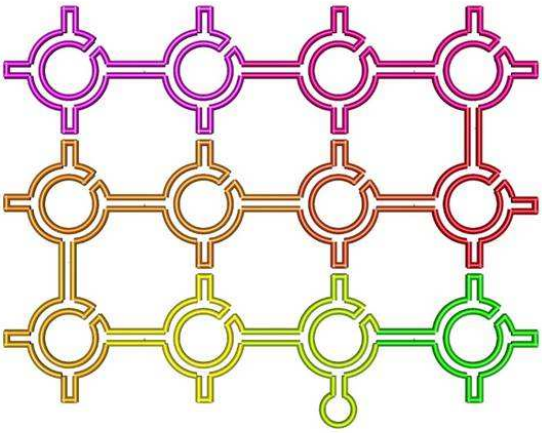
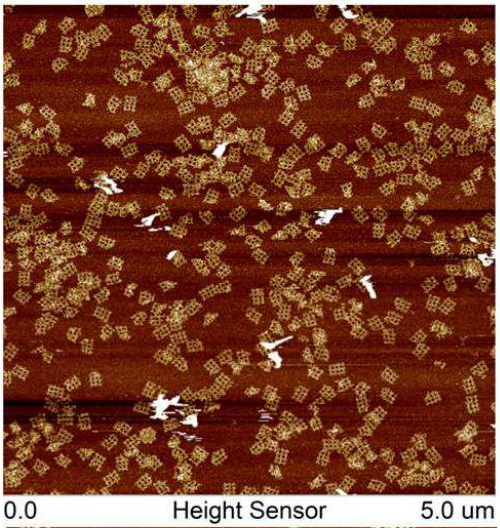
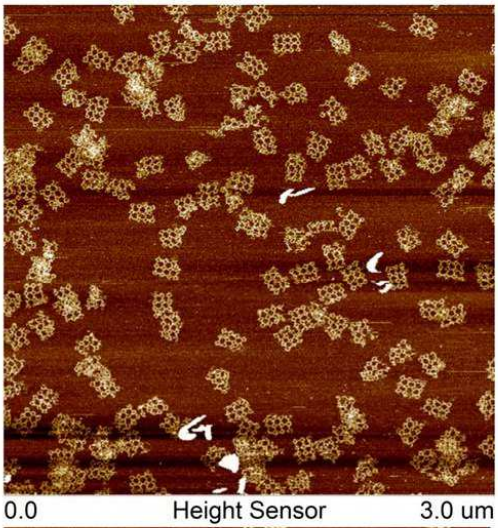
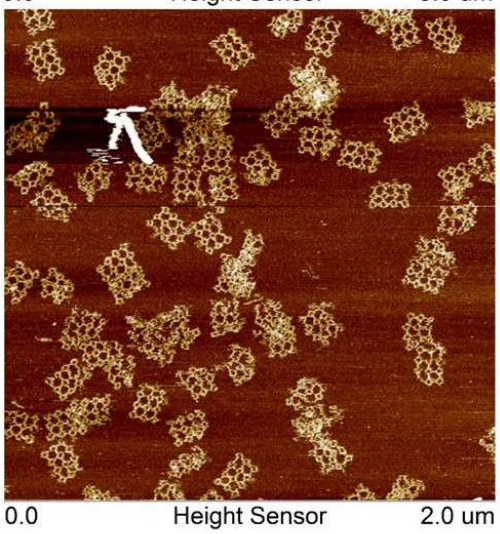


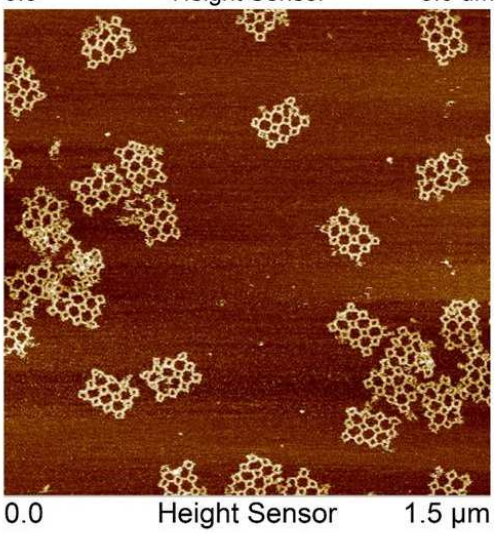
Fig. S58. AFM images of the fishnet




0.0
Height Sensor
5.0 um


0.0
Height Sensor
3.0 um


0.0
Height Sensor
2.0 um


0.0
Height Sensor
1.5 µm

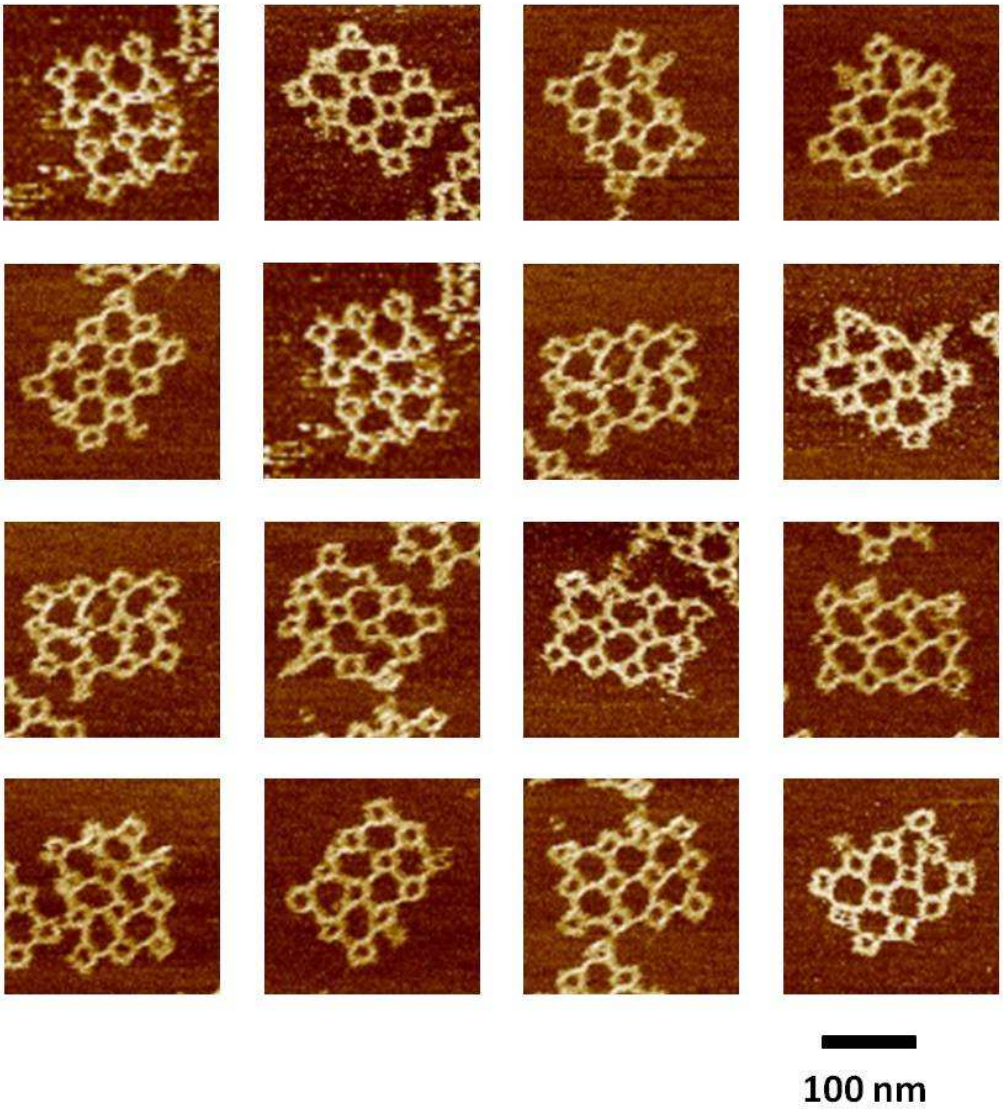
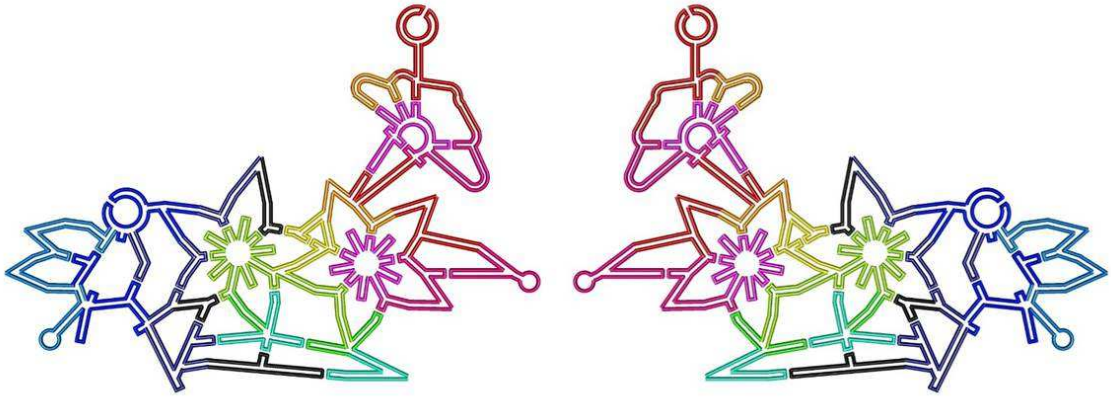
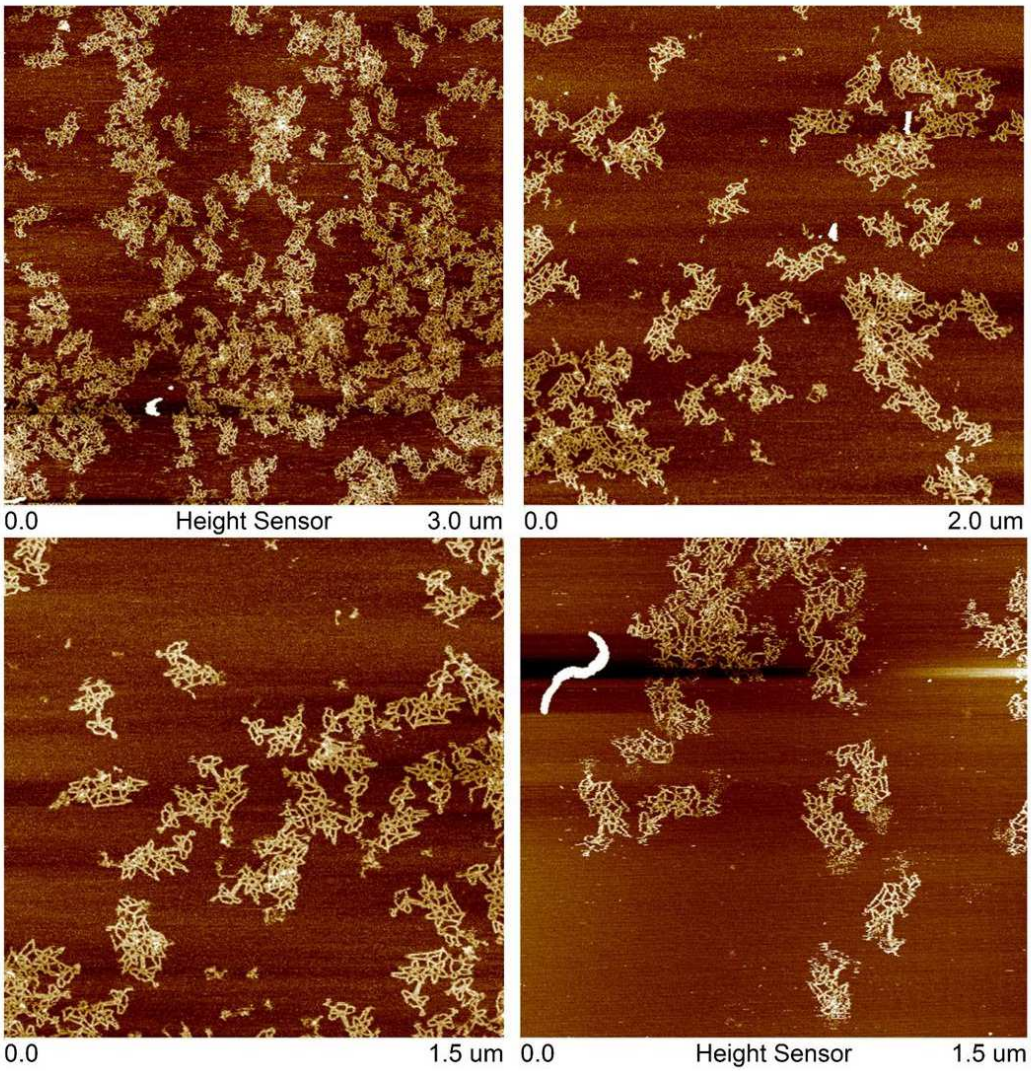
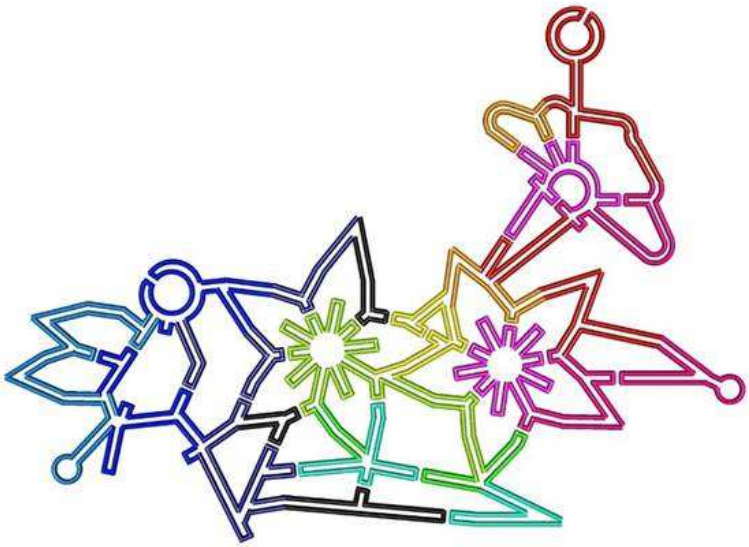
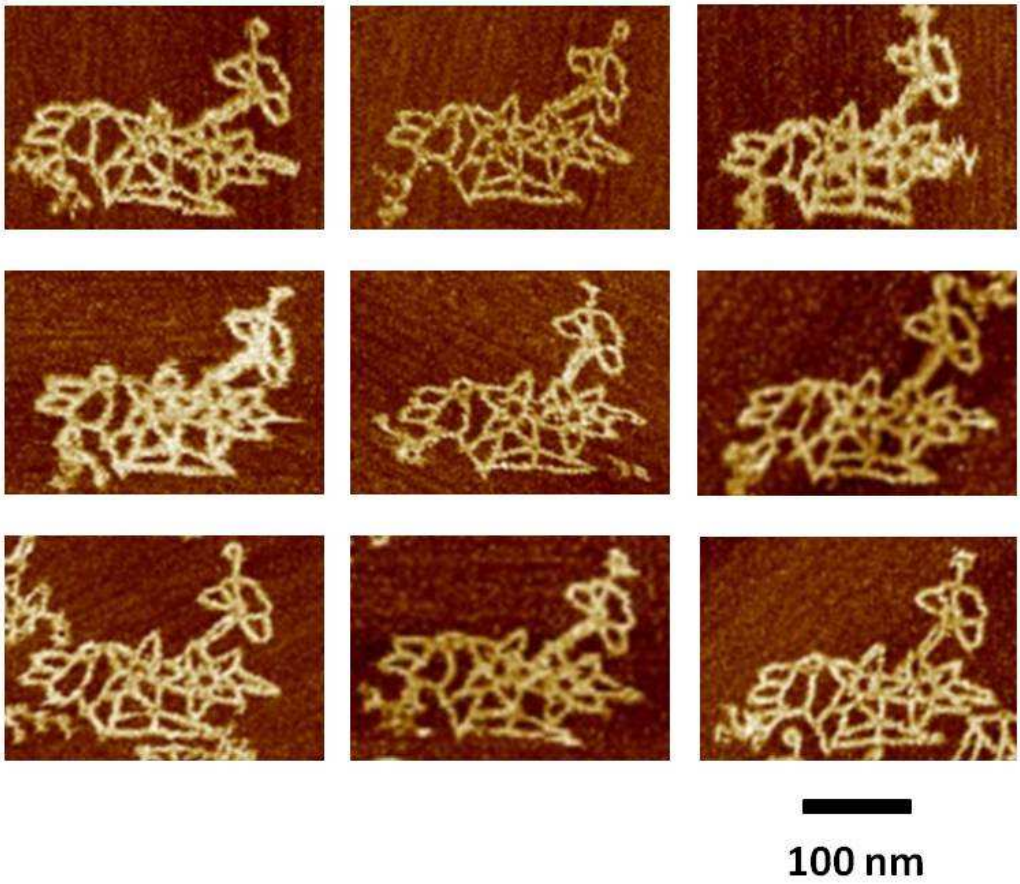


Fig. S59. AFM images of the circle array




0.0
Height Sensor
3.0 um
0.0
2.0 um
0.0
1.5 um
0.0
Height Sensor
1.5 um




100 nm

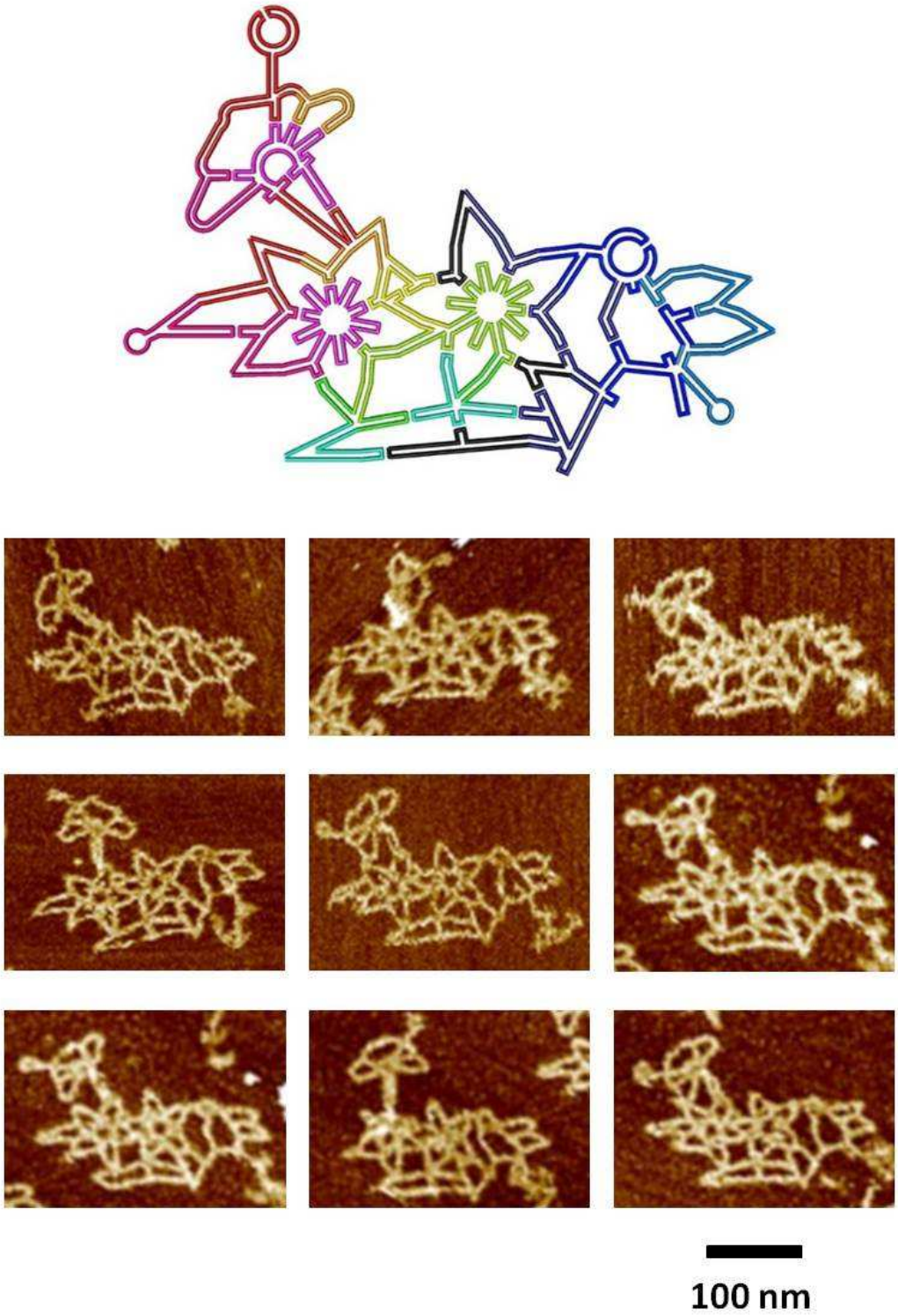


Fig. S60. AFM images of the flower-and-bird pattern. The flower-and-bird pattern presents both mirror images under AFM, which is as expected, as the object may land on the surface in both ways. However, the chirality presented some significant bias, which

reflects different tendencies of the two surfaces of the structure deposited on the mica surface. We counted 54 well-formed flower-and-bird patterns in total, and 34 of them (63%) gave the preferred depositing position with the flowers on the right and the bird on the left. This indicates that there might be a global curvature in the DNA structures that makes the front surface (as shown in Fig. 3G) easier to deposit than the back surface, presumably due to better and larger surface contact. A rational guess is that the middle part of the structure actually bulges away from the 2D plane. Further structural strain analysis is required to predict the global twist or overall bending of the final structure (see supplemental information in section 3 to find a discussion about curvature and chirality). A similar phenomenon was observed for the waving grid structure (as shown in Fig. 3D and fig. S57), which has 59% (28 out of 48) objects adopting the right-handed laying conformation under AFM.

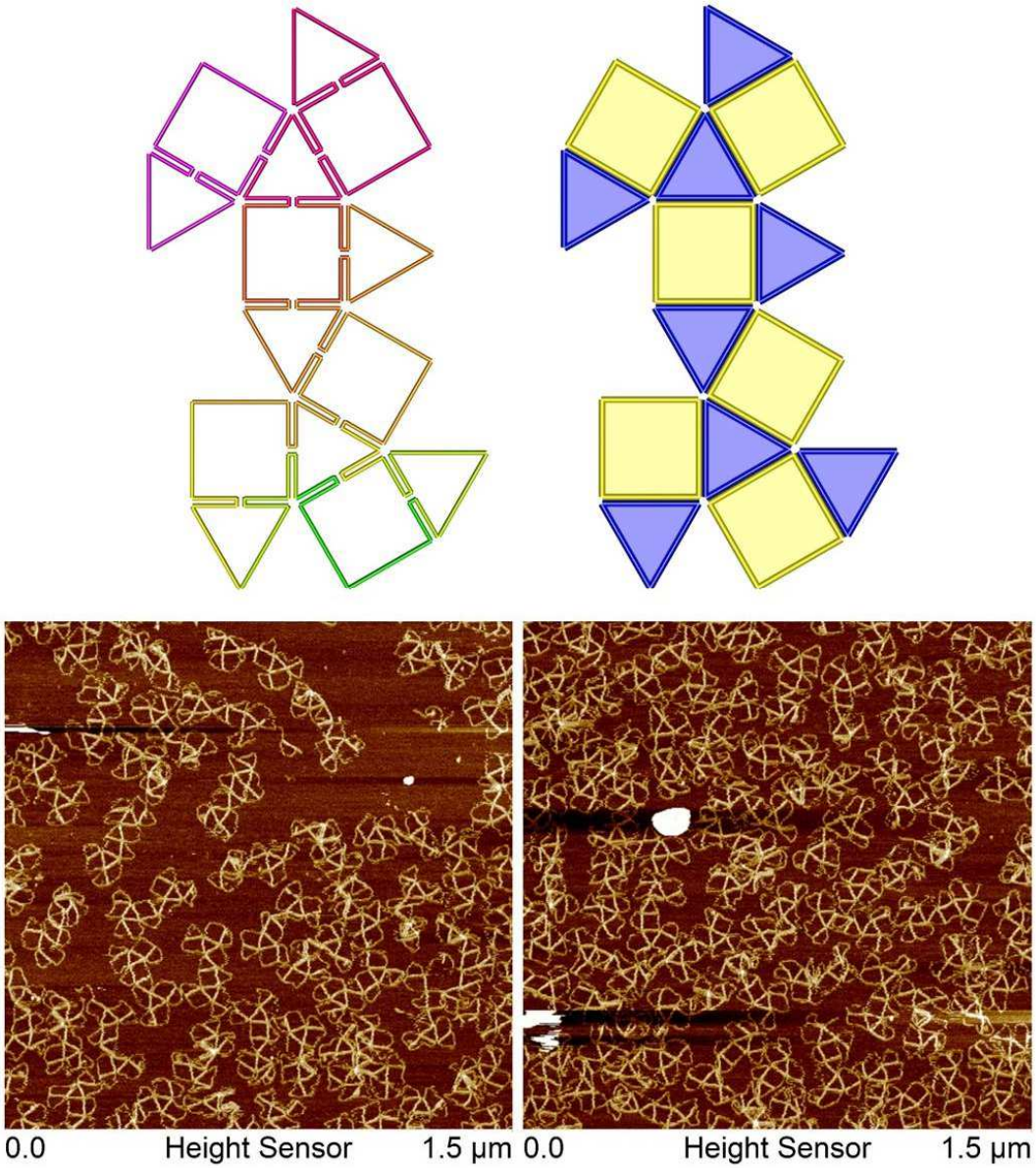


Fig. S61. AFM images of the annealed 2D open net of the cuboctahedron. The 2D open net of this polyhedron was first prepared as shown in Fig. 4C (right). The AFM images reveal variations of the contrast (height), line width, and deformation, indicating that all the edges in the center part are double helixes, while those along the outside are a single helix. All helper strands along the outermost edges of the 2D net were designed with single-stranded toehold extensions (8 nt) that enable the 2D-to-3D transformation by strand displacement.

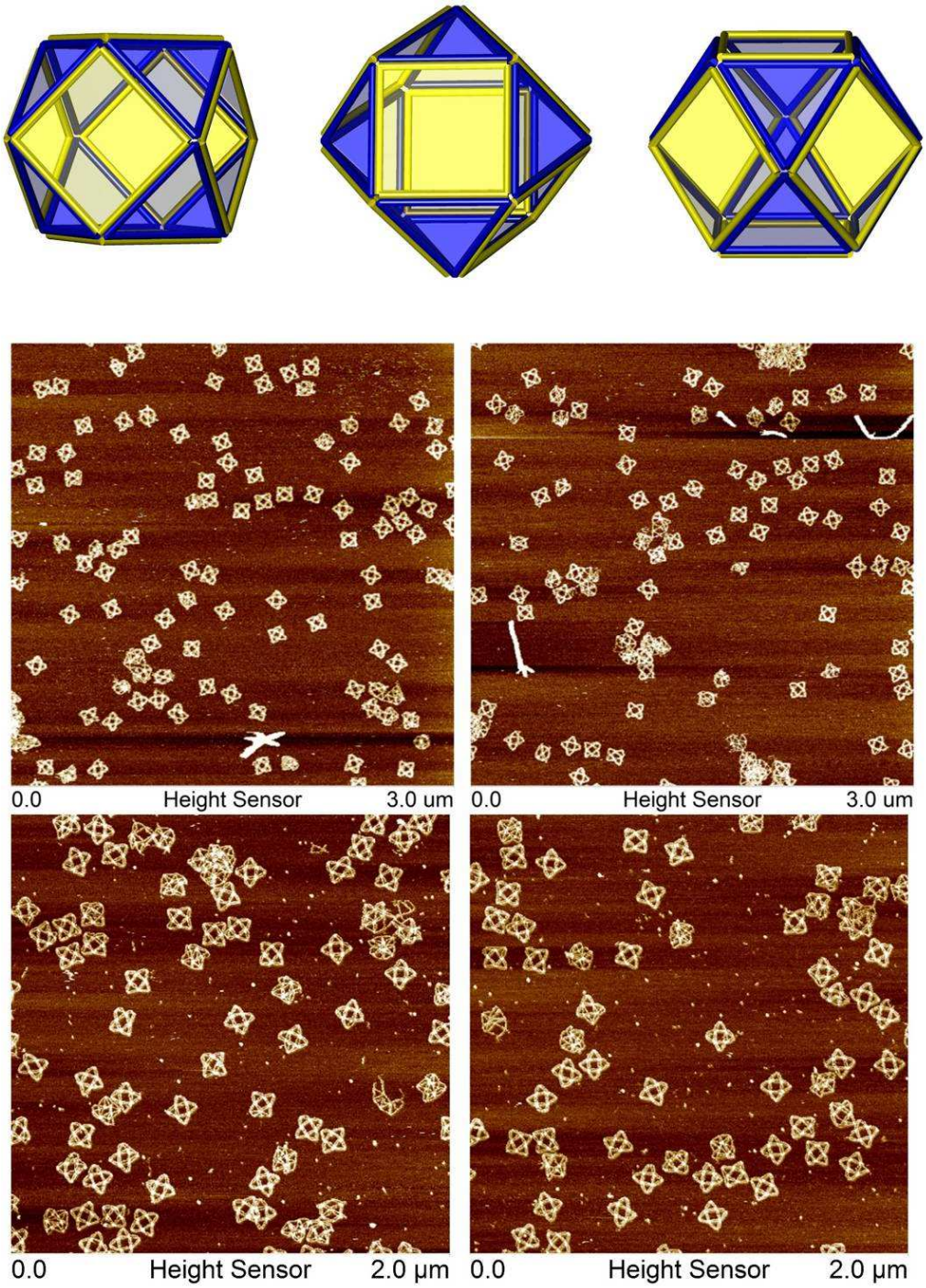


Fig. S62. AFM images of the annealed 3D closed structure of the cuboctahedron

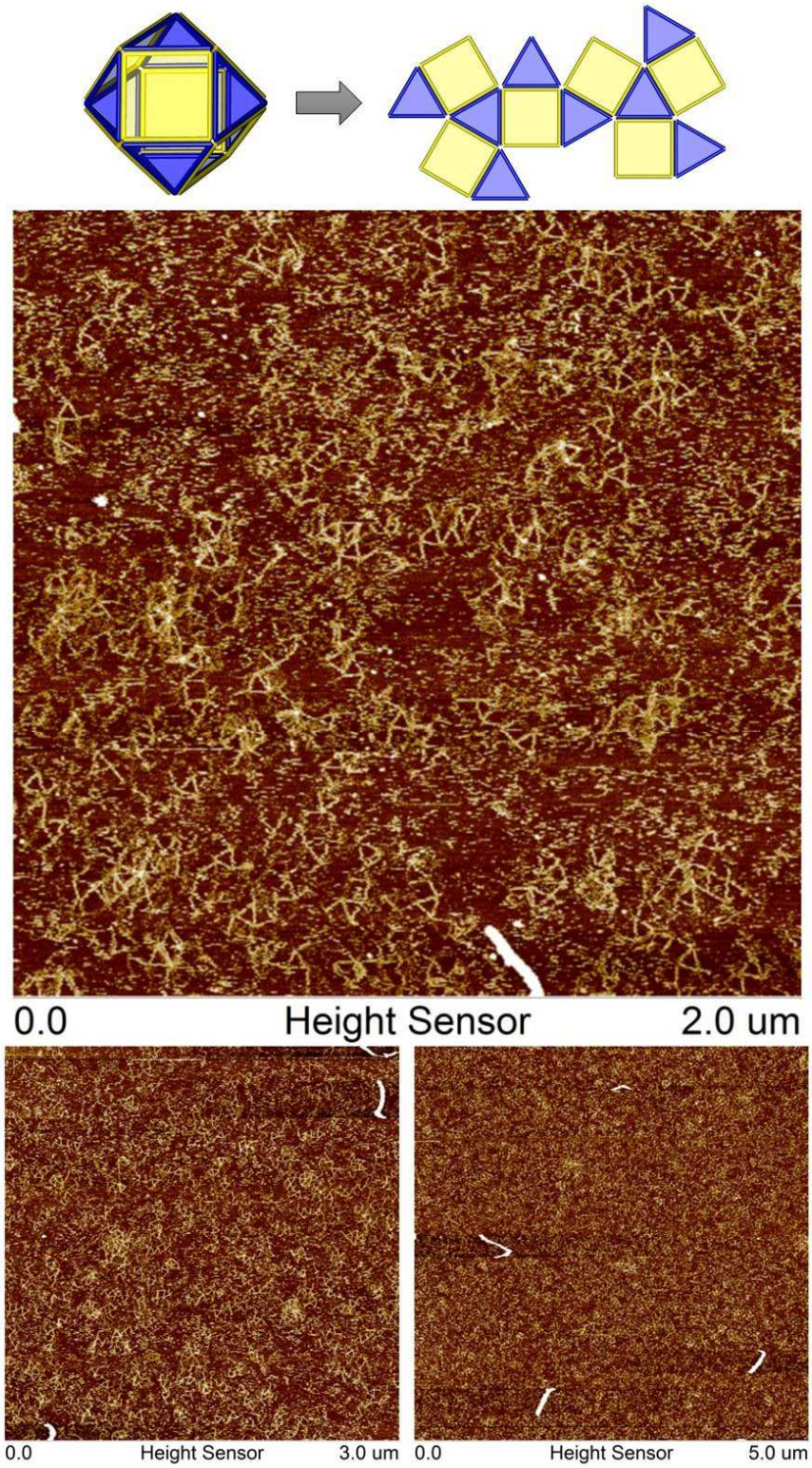


Fig. S63. AFM images of the 2D open net of the cuboctahedron reconfigured from the 3D structure

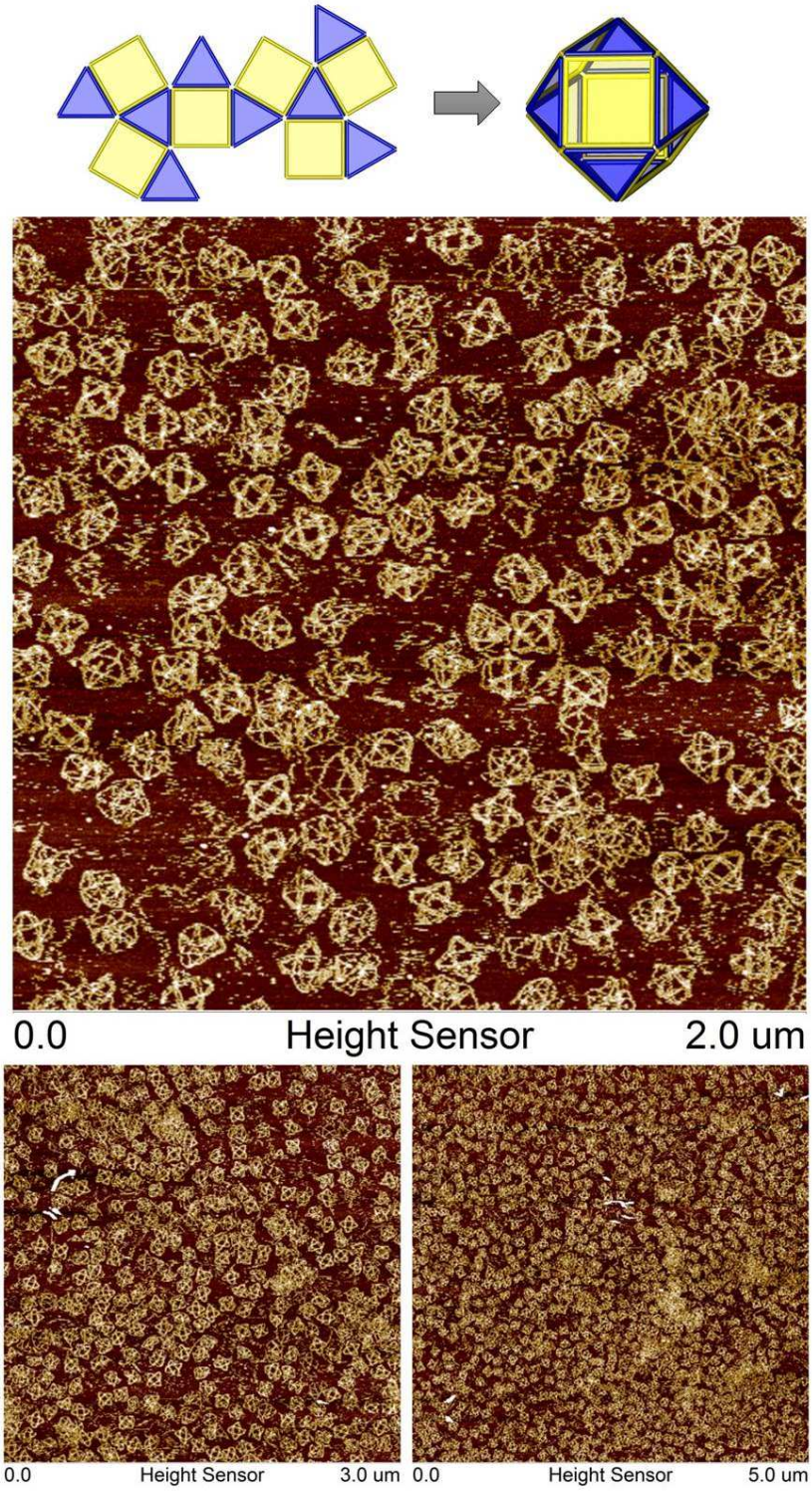
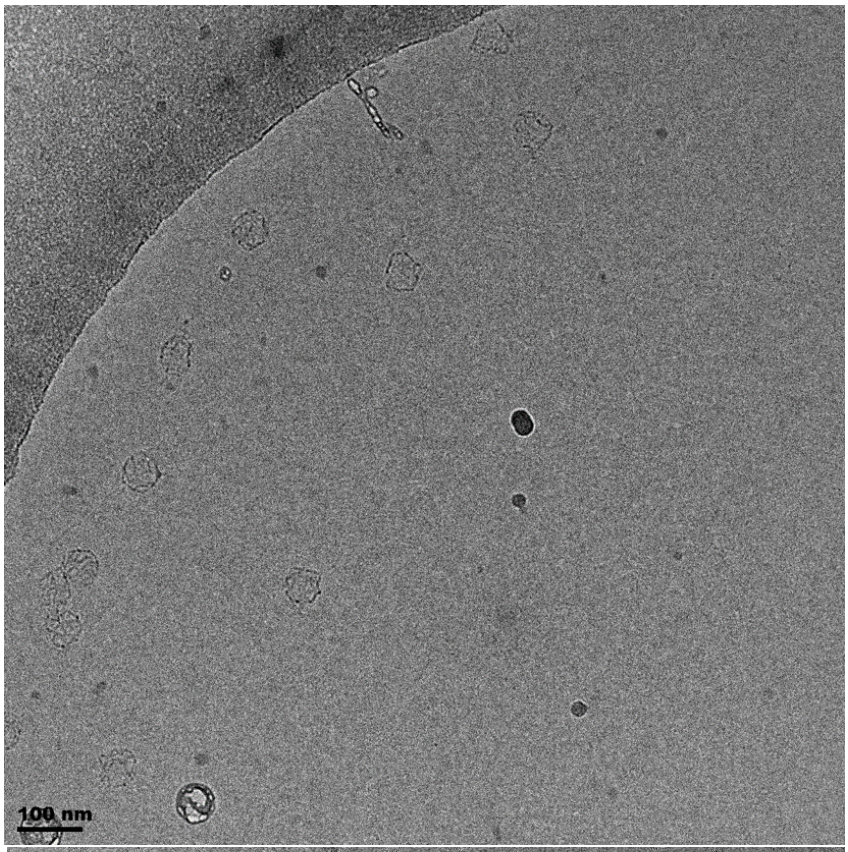
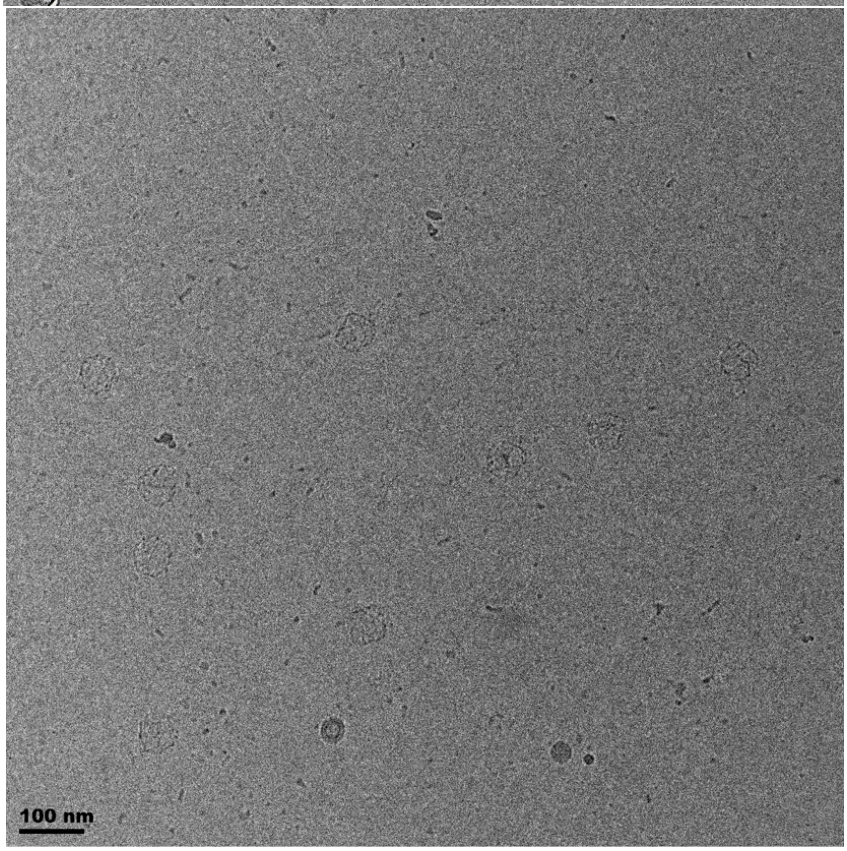


Fig. S64. AFM images of the 3D closed structure of the cuboctahedron reconfigured from the 2D open net


100 nm


100 nm

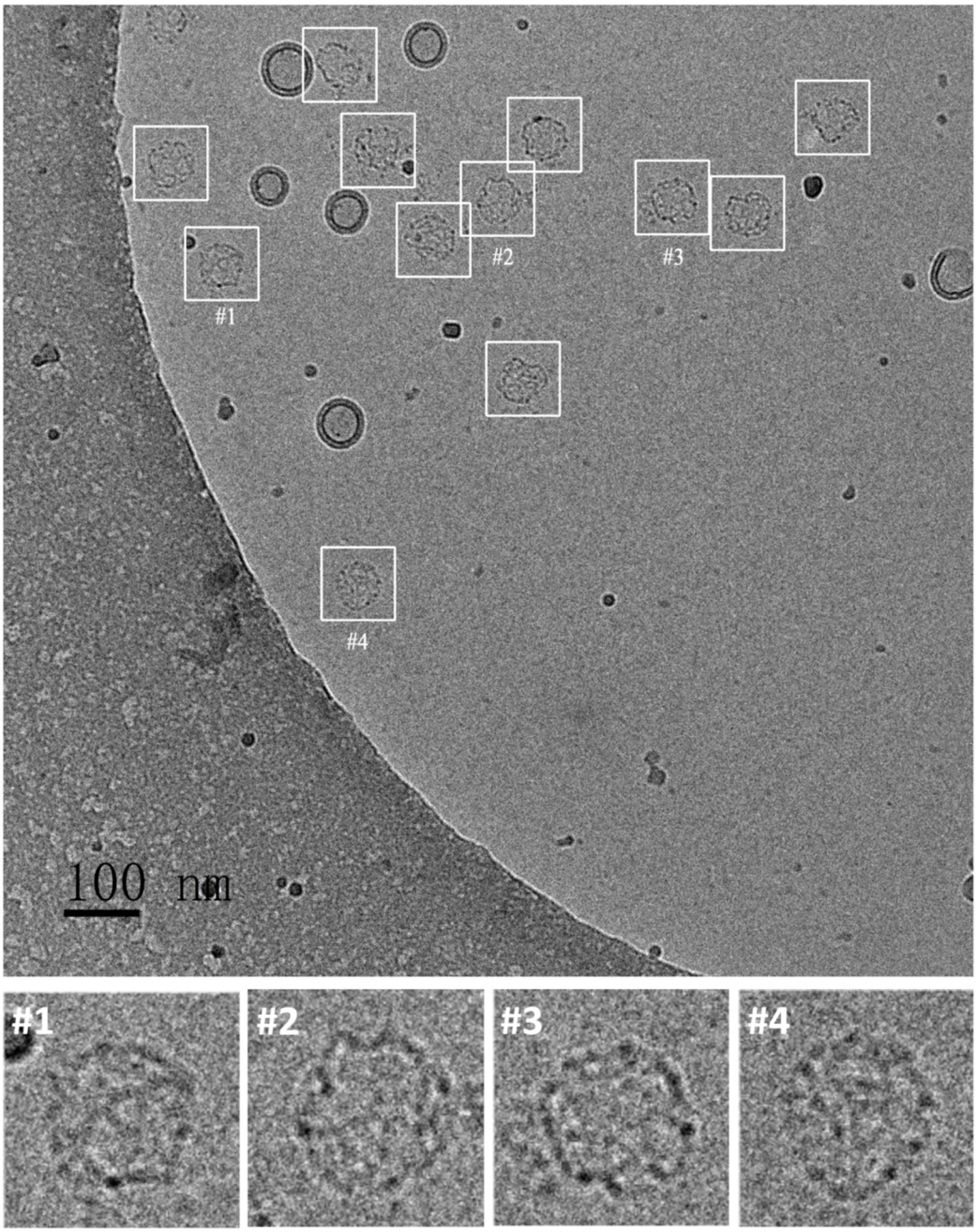


Fig. S65. Typical cryo-EM images of the DNA snub cube. Four numbered particles are shown as zoom-in images.

Section 5. Design Details and Sequences

In this section, the design details and staple strands for all the frame origami are shown. For each structure, an image generated from Tiamat software illustrates the details of the DNA design, and the sequences of the staple strands are listed in the tables. In the images for the design details, the black strand is the M13mp18 scaffold strand. For the two-scaffold structures (figs. S65, S69, and S73), the long dark-blue strand is the PhiX174 scaffold strand. The staple strands are depicted in other colors.

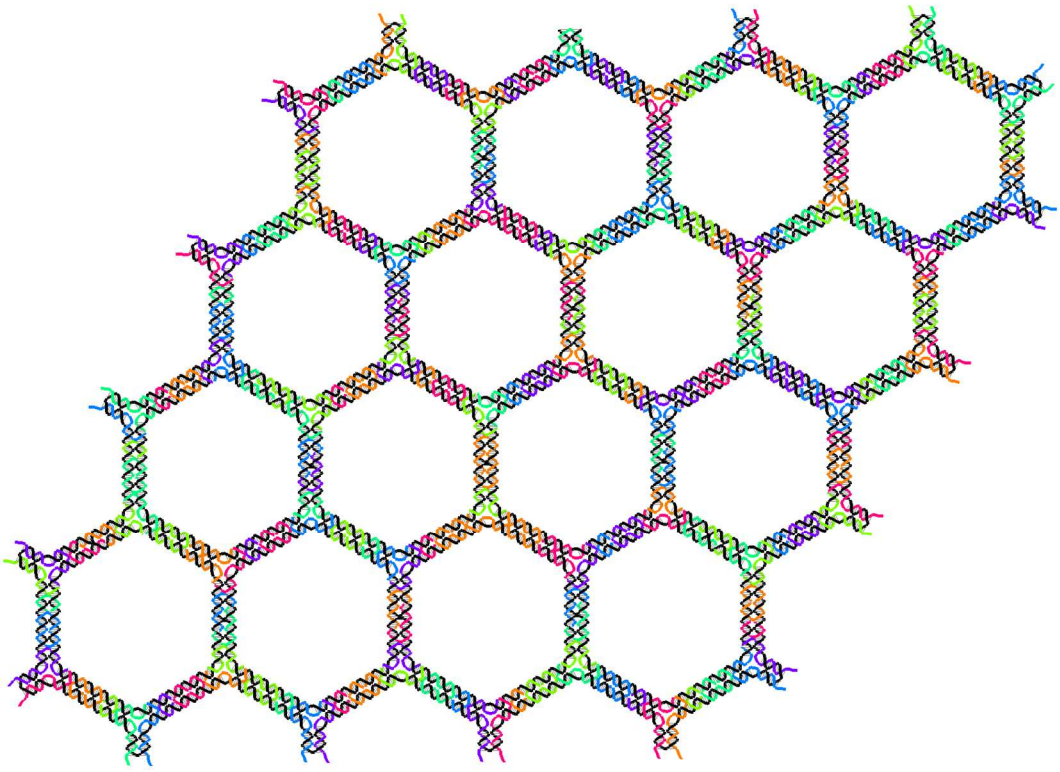


Fig. S66. Design pattern for the honeycomb Platonic tiling

Table S2. Sequences of the honeycomb Platonic tiling

Number	Sequence
Platonic-honeycomb-1	TTCATATTTATCAAAATCACCAGTCAGGACTTTGTTGGGAAGAAAAATTTTT
Platonic-honeycomb-2	GAGCCGTTTCCACCCTCAGAGCCGCCACCATTTGAACC
Platonic-honeycomb-3	GAACCGCCTCCTTATGCGAT
Platonic-honeycomb-4	AGCCACCACCGTTTAAGAACTGGCTCATTATACCGGAACCAG
Platonic-honeycomb-5	GTTTTTCACATTAATTGTTTCGTTGCGCTCGCCAG
Platonic-honeycomb-6	CTGCATTTTTAATGAATCGTTGGGCGCCAGTTTGGTG
Platonic-honeycomb-7	ACCCTCAGAACCGCCACCCTCAGTCGGGAAA
Platonic-honeycomb-8	CCTGTCGTACTGCCCGCTTTCCAGAGCCACC
Platonic-honeycomb-9	TCACCGTTTACTTGAGCCAGCACCATTACCTTTATTAG
Platonic-honeycomb-10	CAAGGCAAGATTAGTTTTTGCTATTTTGCCACCG
Platonic-honeycomb-11	CCAATTGAGGGAGGGATTTAGGTAAATATAACGC
Platonic-honeycomb-12	TAACGATTTGCGTCTTTCCAGCCATATTATTTTTTATC
Platonic-honeycomb-13	AGACGGTTTGAGAATTAACTGAGCGCTAATTTTATCAG
Platonic-honeycomb-14	AGCAAGTTTCAAATCAGATCGAGGCGTTTTTTTAGCGA
Platonic-honeycomb-15	ACCTCCAGCACCGTAATTTTCAGTAGCGACCAAT

Platonic-honeycomb-16	AGCCTTAAATCCGGAAACGT
Platonic-honeycomb-17	GGAGGTTTTGACACCAATGAAACCATCGATAGCCGACTTGCG
Platonic-honeycomb-18	GGTGAATTATACCCAGCTAC
Platonic-honeycomb-19	TATTCATTAAAAATTTTATCCTGAATCTTACCTGACGGAAAT
Platonic-honeycomb-20	CACCAGTATTTGGGAATTAGAGCCGAGGCGG
Platonic-honeycomb-21	TTTGCGTAGCCAACGCGCGGGGAAGCAAAAT
Platonic-honeycomb-22	TGAAAGAGCCGGAAGCTTTATAAAGTGTATGCCC
Platonic-honeycomb-23	TTCACCTTTGCCTGGCCCTATAGCTGTTTCTTTCTGTG
Platonic-honeycomb-24	AAGAAGGTCGACTCTATTTGAGGATCCCCTTGCC
Platonic-honeycomb-25	CCAGCATTTGGCGAAAATCATCCCTTATAATTTATCAA
Platonic-honeycomb-26	CCGTCTTTTATCAGGGCGAAAGTTTTTTGGTTTGGTCG
Platonic-honeycomb-27	AACAGCTGATAAGCCTGGGG
Platonic-honeycomb-28	GTGAGACGGGCTGCCTAATGAGTGAGCTAACTCTTTTCACCA
Platonic-honeycomb-29	AATCATGGTCGAGAGAGTTG
Platonic-honeycomb-30	CTCGAATTCGTCAGCAAGCGGTCCACGCTGGTGGGTACCGAG
Platonic-honeycomb-31	TAATAATTTCGGAATACCCTAAGAGCAAGATTTAACAA
Platonic-honeycomb-32	TGAAATAAGAAAAGTATTTAGCAGATAGCCGCAA
Platonic-honeycomb-33	CTATGGTTTTTGCTTTGACCAGAGCGGGAGTTTCTAAA
Platonic-honeycomb-34	AGAGAACGCAGTATGTTTTTAGCAAACGTGCATT
Platonic-honeycomb-35	AGCCCAATAAAAAAGAACTG
Platonic-honeycomb-36	GAATTGAGTTAGCATGATTAAGACTCCTTATTTAACCCACAA
Platonic-honeycomb-37	GCCCTTTTTAGCAATAGCTATCTCTTTCCTC
Platonic-honeycomb-38	GTTAGAATGAGCACGTATAACGTGTACCGAA
Platonic-honeycomb-39	CGAGGAAACGAACAAAGTTACCAGACCATCA
Platonic-honeycomb-40	CCCAAATCTGGCCCACTACGTGAAAGGAAAC
Platonic-honeycomb-41	GGCCTACCGCTTCTGGTTTTGCCGGAAACAGCCC
Platonic-honeycomb-42	CCGATTTTTTAGAGCTTGAGGACGACGACATTTGTATC
Platonic-honeycomb-43	CGGATTTTTCTCCGTGGGAGTCACGTTGGTTTTGTAGA
Platonic-honeycomb-44	TGGGCAAGGGAAGAAATTTGCGAAAGGAGCCCGT
Platonic-honeycomb-45	AGGTGGCAACTGTTGGTTTGAAGGGCGATAAAAA
Platonic-honeycomb-46	CCCTAAAGGGCAGGCAAAGC
Platonic-honeycomb-47	CTAAATCGGAAGCCATTCGCCATTCAGGCTGCCCGTAAAGCA
Platonic-honeycomb-48	CAGTTTGAGGCGGGGAAAGC
Platonic-honeycomb-49	CGTGCATCTGCCGGCGAACGTGGCGAGAAAGGGCATCGTAAC
Platonic-honeycomb-50	TGCGCGTTTTAACCACCACCAATAGGAACGTTTCCATCAAAAATAATTTTTT
Platonic-honeycomb-51	TTTTCGCGTCTGGCCTTCCTTTTGTAGCCAGCTCACGC
Platonic-honeycomb-52	CAGGATATTTTGTTAATTTAATTCGCATTGCGTA
Platonic-honeycomb-53	ATTTTTTAACACCCGCCGCG
Platonic-honeycomb-54	TAAATCAGCTCCTTAATGCGCCGCTACAGGGCAAATTTTTGT
Platonic-honeycomb-55	GTGTAGCGGTTTCATCAACA

Platonic-honeycomb-56	GGCGCTGGCAATTAAATGTGAGCGAGTAACAACGGGCGCTAG
Platonic-honeycomb-57	CGTTTTTTTCATCGGCATTGTTGAGATTTATTTGGAAT
Platonic-honeycomb-58	ACCACATCGAGAACAATTTGCAAGCCGTTTAGCG
Platonic-honeycomb-59	TTTTCTACGTTAATAAAACGTTTAACTAACGGAATCTT
Platonic-honeycomb-60	CTTTCCTTTTTATCATTCCGGAATTACGAGTTTGCATAGTAAGAGCAATTTT
Platonic-honeycomb-61	TACCGCACTCATTCAACTAA
Platonic-honeycomb-62	ATTAAACCAAGTGCAGATACATAACGCCAAAAAAGAACGGGT
Platonic-honeycomb-63	CGTCAGACTGTTTATTTTCA
Platonic-honeycomb-64	TTTGCCTTTAGTCGTAGGAATCATTACCGCGCCAGAATCAAG
Platonic-honeycomb-65	AGATTCATCATTCGGTCATA
Platonic-honeycomb-66	TACAGGTAGAAGCCCCCTTATTAGCGTTTGCCACAACATTAT
Platonic-honeycomb-67	TTTTTCGAGCTTCAAAGCGATTTACCAGACCGGCGCTCA
Platonic-honeycomb-68	ACAGTTTTAGGGCTTAATAAAGACTTCAATTTATATCGCGTTTTAATTTTT
Platonic-honeycomb-69	TTTTAAATATTCATTGAATCTTTCCCCTCAAATGACAAAAGGTATTTAAGTA
Platonic-honeycomb-70	ATTCTAGCGTCCAATATTTCTGCGGAATCGTCATTTTT
Platonic-honeycomb-71	TTTTCACTATCATAACCCTCTTTGTTTACCAGAGCTGT
Platonic-honeycomb-72	TTTTGATTCAAAAGGGTGAGTTTAAAGGCCGGATAATTA
Platonic-honeycomb-73	CATTTTTTAACAATTTCACAATGCCTGAGTTTTAATGTGTAGGTAAATTTT
Platonic-honeycomb-74	TCGCAATTTATGGTCAATAATCGGTTGTACTTTCAAAA
Platonic-honeycomb-75	ACATTGCTGAGAAGAGTTTTCAATAGTGAACATT
Platonic-honeycomb-76	AACTAAGTTAATTTCATTTTCTT
Platonic-honeycomb-77	CTGACCCTTAATTGCTGTTTAATATAATGCTGTAGTTTT
Platonic-honeycomb-78	GGTTTGAAATAATTTTTGCGGATGGCTTAGAGTAAATTTAAT
Platonic-honeycomb-79	CCGACCGTGTATAAGAGGTC
Platonic-honeycomb-80	TATGCTACCTTTAATTTTTGCTCCTTTTGGATAA
Platonic-honeycomb-81	ATAAGGTTTCGTTAAATAAAGCCTGTTTAGTTTTATCA
Platonic-honeycomb-82	CTAGAAAAGAATAAACACCGGAATAGGAAGG
Platonic-honeycomb-83	TTATCTAAAGTTGAAAGGAATTGCATAATTA
Platonic-honeycomb-84	AGTTGTTTGCAAATCAACAATATCTTTAGTTTGAGCA
Platonic-honeycomb-85	CTAACAAATCTAAAGCTTTATCACCTTGCCTGGTC
Platonic-honeycomb-86	TTAGAGCCGTCTGAGAGCCAGCAGCAAATGAAAACTAATAGA
Platonic-honeycomb-87	AATAGATAATAGTGCCACGC
Platonic-honeycomb-88	GCAACACATTTGAGGATTTTTTAGAAGTATTAAA
Platonic-honeycomb-89	AATACCTTTGAACGAACCAAGTATTAACACTTTCGCCT
Platonic-honeycomb-90	AACATCGCCATTAGACTTTA
Platonic-honeycomb-91	GATAGCCCTAACAAACAATTCGACAACTCGTATGCGCGAACT
Platonic-honeycomb-92	GAAACTGAATGGCTATTTTTAGTCTTTAATTAAA
Platonic-honeycomb-93	TCCTTTTTTGCCCGAACGTCATTTTGCGGATTTACAAA
Platonic-honeycomb-94	CTGAATTTTAATGGAAGGGCGTAAAACAGATTTAATAA
Platonic-honeycomb-95	AGAAAAAATACCTACATTTTTTTGACGCTTACTT

Platonic-honeycomb-96	TTTCAGGTTTACAACAGGAAAAACGCTCATGGTTGCGTAGAT
Platonic-honeycomb-97	ACGTCAGATGCCAGCCATTG
Platonic-honeycomb-98	TACCGAATATACAGTATTTACAGTACCTTTAATA
Platonic-honeycomb-99	ACATCATTTCTTGCCTGAGATATCCAGAACTTTAATAT
Platonic-honeycomb-100	CTTTGATTAGTTACATCGGG
Platonic-honeycomb-101	TAGCAATACTTAGAAACAATAACGGATTCGCCTTAACCGTTG
Platonic-honeycomb-102	GCAAATGATTGCTTTGTTTAATACCAAGTTCTGG
Platonic-honeycomb-103	AGCAAATTTCAAGAGAATCAGAGTCTGTCCTTTATCAC
Platonic-honeycomb-104	AATCGTAAAACTCAGTGAGGCCACCGAGTAAAGATGAACGGT
Platonic-honeycomb-105	TAGCATGTCAGTTTTTATAA
Platonic-honeycomb-106	CAGAATTTTCCTGAGAAGTATCATATGTACTTTCCCGGTTGATAATCATTTT
Platonic-honeycomb-107	TTTTGAAAAGCCCCAAAAACTTTAGGAAGATTGTACGC
Platonic-honeycomb-108	ATAAAGTACCGCTTTAAACA
Platonic-honeycomb-109	GAGAATGACCTCGAGCCAGTAATAAGAGAATGTTCAGAAAAC
Platonic-honeycomb-110	TTAGGCTTTAGAGGCATTTATAAATCAAAATTTATCAGGTCTTTACCCTTTT
Platonic-honeycomb-111	TTTTTGACTATTATAGTCAGTTTAAGCAAAGCGGTAAT
Platonic-honeycomb-112	TTTTGGAGAGGGTAGCTATTTTTTTTGAGAGATTCAAT
Platonic-honeycomb-113	TACCTGTTTAGCAAAAGAAACCGTTCTAGCTTTTGATAAATTAATGCCTTTT
Platonic-honeycomb-114	ATTATTCATTCTACAAAGGC
Platonic-honeycomb-115	CGCAGAGGCGATATCAGGTCATTGCCTGAGAGTACAAAATCG
Platonic-honeycomb-116	ATGATATTCAGATGATGAAA
Platonic-honeycomb-117	TCACCATCAATCAAACATCAAGAAAACAAAATGACAGTCAAA
Platonic-honeycomb-118	CTTTTTTAATGAACCCTCATATATTTTAAATGTTTGAATTAC
Platonic-honeycomb-119	GAAACAGTACAAATTTTTAG
Platonic-honeycomb-120	AATCCTTTATTTCAACTTTGCAAGGATAAATAAA
Platonic-honeycomb-121	TCAATATTTTATGTGAGTGATTAATTTTCCTTTCTTAG
Platonic-honeycomb-122	CGCTATTAAATAACCTTGCTTCTGTCAAAAT
Platonic-honeycomb-123	TATTTGCATTAGAACCTACCATATAAATCGT
Platonic-honeycomb-124	GGGAGAAGCCTTGAAAACAT
Platonic-honeycomb-125	AATACTTTTGCAGCGATAGCTTAGATTAAGACATGACCCTGT
Platonic-honeycomb-126	ACGCCAACATGATTGCATCA
Platonic-honeycomb-127	CATATTTAACAAAAAGATTAAGAGGAAGCCCGTGAGAATCGC
Platonic-honeycomb-128	ACAGGAACGGTATAAGCAAA
Platonic-honeycomb-129	AGGGATTTTAGTATTTAAATTGTAAACGTTAAGGCCGATTAA
Platonic-honeycomb-130	TGCTGGTATAGAAGAACTCAAACTAAGTCAG
Platonic-honeycomb-131	AGGGTAATTGAACACCCTGAACAATCGGCCT
Platonic-honeycomb-132	AGGCGGTCCCAGCAGAAGATAAAAGGTATTC
Platonic-honeycomb-133	TAAGAACGATAGAAGGCTTATCCCAGAGGTG
Platonic-honeycomb-134	AATAGATAACCCTCAATCAATATTGAACCTC
Platonic-honeycomb-135	AAATATCAAAGTCCTGAACAAGATTATCAAC

Platonic-honeycomb-136	GCCAGGCTAATGCAGATTTACGCGCCTGTAAAAT
Platonic-honeycomb-137	AATATCTTTCCATCCTAATTTGCAAAAGAATTTGTTTT
Platonic-honeycomb-138	CGAGAGGCTTTTACGAGCAT
Platonic-honeycomb-139	AACCAAAATAGGTAGAAACCAATCAATAATCGCGACGATAAA
Platonic-honeycomb-140	AACATGTTCAAGGGGGTAAT
Platonic-honeycomb-141	CGACAATAAACAGTAAAATGTTTAGACTGGATGTCCAGACGA
Platonic-honeycomb-142	GATTAGAGAGGTTATACAAA
Platonic-honeycomb-143	CAACAGGTCAGTTCTTACCAGTATAAAGCCAAAAGCAAACTC
Platonic-honeycomb-144	TCGGCAAACTGTTTGATGGTGGTTGTCACAA
Platonic-honeycomb-145	AAATAAACAGAGCCTAATTTGCCTTCTGGCC
Platonic-honeycomb-146	AACATTATTATTAATTTTAAAAGTATGTAAA
Platonic-honeycomb-147	TTTTCTCAACATGTTTTAAATTTTATGC
Platonic-honeycomb-148	TGCTGATGTTGGGTTATATAACTATTTGAGT
Platonic-honeycomb-149	TAACCTTTTCCGGCTTAGGCAAATCCAATCTTTGCAAG
Platonic-honeycomb-150	ACAAAAACAGTTGATTTTTCCCAATTCTGCTTTT
Platonic-honeycomb-151	CATTCCATATGAACGCGAGA
Platonic-honeycomb-152	TCTGGAAGTTTAAACTTTTTCAAATATATTTTAAGTACGGTG
Platonic-honeycomb-153	GAGAGACTACCGAACGAGTA
Platonic-honeycomb-154	ATCATAGGTCTGATTTAGTTTGACCATTAGATATTTATCAAA
Platonic-honeycomb-155	AACAGAGAAGTAATAAAAGGGACAAGTTACA
Platonic-honeycomb-156	CACCAGTTTTCACACGACCTAGAACCCTTCTTTTGACC
Platonic-honeycomb-157	TGAAACTGATTATCAGTTTATGATGGCAAAGATT
Platonic-honeycomb-158	CATCATATTCGCGTAAGAAT
Platonic-honeycomb-159	AGCGGAATTATACGTGGCACAGACAATATTTTCACCAGAAGG
Platonic-honeycomb-160	TTACATTGGCTTCATCAATA
Platonic-honeycomb-161	AAATGGATTATTAATCCTGATTGTTTGGATTACAATCGTCTG
Platonic-honeycomb-162	TCAATAGACGGAATAAGTTTATTTTCCGAAA
Platonic-honeycomb-163	GCCTTAAAAGAAACGCTTTAAAGACACCAAAATT
Platonic-honeycomb-164	CATATGTTTGTTTACCAGCAAAAATGAAAATTTTAGCA
Platonic-honeycomb-165	GTTTAACGTCGCCAAAGACA
Platonic-honeycomb-166	AACGATTTTTTAAAGGGCGACATTCAACCGATCCAAATAAGA
Platonic-honeycomb-167	ATAACATAAAATACATAAAGGTGGCAACATATTACAGAGAGA
Platonic-honeycomb-168	ACAGGGAAGCAGAAAATACA
Platonic-honeycomb-169	CTCTTCGCTATGGACTCCAACGTCAAAGGGCGCGGTGCGGGC
Platonic-honeycomb-170	TACGCCAGCTTAAAGAACGT
Platonic-honeycomb-171	AACAAGTTTAGTCCACTATGGCGAAAGGGGTTTGATGT
Platonic-honeycomb-172	GCTGCCCCAGTCACGATTTCGTTGTAAAATTTGG
Platonic-honeycomb-173	GTTGTTCCAGCGACGGCCAG
Platonic-honeycomb-174	TAGGGTTGAGTTGCCAAGCTTGCATGCCTGCATAGCCCGAGA
Platonic-honeycomb-175	TTTTAAAAAAAAGGCTCCAATTTAAGGAGCCTTAACAG

Platonic-honeycomb-176	AACAACTTTCTAATTGTATC
Platonic-honeycomb-177	GGATTTTGCTAGGTTTATCAGCTTGCTTTCGATTTCTGTATG
Platonic-honeycomb-178	ACGTTATTTGTAAATGAATGGTGAATTTCTTTTTAAACAGCTTGATACTTTT
Platonic-honeycomb-179	TTTTCGATAGTTGCGCCGACTTTAATGACAACATCCAG
Platonic-honeycomb-180	ACGCATAACCGGATCTAAAGTTTTGTCGTCTTACCATCGCCC
Platonic-honeycomb-181	ATATATTCGGTTAGCGTAAC
Platonic-honeycomb-182	CAGACATTTGCCCTCATAGTCGCTGAGGCTTTTTGCAGGGAGTTAAAGTTTT
Platonic-honeycomb-183	TTTTGCCGCTTTTGCGGGATTTTCGTCACCCTCTTCCA
Platonic-honeycomb-184	GCCTGTAGCAAGCAGCGAAA
Platonic-honeycomb-185	CAAACTACAACGACAGCATCGGAACGAGGGTAGTCACCAGTA
Platonic-honeycomb-186	GTTTCGCAACGGCTACTTTAGAGGCTTTGCTCAG
Platonic-honeycomb-187	ACTTTTTCATGGCCACCCTCAGAACCGCCACCAGGACTAAAG
Platonic-honeycomb-188	AGGAAGTTTCCCTCAGAACC
Platonic-honeycomb-189	GGTTTATTTGTACCGCCACCATTAAACGGGTTTTAAAATACGTAATGCTTTT
Platonic-honeycomb-190	TTTTCACTACGAAGGCACCATTTACCTAAAACGCAGGA
Platonic-honeycomb-191	TCACCGTACTAAAGAGGCAA
Platonic-honeycomb-192	GAATAGGTGTAAAGAATACACTAAAACACTCAGTATAGCCCG
Platonic-honeycomb-193	TATAATCTTTGACCCCTTTCAGCGATTATAGAGA
Platonic-honeycomb-194	AAACAAAGTACTTAAGAGGCTGAGACTCCTCAACCAAGCGCG
Platonic-honeycomb-195	AACGGAGATTCATGAAAGTA
Platonic-honeycomb-196	ACCTATTTTTATTCTGAAATGTATCATCGCTTTCTGATAAATTGTGTCTTTT
Platonic-honeycomb-197	TTTTGAAATCCGCGACCTGCTTTTCCATGTTACTCGGA
Platonic-honeycomb-198	CCTGCCTATTTTAGCCGGAA
Platonic-honeycomb-199	AGTTAATGCCCCGAGGCGCAGACGGTCAATCACCGTATAAAC
Platonic-honeycomb-200	CAGTGCTAAGGGAACCGTTTAACTGACCAAACATG
Platonic-honeycomb-201	GGACAGATGAATACCGTTCCAGTAAGCGTCATCTTTGAAAGA
Platonic-honeycomb-202	CGGTGTACAGTCTCTGAATT
Platonic-honeycomb-203	AGAATGTTTGAAAGCGCAGACCAGGCGCATTTTAGGCTGGCTGACCTTTTTT
Platonic-honeycomb-204	TTTTCATCAAGAGTAATCTTTTTGACAAGAACCAAGCC
Platonic-honeycomb-205	ATCCTCATTAGGATATTCAT
Platonic-honeycomb-206	CAAACAAATAATACCCAAATCAACGTAACAAATTGATATTCA
Platonic-honeycomb-207	GGTCAGTTTACGATTGGCCGCTGCTCATTCTTTAGTGAATAAGGCTTGTTTT
Platonic-honeycomb-208	TTTTCCCTGACGAGAAACACTTTCAGAACGAGTAGGCA
Platonic-honeycomb-209	CAGGAGGTTGAGTAAATTGG
Platonic-honeycomb-210	GCCAGCATTGAGCTTGAGATGGTTTAATTTCACAGAGCCGCC
Platonic-honeycomb-211	GCTTTTTTTGATGATACAGCAGTGCCTTGATTTGTAA
Platonic-honeycomb-212	AACGGGGTGAGTGTACTGGTAATATTCCACA
Platonic-honeycomb-213	CAACATACTTGTTATCCGCTCACAAAGTTTT
Platonic-honeycomb-214	AGGATTTTTAGGATTAGCGCCGTCGAGAGGTTTGTTGA
Platonic-honeycomb-215	GATAAGTGGGGTTTTGCTCAGTAGTAACGCC

Platonic-honeycomb-216	AGGGTTTTAAGGCGATTAAGTTGGCCAGGCG
Platonic-honeycomb-217	AGCCACTTTCACCCTCATTTGTACCGTAACTTTACTGA
Platonic-honeycomb-218	GGAACCCATTCAGGGATAGCAAGCAGCCAGC
Platonic-honeycomb-219	TTTCCGGCCAGGAAGATCGCACTCCCCAATA
Platonic-honeycomb-220	TTTCAGTTTCGGAGTGAGAATAATTTTTTCTTTACGTTGAAAATCTCCTTTT
Platonic-honeycomb-221	TGCGAATAATAGAAAGGAACAACTACCGTAA
Platonic-honeycomb-222	TGGGATAGACAAACGGCGGATTGAAAGGAAT
Platonic-honeycomb-223	ACCACACTTTAATCATTTTTGTGAATTACCCTCA

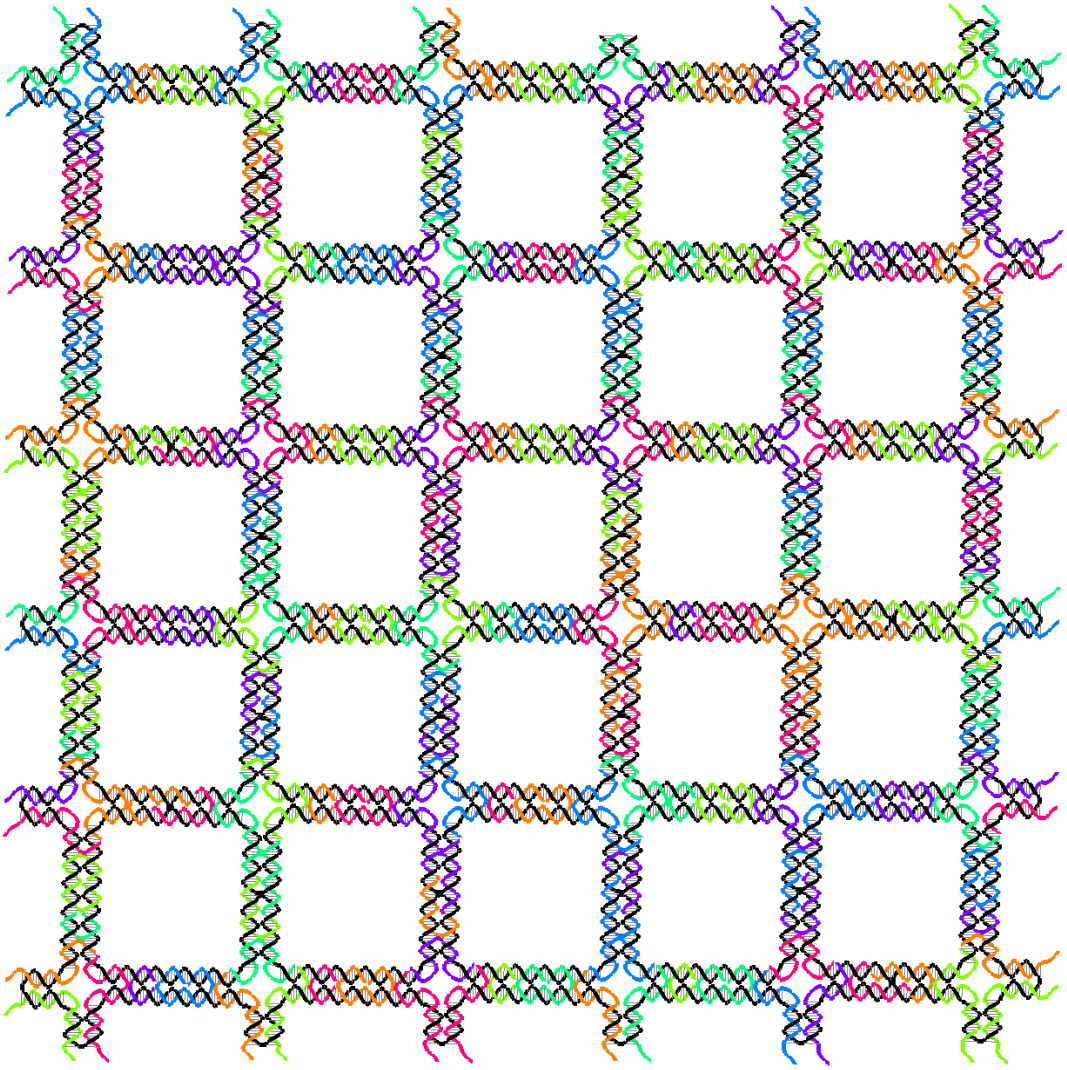


Fig. S67. Design pattern for the square Platonic tiling

Table S3. Sequences of the square Platonic tiling

Number	Sequence
Platonic-square-1	TACCTTTTTTTTTTAATGGATTTACGAGCATTTTTGTAGAAACCACCAAG
Platonic-square-2	CCCATCCTAAAACAGTACAT
Platonic-square-3	AAAAATAATATAAATCAATATATGTGAGTGAACCTGAACAAG
Platonic-square-4	TCAACATTTTATAGATAAGTTAACCTTGCTTTTTTCTGTAAATCGGAAAA
Platonic-square-5	AGATTATTTTGAGCCGTCAAAATAAAAGGGATTTTCATTCTGGCCTAAGA
Platonic-square-6	ATACGTTTTTGGCACAGACATTGAAAGGAATTTTTTGAGGAAGGTCTAAT
Platonic-square-7	TATAACTTTTTATATGTAAATGAGAATCGCCTTTTATATTTAACATTCGA
Platonic-square-8	TCAAGATTTTGTAATCTTGAAAAAAAAGGCTTTTTCCAAAAGGAGTTCGA
Platonic-square-9	GGTGAATTTTTTTCTTAAACTGAAAGAGGACTTTTAGATGAACGGCTTCA
Platonic-square-10	TTAAAGTTTTGCCGCTTTTGTCATCGCCTGATTTTTAAATTGTGTGCCGG
Platonic-square-11	AGCTTGCTCCTTTAATTGTATCGTTCTGACC
Platonic-square-12	TGAAAGCGAACAGAGATAGAACCCGTTTATC
Platonic-square-13	AAGACAAAGAATAAAGCCAACGCTCAACAGTAATCCAATCGC
Platonic-square-14	AGGGCTTAATTGCTGATGCA
Platonic-square-15	GCCAGTTTTTAATAAGAGAATTATCAAAATCTTTTATAGGTCTGAGGTTA
Platonic-square-16	AGAATATTTTACATAAAAACACACCCTGAACTTTTAAAGTCAGAGACATT
Platonic-square-17	ACAAATTTTTTCTTACCAGTACGCGAGAAAATTTTCTTTTTCAAAATTTA
Platonic-square-18	AAATATTTTTCAAACCCTCAGCGAACTGATATTTTGCCCTAAAACCAGAA
Platonic-square-19	GAGGCATTACGCCAACATGTAATCGGGAGAA
Platonic-square-20	TTAACTGAAGGGAAGCGCATTAGATTAGGCA
Platonic-square-21	GAGTAGTTTTTAAATTGGGCTGGGATTTTGCTTTTTAAACAACTTGGAAC
Platonic-square-22	AACTAATTTTAGGAATTGCGGTAACAAAGCTTTTTGCTCATTCAGAGAAC
Platonic-square-23	AGAGTTTTTTGCAGCAAGCGCCTGTTTGATGTTTTGTGGT
Platonic-square-24	GATAAATTTTACAGAGGTGACCACGCTGAGATTTTGCCAGCAGCAACCTC
Platonic-square-25	ATGGTTTTTTTGAAATACCGATAAACACCGGTTTTAATCATAATTGTTAT
Platonic-square-26	CTAAAGTTTTTTTTGTCGTCGTGAATTACCTTTTTTATGCGATTTGTTTA
Platonic-square-27	ATTTTCTGTATTGAGATGGT
Platonic-square-28	TTAGTAAATGATTAATTTCAACTTTAATCATTTTTCCAGACG
Platonic-square-29	TAAAAATAGAGTGAGAATAGAAATCAACAGT
Platonic-square-30	TTCAGCGCCGAACGAACCACCAGATCGCCAT
Platonic-square-31	TTTTCACGTTGATATTCATTACCCAAATCAACAATAATAATT
Platonic-square-32	AAAATCTCCACAAGAACCGG
Platonic-square-33	GAAACACCTGAATAAGGCTTGCCCCCAGCAG
Platonic-square-34	GCGAAAATGTCCACGCTGGTTTGCCTGACGA
Platonic-square-35	TATGCAACTATTTTTAAGAA
Platonic-square-36	TAGCCGAACATGTAGCTCAACATGTTTTAAAAAGTAAGCAGA
Platonic-square-37	TTTTTAGAGCTTAATTGCTGTTTTAATATAATGCAAGTTACCAGATTTTAGGAA
Platonic-square-38	TCGCAATTTTATGGTCAATATAAAGATTCAATTTTAAGGGTGAGACAGAG

Platonic-square-39	AGCCCAAGTACGGTGTTTTTCTGGAAGTTTCATTCTTTT
Platonic-square-40	TTTTCATATAACAGTTGATTTTTTCCCAATTCTGCCACA
Platonic-square-41	AGAATTTTTTGAGTTAAGCCATAGCTATCTTTTTTACCGA
Platonic-square-42	GAGAGATAACCGAACGAGTAGATTTAGTTTGCGCTAATATCA
Platonic-square-43	ACCATTAGATGGTAATTGAG
Platonic-square-44	AAATAGCACAATAATAAGAGCAAGTTTAGTA
Platonic-square-45	TCATATGCACTAGAAAAAGCCTGAAACAATG
Platonic-square-46	CAGTATGTAGGCGTTAAATAAGAACCGTGTG
Platonic-square-47	ATAAATATAGCAAACGTAGAAAATTATTACG
Platonic-square-48	GCATGATTTTTTAAGACTCCTACATACATAATTTTAGGTGGCAACAGGAA
Platonic-square-49	AGCCCATTTTATAGGAACCCTACAAACTACATTTTACGCCTGTAGTAGAC
Platonic-square-50	GTAAAATGTTCATTCCACAGACAGCCCTCATAGGGGGTAATA
Platonic-square-51	AGTTAGCGTAGTTTTGCCAG
Platonic-square-52	AAGCAAACTCCTAATAACGGAATACCCAAAAGACCAGACCGG
Platonic-square-53	AACAGGTCAGGGAAACGCAA
Platonic-square-54	AGTGTAGGTCGACTCTTTTTAGAGGATCCCCGGGTTTTT
Platonic-square-55	TTTTACCGAGCTCGAATTCGTTTTTAATCATGGTGGAAA
Platonic-square-56	GCCGGCTTTTGAACGTGGCGCGCTAGGGCGCTTTTTGGCA
Platonic-square-57	CATAACCCTCTAAGAACTGGCTCATTATACCAGCAACACTAT
Platonic-square-58	AGTCAGGACGGCATAGTAAG
Platonic-square-59	AGCTTGACGGCATAGCTGTT
Platonic-square-60	CCCCGATTTAGTCCTGTGTGAAATTGTTATCCTAAAGGGAGC
Platonic-square-61	AATACACAAGCGAAAGGAGCGGGAGAAAGGA
Platonic-square-62	AGGGAAGATAAAACACTCATCTTGGCAAAAG
Platonic-square-63	CTGCGCGTAACGTGCCAAGCTTGCATGCCTGCAGCGGTCACG
Platonic-square-64	CACCACACCCACGACGGCCA
Platonic-square-65	TTTTGGTTTTCCCAGTCACGTTTTACGTTGTAAAGCCGCGCTTAATTTTTGCGC
Platonic-square-66	TAACGTTTTTGCTTTCCTCGCCGATTAAAGGTTTTGATTTTAGACTGCCG
Platonic-square-67	AATATATTTTATCCTGATTGAACCGTTGTAGTTTTCAATACTTCTAAGAA
Platonic-square-68	CTCAAATTTTCTATCGGCCTCCAGAAGGAGCTTTTGGAATTATCATCATC
Platonic-square-69	TCCGAGGCGGTTTGCGTTTTTATTGGGCGCCAGGGTTTT
Platonic-square-70	TTTTTGGTTTTTCTTTTCACTTTTCAGTGAGACGCTGAG
Platonic-square-71	GCCTGGCCGGCAACAGCTGATTGCCGGAACA
Platonic-square-72	ACATTATTAATAAAACGAACTAACCTTCACC
Platonic-square-73	AAGAAATTTTAATCTACGTTACAGGTAGAAATTTTGATTCATCAGTTGAGTTTT
Platonic-square-74	TTTTCAGTCGGGAAACCTGTTTTTCGTGCCAGCTTAGCC
Platonic-square-75	CGAGATTTTTAGGGTTGAGTAAAGAACGTGGTTTTACTCC
Platonic-square-76	ACTACGTTTTTGAACCATCATAAAGCACTAATTTTATCGG
Platonic-square-77	AACCCGCTCACAATTCTTTTCACACAACATACGAGTTTT
Platonic-square-78	AACGTTTGCGTTGCGCTTTTTCACTGCCCGCTTTCTTTT

Platonic-square-79	ATCAAAAGAAGCATTAATGA
Platonic-square-80	ATCCCTTATAAATCGGCCAACGCGCGGGGAGAAATCGGCAAA
Platonic-square-81	CCAGGCGACAAGAGTCCACTATTGTTGTTCC
Platonic-square-82	AGTTTGGACATAGGCTGGCTGACTGTACAGA
Platonic-square-83	ACCCTCAGCAGAAAGTACAACGGAGATTTGTACGGGATCGTC
Platonic-square-84	CGAAAGACAGAGCGCGAAAC
Platonic-square-85	CCCAGCTTTTGATTATACCACATCGGAACGATTTTGGGTAGCAACTTTCA
Platonic-square-86	TGAGGATTTTAGTTTCCATTGCACCAACCTATTTTAAACGAAAGATGACC
Platonic-square-87	AAAGACTTGGCTACAGAGGCTTTTCACTTGC
Platonic-square-88	CTGAGTAGTTGATTAGTAATAACAGAGGACT
Platonic-square-89	AGAGCGGTAATGCCACTACGAAGAAACGGG
Platonic-square-90	TAAAATACGGAGCTAAACAGGAGGTTAGAATC
Platonic-square-91	TTTGACGAGCGAAGGGCGATCGGTGCGGGCCACTATGGTTGC
Platonic-square-92	TCTTCGCTATCAGGGCGCGT
Platonic-square-93	GAAGGCTTTTTTATCCGGTAATCAGAAAAGCTTTTCCCAAAAACAGGAAGTTTT
Platonic-square-94	AGCAAATCAGAATTTTTGTT
Platonic-square-95	GCCCAATAGCAAAATCAGCTCATTTTTTAACCTCATTACCGC
Platonic-square-96	CCGGTTGATATTCTAAGAAC
Platonic-square-97	TCATATGTACCGCGAGGCGTTTTAGCGAACCTAGCATGTCAA
Platonic-square-98	TTTTATTAATGCCGGAGAGGTTTTGTAGCTATTTAGAGC
Platonic-square-99	CTAATTTTTTTGCCAGTTACAAATAAGAAACTTTTGATTT
Platonic-square-100	AACGGGTTTTTATTAAACCAATGATGAAACATTTTAACATCAAGATGAAT
Platonic-square-101	TAATCATTTTGTGAGGCCACTAGAACCTACCTTTTATATCAAAATTGCAT
Platonic-square-102	ATCGCGTTTTCAGAGGCGAATTTTTATTTTCTTTTATCGTAGGAAAATAG
Platonic-square-103	TGAGAAGTGTACTCCAGCCAGCTTTCCGGCACGCCAGAATCC
Platonic-square-104	CCGCTTCTGGAGGAACGGTA
Platonic-square-105	AAGCAAGCCGTTATTCATTT
Platonic-square-106	TCATCGAGAACCAATTACCTGAGCAAAAGAAGAGTACCGCAC
Platonic-square-107	TTATCATTATCAATAATCGGCTGTTAGTTGC
Platonic-square-108	TATTTTGCAAGCCTTAAATCAAGATCTTTCC
Platonic-square-109	CTTGCGTTTTGGAGGTTTTGACCCAGCTACATTTTATTTT
Platonic-square-110	ATCCTGCCTGAGAGTCTTTTTGGAGCAAACAAGAGTTTT
Platonic-square-111	TTTTAATCGATGAACGGTAATTTTTCGTAAAACTCCCGA
Platonic-square-112	TCAGGTCATTGAATCTTACC
Platonic-square-113	GCGTCTTTCCTTGAGAGATCTACAAAGGCTAAACGCTAACGA
Platonic-square-114	TTTGTTGATATTCAACTTTTCGTTCTAGCTGATAATTTT
Platonic-square-115	CCCAATCCAAAATAAACAGCCATAATGCAGA
Platonic-square-116	TTTTAATTTTAAGTTTGAGTAGCCATTGCAATTTTCAGGAAAAACTCGTC
Platonic-square-117	ACGCTCAAGCTCATGGAAATACCTTCGCTGA
Platonic-square-118	AAGTTTTTGCTCCATGTTACTTACGAAATCC

Platonic-square-119	TTTTCCGGAAGCATAAAGTGTTTTTAAAGCCTGGGGCCC
Platonic-square-120	GCGACCTTGGGGTCGAGGTGCCGCCCAAATC
Platonic-square-121	TCAGGGCGATGGTGCCTAAT
Platonic-square-122	AAAACCGTCTAGAGTGAGCTAACTCACATTAACAAAGGGCGA
Platonic-square-123	AACGAGTTTTGCGCAGACGGCAACAACCATCTTTTGCCCACGCATGGGAG
Platonic-square-124	GGCTTGCAAACCGATATATTCGGACATTTTG
Platonic-square-125	CCGACAATGATCAATCATAA
Platonic-square-126	CGATAGTTGCGGGGAACCGAACTGACCAACTTAGCTTGATAC
Platonic-square-127	TGAAATTTTTGGATTATTTAAGACTTTACAATTTTACAATTCGACATTAA
Platonic-square-128	CATAGCTTTTGATAGCTTAGTGTCCAGACGATTTTCGACAATAAAGTTTA
Platonic-square-129	ACGCGCCTCAACATGTTCAGCTATTATTTAT
Platonic-square-130	AAAGTAATTCATTAAGACGC
Platonic-square-131	CGACAAAAGGTTGAGAAGAGTCAATAGTGAATTATAAAGTAC
Platonic-square-132	ACCATCAATATTAACGTCAA
Platonic-square-133	GCAGCCTTTAAAGGCCGGAGACAGTCAAATCAAATGAAAATA
Platonic-square-134	ACCACGGAATAAGCAAAGCG
Platonic-square-135	AAAGATTAAGATATAAAAGAAACGCAAAGACGATTGCATCAA
Platonic-square-136	TATTTTTTTTGTCACAATCACCGCCACCCTCTTTTAGAACCGCCAGAGAA
Platonic-square-137	TAGGTGTTTTTATCACCGTAAGACAAAAGGGTTTTCGACATTCAAGGAAA
Platonic-square-138	CCAGCGCCAACTCAGGAGGT
Platonic-square-139	CATATGGTTTATTAGTACCGCCACCCTCAGAAATAGAAAATT
Platonic-square-140	AGTTAATTAAGGTAAATATTGACCCGATTGA
Platonic-square-141	GGGAGGGTCATCTTCTGACCTAATATATTTT
Platonic-square-142	TTATTCTTTTATTAAAGGTGTGCTCAGTACCTTTTAGGCGGATAACGGAA
Platonic-square-143	TAAGAGTTTTGCTGAGACTCGGAATTAGAGCTTTTCAGCAAAATCGGAAA
Platonic-square-144	GAGCCATTTGCTCAAGAGAA
Platonic-square-145	GTCACCGACTTGGATTAGGATTAGCGGGGTTTAATTATCACC
Platonic-square-146	GCAAGGCCACCAGTAGCACCATTCCTCCGGC
Platonic-square-147	TTAGGTTGGAGACTACCTTTTTAAACCATTA
Platonic-square-148	CGTCACTTTTCAATGAAACCTAAACAGTTAATTTTTGCCCCCTGCAGTAT
Platonic-square-149	GGTAATTTTTAAGTTTTAACGAATCAAGTTTTTTTGCCTTTAGCGTCGGT
Platonic-square-150	AGTAGCGACAGGGGTCAGTG
Platonic-square-151	GCACCGTAATCCCTTGAGTAACAGTGCCCGTAATCGATAGCA
Platonic-square-152	CGGCATTTTCAGACTGTAGCGCGTTCCCTTA
Platonic-square-153	GAATCCTTTCGCTATTAATTAATTTTTTCAT
Platonic-square-154	CATAGCTTTTCCCCTTATTACTGAATTTACCTTTTGTTCCAGTAAGTACT
Platonic-square-155	CACAAATTTTCAAATAAATCCGGAACCAGAGTTTTCCACCACCGGAACCG
Platonic-square-156	TCAAAATCACCTCATTAAAG
Platonic-square-157	TCTTTTCATAACCAGAATGGAAAGCGCAGTCTGCGTTTGCCA
Platonic-square-158	ACCCTCAGAACCGCCTCCCTCAGATTTAACA

Platonic-square-159	ATTTCATTAAACAAAATTAATTACAGCCGCC
Platonic-square-160	CCACCCTTTTTCAGAGCCACGAGCCGCCGCCTTTTAGCATTGACAATATT
Platonic-square-161	AATATACACCAGAACCACCACCACACCCTC
Platonic-square-162	GATTTTTTTTCAGGTTTAACTACATCGGGAGTTTTAAACAATAACCCGTC
Platonic-square-163	AGAGCCGCCAGTAACAGTACCTTTGTCAGATG
Platonic-square-164	GAGTAACAACGGATTCGCCT
Platonic-square-165	ATACCAAGTTTCATCAACATTAAATGTGAGCGATTGCTTTGA
Platonic-square-166	TTTTCAGGGACCCCTCAAAT
Platonic-square-167	TTCAGAAAACCCCTCAGAGCCACCACCCTCAGCTTTAAACAG
Platonic-square-168	CAACAGTGGGCGGTCAGTATTAACGAGTTTC
Platonic-square-169	GTCACCAGATGTACCGTAACACTACCGCCTG
Platonic-square-170	GAGAGGGTGCATCACCTTGCTGAAATGAAAA
Platonic-square-171	ATCTAAATGATATAAGTATAGCCGTGCCGTC
Platonic-square-172	AAATCAACAGATATTTTTGA
Platonic-square-173	GGTCAGTTGGCATGGCTATTAGTCTTTAATGCATCAATATCT
Platonic-square-174	ACTAACAATATCTAAAATATCTTTATTCTGA
Platonic-square-175	AACATGAACTATTTCGGAACCTATTAGGAGC
Platonic-square-176	TAGAAGTATTCATTGGCAGA
Platonic-square-177	ATTTGAGGATTTTCACCAGTCACACGACCAGTTAGATAATAC
Platonic-square-178	CGAACGTTAACTCGTATTAAATCTTGATGAT
Platonic-square-179	ACAGGAGTGCGTCATACATGGCTTCTTTGCC
Platonic-square-180	AAAGAAACCATGCTGGTAAT
Platonic-square-181	TTTTGCGGAACATCCAGAACAATATTACCGCCAACATTATCA
Platonic-square-182	ATGGCAATTCATATTCCTGATTACAGACGAT
Platonic-square-183	TGGCCTTGGGAGGTTGAGGCAGGTTCAGATG
Platonic-square-184	ACTTCTGAATAAGTCTGTCCATCACGCAAATTTTTGGATTAT
Platonic-square-185	ATGGAAGGGTCGAGTAAAAG
Platonic-square-186	ATCGTAACCGTATTTGCACG
Platonic-square-187	TAAAGAAATTCACGTTGGTGTAGATGGGCGCTAAAACAGAAA
Platonic-square-188	TTTTATTGTATAAGCAAATATTTTTTTAAATTGTAAACGTTTT
Platonic-square-189	TTTTTTAATATTTTGTTAAATTTTATTCGCATTAATATA
Platonic-square-190	GAACGCTTTTCATCAAAAATAATTCTTTT
Platonic-square-191	TTTTGCGTCTGGCCTTCCTGTTTTTAGCCAGCTTACAAA
Platonic-square-192	GGATTCTTTTTCCGTGGGAACAAACTTTT
Platonic-square-193	TTTTGGCGGATTGACCGTAATTTTTGGGATAGGTGCGTA
Platonic-square-194	CTGCCATTTTGTTTGAGGGGACGACTTTT
Platonic-square-195	TTTTGACAGTATCGGCCTCATTTTGGAAGATCGCTTTTA
Platonic-square-196	GAAACCTTTTAGGCAAAGCGCCATTTTTT
Platonic-square-197	TTTTCGCCATTCAGGCTGCGTTTTCAACTGTTGGACGTA
Platonic-square-198	CGCTATACGCCAGCTGTTTTGCGAAAGGGGGATGTTTTT

Platonic-square-199	TTTTGCTGCAAGGCGATTAATTTTGTTGGGTAACGCCAGTTTT
Platonic-square-200	TTTTTTTTGATAAGAGGTCATTTTTTTTTGCGGATGGCTTTTT
Platonic-square-201	ACCGAGATTAGAGAGTTTTTACCTTTAATTGCTCCTTTT
Platonic-square-202	TTTTGCGTTTTAATTCGAGCTTTTTTCAAAGCGAAACTG
Platonic-square-203	GCCCGATTTTAAGACTTCAAATATCTTTT
Platonic-square-204	TTTTGTCTTTACCCTGACTATTTTTTATAGTCAGAAGTT
Platonic-square-205	TGACCATTTTTAAATCAAAAATCAGTTTT
Platonic-square-206	TTTTGAATCGTCATAAATATTTTTTCATTGAATCTAGCA
Platonic-square-207	TGGATATTTTGCGTCCAATACTGCGTTTT
Platonic-square-208	TTTTAATAGCGAGAGGCTTTTTTTTGCAAAAGAAACGAT
Platonic-square-209	CCAGACTTTTGACGATAAAAACCAATTTT
Platonic-square-210	TTTTTACATAACGCCAAAAGTTTTGAATTACGAGTTGGG
Platonic-square-211	TTTTATTTAGGAATACCACATTTTTTCAACTAATGCAGATTTT

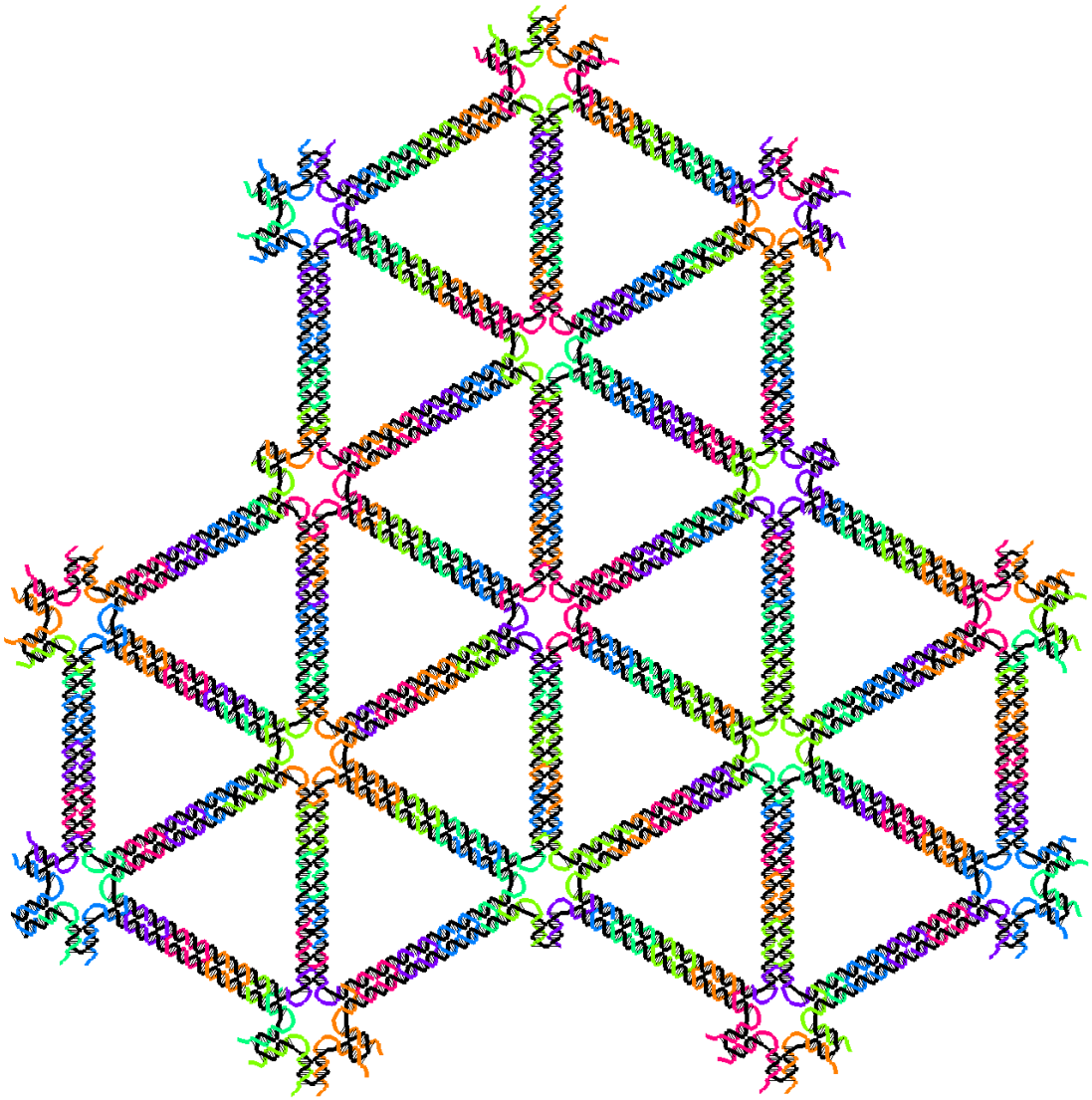


Fig. S68. Design pattern for the triangular Platonic tiling

Table S4. Sequences of the triangular Platonic tiling.

Number	Sequence
Platonic-triangular-1	CAGATGTTTTTATGGCAATTCTCTGAATAATGTTTTTGAAGGGTTAGACAGA
Platonic-triangular-2	AATAAATTTTTGAAATTGCGTAGTAACAGTACTTTTTCTTTT
Platonic-triangular-3	AAGACACATTATTTGCACGTAAAAACCTACC
Platonic-triangular-4	CGAGCATTTTTTGTAGAAACCCCTTATTACGCTTTTTAGTATGTTAGCAAAG
Platonic-triangular-5	ACAAAATTTTTGGGCGACATTCCTCAAGAGAATTTTTGGATTAGGATAACCG
Platonic-triangular-6	ATATCAAACACGGAATAAGTTTAGAAACGCA
Platonic-triangular-7	AACATATAAAATTTTGTCACAATCAATAGAAATAAAGGTGGC
Platonic-triangular-8	TTACCAGCGCCAAACGTAGAAAATACATACAATTCATATGGT

Platonic-triangular-9	TTTTAACAGTTCAGAAAACGTTTTTAGAATGACCATCACG
Platonic-triangular-10	TTGAAATTTTTATCTCCAAAACACGCATAACCTTTTTGATAT
Platonic-triangular-11	GCGTAATTTTTCGATCTAAAGTTTTCTGTATGTTTTTGGATT
Platonic-triangular-12	TTGCTGGAAGCCCGAATTTTTAGACTTCAAATATCG
Platonic-triangular-13	AAGATTAAGAAAACAACTTT
Platonic-triangular-14	GCGGAGTGAGAGCAAAGCGGATTGCATCAAACAACAGTTTCA
Platonic-triangular-15	ACAACTAAAGACCCTGACTATTATAGTCAGAAATAGAAAGGA
Platonic-triangular-16	AATAATTTTTTAAATCAAAAATCAGGTCTTTGAATTGCGAAT
Platonic-triangular-17	GTTGAGTTTTTTGTTGTTCCAGGACTCCAACGTTTTTTCAAAGGGCGGTGAA
Platonic-triangular-18	GGGTAAACACTATTAAAGAACGTGTTTGGAA
Platonic-triangular-19	AGGGTATTTTTGCAACGGCTATAAAACACTCATTTTTTCTTT
Platonic-triangular-20	CAAGAGTCATACGTAATGCCACTCATTAAAC
Platonic-triangular-21	GAGGAAGTTTCACGAAGGCACCAACCTAAAACACTTTTTCAT
Platonic-triangular-22	AAGAATACACCAGAGGCTTTGAGGACTAAAGGAAAGAGGCAA
Platonic-triangular-23	CCACCCTTTTTTCAGAACCGCATAGCAAGCCCTTTTTAATAGGAACCATTTA
Platonic-triangular-24	CTTAATGACCCTCATTTTCAGGGCACCCTCA
Platonic-triangular-25	CCATCATTTTTCCCAAATCAACGAAAGGAGCGTTTTTGGCGCTAGGGTCGTT
Platonic-triangular-26	AGAATCTTTTTAGAGCGGGAGATTGCCCTTCATTTTTCCGCCTGGCCATAGG
Platonic-triangular-27	GAGCCACCCGCCGCTACAGGGCGCCGCCGCG
Platonic-triangular-28	AACCACCACACCGTACTATGGTTGCTTTGACGCGCTGCGCGT
Platonic-triangular-29	CGTGCTTTCCCGCTGGCAAGTGTAGCGGTCAAGCACGTATAA
Platonic-triangular-30	ACCTGCTTTTTTCCATGTTACAAGGGAACCGATTTTTACTGA
Platonic-triangular-31	AATCCTGGCAGACGGTCAATCATTTAGCCGG
Platonic-triangular-32	AACGAGGCTTTGATGGTGGTTCCCAGGCGAA
Platonic-triangular-33	GTTTGCCCCAGGAAATCGGCAAAATCCCTTATGTCCACGCTG
Platonic-triangular-34	ATAGCCCGAGCTGAGAGAGTTGCAGCAAGCGAAATCAAAAGA
Platonic-triangular-35	CGAGGTGGGCGATGGCCCACTACAAAAACCG
Platonic-triangular-36	TCTATCAGAATTTCTTAAACAGCCTTGCTTT
Platonic-triangular-37	CGGTTTATCAGTTGATACCGATAGTTGCGCCGTTAATTGTAT
Platonic-triangular-38	AACCATCGCCAAAAGGCTCCAAAAGGAGCCTACAATGACAAC
Platonic-triangular-39	GGGAAGAAAGGTTTTTTGGGGTCGAGGTGCCCGAGAAAGGAA
Platonic-triangular-40	AAATCGGAACCGGGGAAAGCCGGCGAACGTGGGTAAAGCACT
Platonic-triangular-41	TCTTTCCAGATTTAGAGCTTGACCTAAAGGG
Platonic-triangular-42	AGCCCCCGACGTTAGTAAATGAATTTTGTCG
Platonic-triangular-43	AATGAATTTTTTCGGCCAACGCAGTGAGGCCATTTTTCCGAGTAAAAGCAAT
Platonic-triangular-44	GCAACAGCTGCTAAACAGGA
Platonic-triangular-45	GGGATTTTAGTTCTTTTCACCAGTGAGACGGGGCCGATTAAA
Platonic-triangular-46	ACGCCAGAATTATTGGGCGCCAGGGTGGTTTACAGGAACGGT
Platonic-triangular-47	TTTTTATAATCGCGGGGAGAGGCGGTTTGCGCCTGAGAAGTG
Platonic-triangular-48	TAAGCATTTTTGATAGCCGAAAAGCAAGCCGTTTTTTTTTTATTTTCAAGCC

Platonic-triangular-49	TTAAATTTTTTCAAGATTAGTGAATCTTACCATTTTTACGCTAACGAAAATA
Platonic-triangular-50	AACACCGGCTACAATTTTATCCTTGCTATTT
Platonic-triangular-51	CTTCTGTTTTTACCTAAATTTCCAGTATAAAGTTTTTCCAAC
Platonic-triangular-52	TGCACCCAGAATCATAATTACTATAAGAATA
Platonic-triangular-53	AAGGCGTTAAAGAAAAAGCCTGTTTAGTATCATGTGATAAAT
Platonic-triangular-54	CAAATTCTTAAATGGTTTGAAATACCGACCGTATGCGTTATA
Platonic-triangular-55	AAGAGTCTAATTTGCCAGTTACAGCGTCTTT
Platonic-triangular-56	CGCTATTTTTTTAATTAATTTAGGTTGGGTTATTTTTTATAA
Platonic-triangular-57	CCAGAGCCAATAGTGAATTTATCACGCTGAG
Platonic-triangular-58	CTTAGATTAAGAAAATCATAGGTCTGAGAGACATAGCGATAG
Platonic-triangular-59	CCTCCGGCTTTCCCTTAGAATCCTTGAAAACTACCTTTTTAA
Platonic-triangular-60	AACAGCTTTTTCATATTATTTAACACCCTGAATTTTTCAAAGTCAGAAAAAG
Platonic-triangular-61	GAATTAACTGATCCCAATCCAAATAAGAAACATTAGACGGGA
Platonic-triangular-62	TTAACGTCAAAAACATAAAAACAGGGAAGCGCGATTTTTTGT
Platonic-triangular-63	TGAAACAACTTTACAGAGAGAATAATGAAAA
Platonic-triangular-64	AATTACTTTTTCTGAGCAAAAATTAATTACATTTTTTTTAAC
Platonic-triangular-65	TAGCAGCACATCAAGAAAACAAAGAAGATGA
Platonic-triangular-66	CCTTTTTAAGGGGTAATTGAGCGCTAATATCCTTACCGAAGC
Platonic-triangular-67	CCCACAAGAATCAATGAAATAGCAATAGCTATAGAGAGATAA
Platonic-triangular-68	AGGTTTAAAATAAGAGCAAGAAATGAGTTAA
Platonic-triangular-69	GCCCAATCGTCAGATGAATATACAGATTTTC
Platonic-triangular-70	CATCGAGAACCAAAGTTACC
Platonic-triangular-71	GAGGAAACGCTATTAAACCAAGTACCGCACTAGAAGGAAACC
Platonic-triangular-72	AATACCCAAACCTTATCATTCCAAGAACGGGAATAATAACGG
Platonic-triangular-73	GATTAAGACTAATCAATAATCGGCTGTCTTTAGAACTGGCAT
Platonic-triangular-74	GGAGGTTTTGATCGTAGGAATCATTACCGCGCCCGACTTGCG
Platonic-triangular-75	AGCAAATCAGAGCGAGGCGTTTTAGCGAACCTCCCAATAGCA
Platonic-triangular-76	AAAGGTAACGGTATTCTAAGAACTATAGAAG
Platonic-triangular-77	ATAAGATTTTTGAATATAAAGCGACGACAATATTTTTAACAACATGTAGTTA
Platonic-triangular-78	GCTTATCAGTAATTCTGTCCAGATACCGACA
Platonic-triangular-79	AGTAACTTTTTAGTGCCCGTATTGAGCCATTTTTTTTGGGAATTAGAGTCAG
Platonic-triangular-80	ACTGTATTTTTGCGCGTTTTCCGTTTGCCATCTTTTTTTTTCATAATCCGCC
Platonic-triangular-81	ACATTTGATAGCCCCCTTATTAGATCGGCAT
Platonic-triangular-82	GGAAGGTTTTTTTATCTAAAACCCGAACGTTATTTTTTTAAT
Platonic-triangular-83	TTTCGGTCAGGATTTAGAAGTATTAGATAAT
Platonic-triangular-84	AGAGCCGTCAATAGACTTTACAAACAATTCGACTAATAGATT
Platonic-triangular-85	AAATCCTTTGTATCTTTAGGAGCACTAACAACAACTCGTATT
Platonic-triangular-86	TAAAACAAGAGCCACCACCGGAACAAAATCA
Platonic-triangular-87	TAATGCTTTTTGCGAACTGATGCAAATGAAAATTTTTATCTA
Platonic-triangular-88	CCGGAACCGAGGTGAGGCGGTCAGCAGAAGA

Platonic-triangular-89	ACGAACCACCAGTATTAACACCGCCTGCAACAAAAATACCGA
Platonic-triangular-90	AGAGCCAGCAAGCCCTAAAACATCGCCATTAGTGCCACGCTG
Platonic-triangular-91	TCCCTCTTTTTAGAGCCGCCATATTCACAAACTTTTTAAATAAATCCCCTTG
Platonic-triangular-92	TTGGCCTTGACCCTCAGAACCGCCACCCTCAAGGTCAGACGA
Platonic-triangular-93	CCTCAGAGCCGGCATTGACAGGAGGTTGAGGCGAGCCACCAC
Platonic-triangular-94	AGATTCACCCAGAGCCGCCGCCACCACCAGA
Platonic-triangular-95	CTGAAATTTTTTGGATTATTTAATAAAAGGGATTTTTCATTC
Platonic-triangular-96	ACCACCACAGTCACACGACCAGTACATTGGC
Platonic-triangular-97	GGGGTCAGTGTCATTAAAGCCAGAATGGAAATAAGTTTTAAC
Platonic-triangular-98	TGAATTTACCGGATACAGGAGTGTACTGGTAAGCGCAGTCTC
Platonic-triangular-99	ACTTGCCTTACATGGCTTTTGATTTCCAGTA
Platonic-triangular-100	ACTTCTTTTTTTTGATTAGTAACTATCGGCCTTTTTTTGCTG
Platonic-triangular-101	AGCGTCAGAGTAGAAGAACTCAAATAACATC
Platonic-triangular-102	CGTCACCGACTAAACAGTTA
Platonic-triangular-103	CTATTTCGGATCATTAAAGGTGAATTATCACATGCCCCCTGC
Platonic-triangular-104	TGAAACATGAGTAAATATTGACGGAAATTATACCTATTATTC
Platonic-triangular-105	GGCTGAGACTCAACCGATTGAGGGAGGGAAGAAGTATTAAGA
Platonic-triangular-106	TGCCTTTAGCGCCAGCAAAATCACCAGTAGCAGAATCAAGTT
Platonic-triangular-107	TTAGCAAGGCCAGCACCGTAATCAGTAGCGACACCATTACCA
Platonic-triangular-108	AATCCTGAGAAACCATCGATAGCGGAAACGT
Platonic-triangular-109	CACCAATTTGTTTGGATTATACTATCAATAT
Platonic-triangular-110	CCATCACGTATAGCCCGGAATAGGAGGGTTG
Platonic-triangular-111	ATATAAGCAAATTAACCGTTGTAGAGTCTGT
Platonic-triangular-112	AGTGCCGTCGAGTGTATCACCGTACTCAGGAGCAGGCGGATA
Platonic-triangular-113	CCACCCTCAGTAGCGGGGTTTTGCTCAGTACGTTTAGTACCG
Platonic-triangular-114	CCAACCTGCTCATTCATTTTTGTGAATAAGGCTTGCTTTT
Platonic-triangular-115	CGTAACAAAGTTTGAAAGAGGACAGATGAACACCCAAATCAA
Platonic-triangular-116	CCAGGCGCATAGACAAGAACCGGATATTCATTGGTGTACAGA
Platonic-triangular-117	GTCGGGAAATCAAGAGTAATCTTGGCTGGCT
Platonic-triangular-118	CATTAATTGCGTTGCGTTTTTCTCACTGCCCGCATT
Platonic-triangular-119	GACCTTCACCTGTCGTGCCAGCTGCTTTCCA
Platonic-triangular-120	TTTTAGGTCATTGCCTGAGATTTTTGTCTGGAGCAATTTC
Platonic-triangular-121	ACATCAGCCCCAAAAATTTTTCAGGAAGATTGTATA
Platonic-triangular-122	TAATCAGAAAGGGAGAAACA
Platonic-triangular-123	GCCTGATTGCAATCATATGTACCCCGGTTGAATAACGGATTC
Platonic-triangular-124	AGTTACAAAATAATCGTAAAACTAGCATGTCTTTGAATACCA
Platonic-triangular-125	CGAATTATTCAACAAGAGAATCGATGAACGGTCGCGCAGAGG
Platonic-triangular-126	GCTCATTTTGCGGATGTTTTTGCTTAGAGCTTAATTTTTT
Platonic-triangular-127	CGTTTTAATTCGAGCTTTTTTTCAAAGCGAACAGTA
Platonic-triangular-128	ATTTTCGAGCCCAGACCGGA

Platonic-triangular-129	ACAGGTCAGGACATGTAATTTAGGCAGAGGCAGCAAACTCCA
Platonic-triangular-130	CCTTTAATTGTCGCCATATTTAACAACGCCAATTAGAGAGTA
Platonic-triangular-131	AAGAGGTCATACAGTAGGGCTTAATTGAGAACTCCTTTTGAT
Platonic-triangular-132	CAGCCCTCATTCAGCTAATG
Platonic-triangular-133	TGTTTATCAAACGCCTGTAGCATTCCACAGACAGAACGCGCC
Platonic-triangular-134	TCCTGAACAATCGTCACCAGTACAAACTACACAATAGATAAG
Platonic-triangular-135	TCCCATCCTACATGTACCGTAACACTGAGTTGAAAAATAATA
Platonic-triangular-136	AGCAAATATTTAAATTTTTTTGTAAACGTTAATTAT
Platonic-triangular-137	TTTAAATAATTCGCGTTTTTTCTGGCCTTCCTGTAGTTTT
Platonic-triangular-138	GCCATCAAAAAAGTTTGAGTAACATTATCATCCAATAGGAAC
Platonic-triangular-139	CATTTTTTAATTTGCGGAACAAAGAAACCACTTAAATCAGCT
Platonic-triangular-140	TAAATTTTTGCAGAAGGAGCGGAATTATCATAAAATTCGCAT
Platonic-triangular-141	CATATTCCTGATATTTTGTT
Platonic-triangular-142	GTAATGGGTGCCTAATTTTTTGAGTGAGCTAACTCA
Platonic-triangular-143	TTTTGAATTCGTAATCATGGTTTTTTCATAGCTGTATCGT
Platonic-triangular-144	TTGACGCTCATTCCTGTGTGAAATTGTTATCATACCTACATT
Platonic-triangular-145	GCTCATGGAACGCTCACAATTCCACACAACAACAGGAAAAAC
Platonic-triangular-146	AGCCATTGCATACGAGCCGGAAGCATAAAGTATATTACCGCC
Platonic-triangular-147	GTAAAGCCTGATCCAGAACA
Platonic-triangular-148	TTTTATCATTGTGAATTACCTTTTTTTATGCGATTCCGCG
Platonic-triangular-149	GACCCCAACATTATTATTTTTCAGGTAGAAAGATTCTTTT
Platonic-triangular-150	ACTAACGGAACCAGCGATTATACCAAGCGCGTAATAAAACGA
Platonic-triangular-151	AAATCTACGTAAACAAAGTACAACGGAGATTGTTGGGAAGAA
Platonic-triangular-152	CAGTCAGGACTGTATCATCGCCTGATAAATTGCTCATTATAC
Platonic-triangular-153	GTGTCGAAATTTAAGAACTG
Platonic-triangular-154	ATTCGGGGGGTAATAGTTTTTTAAAATGTTTAGACTTTTT
Platonic-triangular-155	TTTTACGAGGCATAGTAAGATTTTTGCAACACTATGAACG
Platonic-triangular-156	GACAGCATCGCATAACCCTCGTTTACCAGACCAGCAGCGAAA
Platonic-triangular-157	TCGTCACCCTGACGATAAAAACCAAAATAGCCTTTTGCGGGA
Platonic-triangular-158	TTAAAGGCCGGAGAGGCTTTTGCAAAAGAAGCTTGCAGGGAG
Platonic-triangular-159	TTTTGCCAGAGTCGCTGAGG
Platonic-triangular-160	TTTTCCTGACGAGAAACACCTTTTTAGAACGAGTAGTAAATTTT
Platonic-triangular-161	TTTTTTGGGCTTGAGATGGTTTTTTTTAATTTCAACTTTATTTT
Platonic-triangular-162	TTTTATCAGTTGAGATTTAGTTTTTGAATACCACATTCAATTTT
Platonic-triangular-163	TTTTCTAATGCAGATACATATTTTTACGCCAAAAGGAATTTTTT
Platonic-triangular-164	TTTTAATATTCATTGAATCCTTTTTCCCTCAAATGCTTTATTTT
Platonic-triangular-165	TTTTGGATAGCGTCCAATACTTTTTTGCGGAATCGTCATATTTT
Platonic-triangular-166	CTATAATATTTTCATTTTTTTTGGGGCGCGAGCTGATTTT
Platonic-triangular-167	TTTTGCTGAATATAATGCTGTTTTTTAGCTCAACATGTTTTTTT
Platonic-triangular-168	TTTTTAAATATGCAACTAAATTTTTGTACGGTGTCTGGAATTTT

Platonic-triangular-169	TTTTGTTTCATTCCATATAATTTTTCAGTTGATTCTTCAT
Platonic-triangular-170	TTTAGTTAATCCAATTCTGC
Platonic-triangular-171	TTTAGTTTGAAGAAAACTTTTTCAAATATATGAACGAGTAGA
Platonic-triangular-172	ATTTCGCAAAAATCGCAAGACAAAGAACGCGCCATTAGATAC
Platonic-triangular-173	CTGTTTAGCTTGTAAATGCTGATGCAAATCCTGGTCAATAAC
Platonic-triangular-174	TTTTAGCTATTTTTGAGAGATTTTTTCTACAAAGGCTATCTTTT
Platonic-triangular-175	TTTTTTCTAGCTGATAAATTTTTTTAATGCCGGAGAGGGTTTTT
Platonic-triangular-176	AATTTATCACCATCAATTTTTTATGATATTCAACCGTTTT
Platonic-triangular-177	TTTTAGGATAAAAATTTTTATTTTTGAACCCTCATATCGT
Platonic-triangular-178	GCTTCTGTAAATATTTTAAATGCAATGCCTGTGAATAACCTT
Platonic-triangular-179	TATATGTGAGAGTAATGTGTAGGTAAAGATTACATAAATCAA
Platonic-triangular-180	TGGAAACAGTCAAAAGGGTGAGAAAGGCCGGACCTTTTTTAA
Platonic-triangular-181	AGACAGTCAACATTTGAATT
Platonic-triangular-182	GCGGGAGAAGCTTTTTCTTTATTTCAACGCATTTT
Platonic-triangular-183	TTTTAAAGGTGGCATCAATTTTTTTCTACTAATAGTAGTAGCATTTAATACTTTT
Platonic-triangular-184	AAGCACAGGAAGATCGTTTTTCACTCCAGCCAGCTTTTTT
Platonic-triangular-185	TTTTCCAGCTTTCATCAACATTTTTTTAAATGTGAGCGAGTTTT
Platonic-triangular-186	TTTTTAACAACCCGTCGGATTTTTTTCTCCGTGGGAACAATTTT
Platonic-triangular-187	TTTTACGGCGGATTGACCGTTTTTTAATGGGATAGATTGA
Platonic-triangular-188	GTTGAAAGGAGTCACGTTGG
Platonic-triangular-189	GCATCGTAACTGGTCAGTTGGCAAATCAACATGTAGATGGGC
Platonic-triangular-190	CAGTTTGAGGATCAAACCCTCAATCAATATCCGTGCATCTGC
Platonic-triangular-191	GTATCGGCCTTCACCTTGCTGAACCTCAAATGGACGACGACA
Platonic-triangular-192	TTTTCGACTCTAGAGGATCCTTTTTCCGGGTACCGAGCTCTTTT
Platonic-triangular-193	TTTTCGGCCAGTGCCAAGCTTTTTTTGCATGCCTGCAGGTTTTT
Platonic-triangular-194	TGGCCTTCCCAGTCACTTTTTGACGTTGTAAAACGATTTT
Platonic-triangular-195	TTTTCTGTTGGGAAGGGCGATTTTTTCGGTGCGGGGTCTT
Platonic-triangular-196	ATGGCTATTACCTCTTCGCTATTACGCCAGCAATATTTTTGA
Platonic-triangular-197	TGGCACAGACTGGCGAAAGGGGGATGTGCTGGTAAGAATACG
Platonic-triangular-198	ACCTGAAAGCCAAGGCGATTAAGTTGGGTAAGAACCCTTCTG
Platonic-triangular-199	CGCCAGGGTTAACAGAGATA
Platonic-triangular-200	TTTTCAAAGCGCCATTCGCCTTTTTATTCAGGCTGCGCAATTTT
Platonic-triangular-201	TTTTTCCGGCACCGCTTCTGTTTTTGTGCCGGAAACCAGGTTTT

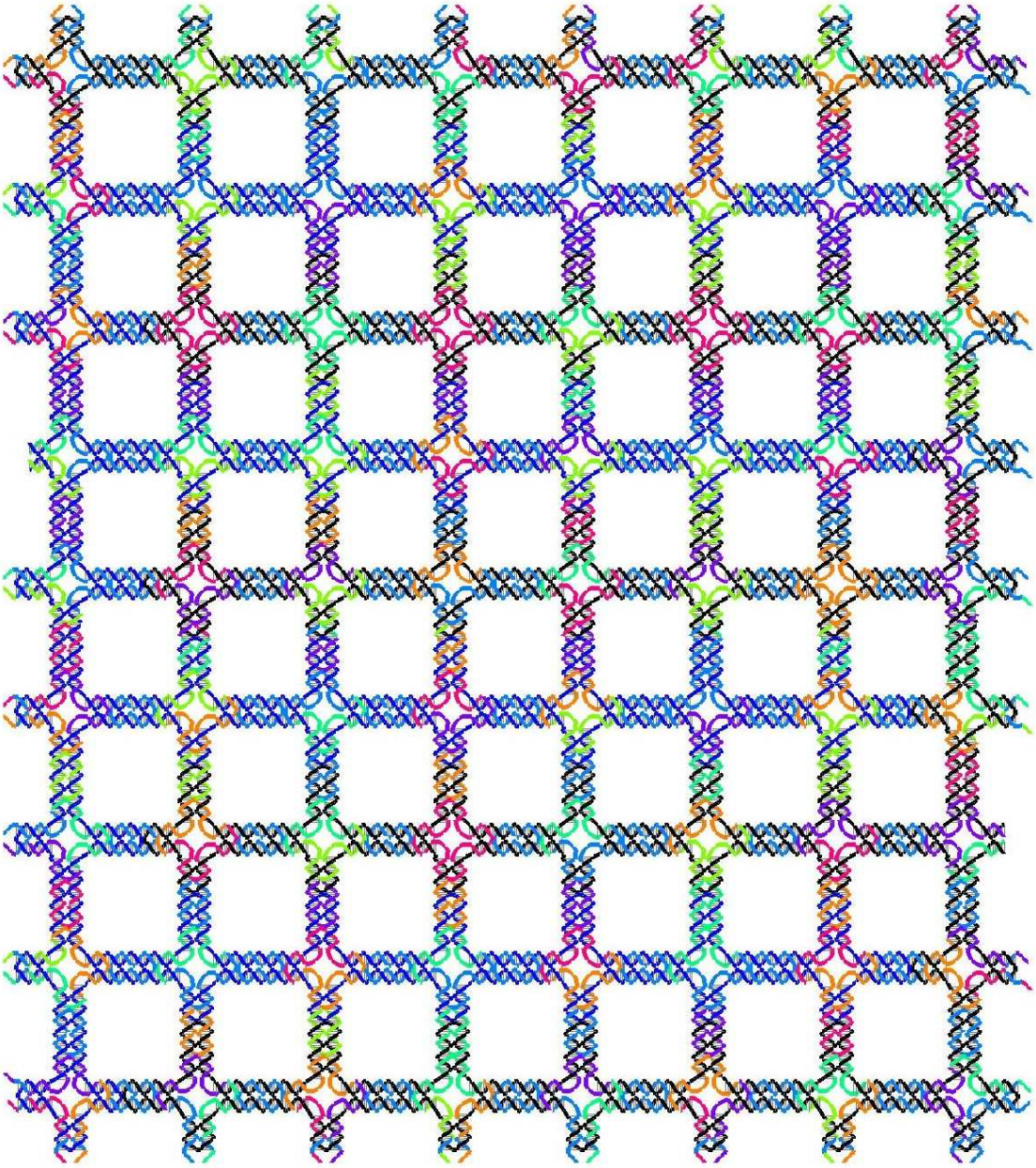


Fig. S69. Design pattern for the two-scaffold square tiling. The black strand is the M13mp18 scaffold DNA, the dark blue strand is the phiX174 scaffold DNA, and the other colorful strands are the staple strands.

Table S5. Sequences of the two-scaffold square tiling

Number	Sequence
2Scafs-8x9square-1	TTTTGCAAGGTCCATATCTGATTTTCTTTTTGTTAAGCTA

2Scafs-8x9square-2	TTTAACTTTTTGGCGGCGATAAGCAGCATCATTTTGTGACGACATCATTC
2Scafs-8x9square-3	ATGGGCTTTTATACTGTAACCCAACAGCCATTTTTATAACTGGTAGCTTTTTTT
2Scafs-8x9square-4	CACATAGAAACATAAGGCCACGTATTTTGCAACGTATTTAGC
2Scafs-8x9square-5	GGGATCTTTTGTCACCCTCAGCAACGGCTACTTTTAGAGGCTTTGCCATA
2Scafs-8x9square-6	GAATATATCGGAACGAGGGTAGCAGCGAAAGACAGCCCTTAA
2Scafs-8x9square-7	TAATTATTTTTACTCATCGCGAGGGCGTTCATTTTGCAGCCAGCTTCTCT
2Scafs-8x9square-8	AGACGCTTTTTGAGAAGAGTTATCAAAATCATTTTTAGGTCTGAGCTTTA
2Scafs-8x9square-9	CAATAGTCAACATCATAGCCAGTTTGGTCAGTTCCATGAATT
2Scafs-8x9square-10	AAACTGTTTTGCCTAACGACGATGCCCAGAGTTTTATTAGAGCGCCACTG
2Scafs-8x9square-11	AGTCATTTTTGATTGAATCGAGATTGCGATATTTTAACGGTCACATCAGC
2Scafs-8x9square-12	GTCATATTTTAGAGGTTTTACGCTGCATGAATTTTGTAATCACGTCACGA
2Scafs-8x9square-13	CAACAATTTTCTGAACGGACTCATAATCATGTTTTGTGGCGAATACCAGA
2Scafs-8x9square-14	AGGACGGTTGTTAAATTTAACCTGACTATTCATGACAAGTAA
2Scafs-8x9square-15	AAAACGTTTTAACAAGCGCAAAGAAACGCGGTTTTCACAGAATGTCCAGA
2Scafs-8x9square-16	TATATGCCATGCTCAGGAACAAGAGTAAACATAGTGTAAATG
2Scafs-8x9square-17	GGTTGGTTTTGTTATATAACCTGATGCAAATTTTTCCAATCGCAAAGGAA
2Scafs-8x9square-18	TTGAACACGAAGTACGCGTTCTTGCAAATCATTATAGGTCTG
2Scafs-8x9square-19	AGGCGGTTTTTTCCTGAATGTCAAGAAGGTGTTTTATAAGCAGGAGGCGG
2Scafs-8x9square-20	GTTGCCTTTTATACAAAACAGCACCTTTAGCTTTTGTTAAGGTACCGATA
2Scafs-8x9square-21	TTTTGGGATGTGCTGCAAGGTTTTCGATTAAGTTAAAAA
2Scafs-8x9square-22	GCCTGTTTTTTTAGTATCATAATTCTTACCATTTTGTATAAAGCCTCGCT
2Scafs-8x9square-23	CCACTGTTTTACCCTCAGCATTAGACGAATCTTTTACCAGAACGGGGCAA
2Scafs-8x9square-24	ATGCGTGAAATTTCACGCGGCAAAACATCCTTCATATATACA
2Scafs-8x9square-25	AGGGTTTTCCCCACCGGAATCATAATTACTAGGGGTAACGCC
2Scafs-8x9square-26	CGCCTGTTTTCAACAGTGCCAAGGCGTTAAATTTTTAAGAATAAAAGTCA
2Scafs-8x9square-27	CGACGTTTTTTGTAAAACGACGGCCTTTT
2Scafs-8x9square-28	ATTTCATCGGTATAAGTCAAAGGGTCGCCAGCAATATCTTCT
2Scafs-8x9square-29	AAATATTTTTATTTTAGTTAGACCTAAATTTTTTTAATGGTTTGATGAAA
2Scafs-8x9square-30	AGTCGCAGTAGAAACATACGAAGGCGCATAATGAATCTCTTT
2Scafs-8x9square-31	TCAACCTTTTCCTCAGCGGCTTTTACCGCTTTTTTCGGCGTTATAGCATA
2Scafs-8x9square-32	AGCAGCTTTTTTGCAGACCCATTTGGCGAGATTTTAAGCTCAGTCCTTGG
2Scafs-8x9square-33	ATGCAAATTAACCTCACACTCAATCTTTTATTCTTGGTCAGT
2Scafs-8x9square-34	CGTCGCATAACATCATGGTAACCTCCAAATGAAGAATATTAA
2Scafs-8x9square-35	CCTTGCTTTTTTCTGTAAATTTAATTTTCCCTTTTTTAGAATCCTGAAAC
2Scafs-8x9square-36	AATTTGAGCAGTTTTATTTGTCGTCAGAAG
2Scafs-8x9square-37	TGTTAGTCATGGAAGCTTTTGATAAAACTC
2Scafs-8x9square-38	CTGGAGTTTTTAACAGAAGTGGCGCTTTAAATTTTATAGT
2Scafs-8x9square-39	AATAACCCTGATCCGAAAGTGTTAACTTCTGCTAGATATTCA
2Scafs-8x9square-40	TTTTAGATAATTTTTCGACTCTTTTATCAGAAATAAACAA
2Scafs-8x9square-41	ATGCTTTTTTAGGGATTTTAGCGGCATGGTCTTTTAATATAACCACAAGC

2Scafs-8x9square-42	TTGAATTTTTTATGGCGAGAAAACATGATTATTTTAACTCCTAAGCAAAT
2Scafs-8x9square-43	GTTTTTTTTTGAGATGGCAGTTAAGCTCATTTTTTAGGGTTAGCCAAAGT
2Scafs-8x9square-44	AGGCATCCACGAGAACCAGCTTATCAGAAAATCGGTACGGTC
2Scafs-8x9square-45	CATTTCCGAGCATCATCTTGACAACGGAAACCATAAAATTAC
2Scafs-8x9square-46	GCAGAGTTTTGCGAATTATTCTGAGCAAAAGTTTTAAGATGATGAAGATG
2Scafs-8x9square-47	CGGAGCAGTCCAGAAAACCTACCGCGCTTCGTCAGGAGGAAG
2Scafs-8x9square-48	CATTTGGTGGTAGAAGTCGTCATAATGTCAATAGATAATTAC
2Scafs-8x9square-49	TTACATTTTTTTAACAATTTCTTTTTTAATGTTTTGAAACAGTACCTGAA
2Scafs-8x9square-50	ATTACGTTTTCCAGCTGGCGAAAGGTTTT
2Scafs-8x9square-51	AACGTCTTTTAAAAATGAAAACAGAGAGAATTTTTAACATAAAAACGAGC
2Scafs-8x9square-52	GTCTTTTTTTCCAGAGCCTATATTTATCCCATTTTATCCAAATAATGTTT
2Scafs-8x9square-53	AACGCATTTTAAGACACCACTTTTGTCACAATTTTTCAATAGAAAGCAGA
2Scafs-8x9square-54	TGTAATTTAGATTCATATGGTTTACCAGCGCCAACGCCAACA
2Scafs-8x9square-55	ACCAGGTTTTCAAAGCGCCATTCGCTTTT
2Scafs-8x9square-56	TTTTCATTCAGGCTGCGCAATTTTCTGTTGGGAAATTGA
2Scafs-8x9square-57	GAATCGTTTTCCATATTTAACAAAGACAAAATTTTGGGCGACATTCGGAA
2Scafs-8x9square-58	TGCGGGCCTCTAACGCTCAACAGTAGGGCTTAGGGCGATCGG
2Scafs-8x9square-59	GGCATTTTTTTTCGAGCCAGGTAAAGTAATTTTTTCTGTCCAGACAAAGA
2Scafs-8x9square-60	GATTAATTTTGACTCCTTATTAGCAAACGTATTTTGAAAATACATCAGCT
2Scafs-8x9square-61	ACAACATGTTACATAAAGGTGGCAACATATAGACGACAATAA
2Scafs-8x9square-62	ATCTTAAAGTACCGACAAAAGTAATAAGAGAATATAAACTTC
2Scafs-8x9square-63	AATGCATTTTGAACGCGCCTGAAAAATAATATTTTTCCCATCCTAGGCAT
2Scafs-8x9square-64	CAGATATTTTGCCGAACAAAAAACCGAGGAATTTTACGCAATAATAATCA
2Scafs-8x9square-65	TGTAGAAACCAACGGAATACCCAAAAGAACTATTTACGAGCA
2Scafs-8x9square-66	AATGGGATAAGTCCTGAACAAGTTTATCAACAATAGAAGCCT
2Scafs-8x9square-67	ATAATCTTTTGGCTGTCTTTAAGTACCGCACTTTTTCATCGAGAAGTAAG
2Scafs-8x9square-68	AAGCCCTTTTAATAATAAGAGAAATAGCAATTTTTAGCTATCTTACATCG
2Scafs-8x9square-69	TTTTTATTTTCCGAAGCCCTTTTTAAGAAAACAAGCAAGCCG
2Scafs-8x9square-70	TGGAAAAACGGGTATTAAACCCCTTATCATTCCAAGCACTGG
2Scafs-8x9square-71	TAGGAATTTTTCATTACCGCGCTTATCCGGTTTTTATTCTAAGAAGAGTT
2Scafs-8x9square-72	TAACTGTTTTAACACCCTGAGGTAATTGAGCTTTTGCTAATATCATCCCG
2Scafs-8x9square-73	TTAGCGAACCGAGAGATAACCCACAAGAATTCGCGAGGCGTT
2Scafs-8x9square-74	CGAGTGAATCAGATATAGAAGGCCCAATAGCAAGCAGTCGGC
2Scafs-8x9square-75	ACTTGCTTTTGGGAGGTTTTTGCACCCAGCTTTTTACAATTTTATAGAAT
2Scafs-8x9square-76	CCAACGCTAACAGGGAAGCGCATTAGACGGGCCTGAATCTTA
2Scafs-8x9square-77	AAAAATGATTAGTTGCTATTTGAAGCCTTAAATCAATAAAAT
2Scafs-8x9square-78	TTTTAATCTCATCTCTCTTTTTTTTTGCGTTCTGCATGAA
2Scafs-8x9square-79	TACAGTTTTTAACTTTTCCCAGCCTCTTTT
2Scafs-8x9square-80	GAAAACTTTTCACCATTACCCGTTGACGATGTTTTTAGCTTTAGGTCGGC
2Scafs-8x9square-81	ACAGGTGCCGAACAGGTTGCGCCGCCAAAACGTGTCTGTAAA

2Scafs-8x9square-82	GAAACGACTATCAAAATATAAAGCATTAACCGTCAAATTTTT
2Scafs-8x9square-83	AATAAAAATAAACAGCCATATATTTGCCAGTTACAAAGTCTG
2Scafs-8x9square-84	TTGATATAACCGTCTTCTCGTTGCGGCAAAACTGCGCCGATA
2Scafs-8x9square-85	AAATCATTTTCCAGTAGCACCAAAAGGAGCCTTTTTTTAA
2Scafs-8x9square-86	TTGTATTTAAATATGCTTTTAACTAAAGTA
2Scafs-8x9square-87	GATAAGTTTTAGGTCATTTTTGCGGTTTT
2Scafs-8x9square-88	TTTTATGGCTTAGAGCTTAATTTTTTGCTGAATAGAGGT
2Scafs-8x9square-89	GAATTTTTTTCTTAAACAGCGTTGCGCCGACTTTTAATGACAACACTTTT
2Scafs-8x9square-90	GCTCAACATGTTCGGTTTATCAGCTTGCTTTCTAATGCTGTA
2Scafs-8x9square-91	AAAAACTTTTCATTTTTCGTTAATATCAAGTTTTTTGGGGGAGCATGGGG
2Scafs-8x9square-92	AAGCCTTTTTCTACGCGATTCCTCCAGCAATTTTTCTTGAACACTCATCC
2Scafs-8x9square-93	TGTGCCAATTCATCCTTAATACCTTTCTTTTCATTGTAGCAT
2Scafs-8x9square-94	AAGGAAGGTCTATAGTGTTATCCCCTTCGGGGCGGTTTGCGA
2Scafs-8x9square-95	TAGAAATTTTGGAACAACTAATAATAATTTTTTTTTTCACGTTGAAAACG
2Scafs-8x9square-96	ATTAACTTTTTTCTCAGTAACAGCCTTATGGTTTTCCGTCAACATTAGCA
2Scafs-8x9square-97	GTGAGATTTTGTGTCAAAAACATCAGCATGATTTTGCCTGTCGCACAACG
2Scafs-8x9square-98	TTATCGAACTTTGCATTCATCAAACGCTGAAACATATCACCA
2Scafs-8x9square-99	ATGAATCACGAACGTCAGAAGCAGATACAAACTCATTTTCTG
2Scafs-8x9square-100	TTCCAGTTTTACGTTAGTAATATGGGATTTTTTTTGCTAAACAACCGGCA
2Scafs-8x9square-101	CCCTGCTTTTATACGAAAAGTATCCATCTGCTTTTTTATGGAAGCCCAAC
2Scafs-8x9square-102	CGACCATTTTATATCACGAAAAGCATTGGGATTTTTTATCATAAAGAGTA
2Scafs-8x9square-103	GATTGAGAAAACGCCTCTAATCGGTCGTCAGCAAGCATTGGG
2Scafs-8x9square-104	CTGTAGAGAGCTTGATGCGGTACAGAATCTCTTCCACATTCC
2Scafs-8x9square-105	GTACAATTTTACTACAACGCACAGACAGCCCTTTTTCATAGTTAGCAGAG
2Scafs-8x9square-106	GAAATGTTTTCCACAAGCCTCAAAGTCAAAATTTTTAATCAGCGTCCATA
2Scafs-8x9square-107	TCAGGCTTTTACACAAAAATCGGCGTGTGAATTTTTCATTAGCCTAACGA
2Scafs-8x9square-108	GGGTAATAAGTGCGACCCTCGGCAGCAAGAAGACATTCAGAA
2Scafs-8x9square-109	TTTCAGGAGCCTCGATACGCTCAATAGCAGGTTTAAGGATAG
2Scafs-8x9square-110	GAGCCATTTTCCACCCTCATCAAGCCCAATATTTTGGAACCCATGGAGGT
2Scafs-8x9square-111	ACCATATTTTAAAAAGCCTCGGGAGGGTGTCTTTTAATCCTGACGCATAC
2Scafs-8x9square-112	TCAACGTTTTCTACCTGTAGTAAAGTGCACCTTTTGCATGGAAATAGCCA
2Scafs-8x9square-113	GACAAATTAGGAAGACGGCCATTAGCTGTACGTTATTTCCTA
2Scafs-8x9square-114	ACTCAGTGAAAACATACAATTCAAGATTTGGAGGCAGAGGTT
2Scafs-8x9square-115	ATAGGTTTTTGTATCACCGTTAGTACCGCCATTTTCCCTCAGAACCGTTC
2Scafs-8x9square-116	ATACCATTTTTCAGCTTTACCAACAGCTTTATTTTTCAATACCATTAGGG
2Scafs-8x9square-117	TTTGCTTGTTCCAAGTATCGGCGTCTTTCCAGAAATCAGTAC
2Scafs-8x9square-118	TTAGGATTTTTTAGCGGGGTCAGGCGGATAATTTTGTGCCGTCGATATAA
2Scafs-8x9square-119	ACCACACCAGTGCGTACCCGACGACCAAAATGAAAAATATCA
2Scafs-8x9square-120	ATATGATTTTGAAGAGCCATAGTCAACCTCATTTTGCACTAACCTCCGAA
2Scafs-8x9square-121	TTTTAACAGCTTCTTGGGAAGTTTTTAGCGACAGCCGCCA

2Scafs-8x9square-122	TTTAGCTTTTTCCTAGACCTTTAGCATTTT
2Scafs-8x9square-123	CTTTGCAGTAGTTGGTTTTTAGTGAGTTGTTCTAGAAATATC
2Scafs-8x9square-124	CGAGCATTTTCGTATAACGTAGAATCAGAGCTTTTGGGAGCTAAATTGGG
2Scafs-8x9square-125	TTTTAATCGGAACCCTAAAGGTTTTGAGCCCCCGATTTGA
2Scafs-8x9square-126	GCTTAATTTTTGCGCCGCTACCGGCGAACGTTTTTGGCGAGAAAGGAAGGGTTTT
2Scafs-8x9square-127	ACGGGGAAAGCAGGGCGCGTACTATGGTTGCTTTAGAGCTTG
2Scafs-8x9square-128	TTTTGCAAGTGTAGCGGTCACTTTTGCTGCGCGTAGCCGC
2Scafs-8x9square-129	TTTTAAGAAAGCGAAAGGAGCTTTTGGGCGCTAGGGCGCTGTTTT
2Scafs-8x9square-130	GTCGAGTTTTGTGCCGTAAAGCACTATTTT
2Scafs-8x9square-131	TTTTTATCAGGGCGATGGCCCTTTTACTACGTGAATTTAG
2Scafs-8x9square-132	ACAGGATTTTACGGTACGCCAGTGTTTTTATTTTTAATCAGTGAGTCCAA
2Scafs-8x9square-133	ATCAAGTTTTCAGGAGGCCGATTAAAGGGATCCATCACCCAA
2Scafs-8x9square-134	TATCGGTTTTCCTTGCTGGTAATATTACCGCTTTTCAGCCATTGCGTCCA
2Scafs-8x9square-135	TTTTAAAATCCTGTTTGATGGTTTTTGGTTCCGAACAAAC
2Scafs-8x9square-136	CGTCAATTTTAGGGCGAAAAACCGTCTTTT
2Scafs-8x9square-137	CACGCATTTTAATTAACCGTTCTTTGATTAGTTTTTAATA
2Scafs-8x9square-138	TTTTAGTGTTGTTCCAGTTTGTTTTGAACAAGAGTTCCAT
2Scafs-8x9square-139	GAACGTGGACGCCACCGAGTAAAAGAGTCTGCCACTATTAAA
2Scafs-8x9square-140	ACATCTCAAAAGAATATTTTGCCCGAGATAGGGTTGTTTT
2Scafs-8x9square-141	CCCTTATAAAACTTGCCTGAGTAGAAGAACTATCGGCAAAAT
2Scafs-8x9square-142	TTTTGGCCAACGCGCGGGGAGTTTTAGGCGGTTTGAATAA
2Scafs-8x9square-143	CTACATTTTTTTTGACGCTCTGGATTATTTATTTTCATTGGCAGATTTTC
2Scafs-8x9square-144	TTGAATTTTTGGCTATTAGTCAACATACGAGTTTTCCGGAAGCATAAAGTTTTT
2Scafs-8x9square-145	TTTTGTAAAGCCTGGGGTGCTTTTCTAATGAGTGAGCTAATTTT
2Scafs-8x9square-146	TTTTCTCACATTAATTGCGTTTTTTGCGCTCACTGTATTT
2Scafs-8x9square-147	AAGGGATTTTCATTCTGGCCACCCTTCTGACTTTTCTGAAAGCGTCTGTC
2Scafs-8x9square-148	GTCGGGAAACAAGAATACGTGGCACAGACAACCCGCTTTCCA
2Scafs-8x9square-149	GTGCCATTTTGCTGCATTAATGAATCTTTT
2Scafs-8x9square-150	CCAGGGTGGTTTCACCAGTCACACGACCAGTCGTATTGGGCG
2Scafs-8x9square-151	CGCTGGTTTTTTTGCCCCAGCAGGCGTTTT
2Scafs-8x9square-152	TTTTCAGCTGATTGCCCTTCATTTTCCGCCTGGCCAATAC
2Scafs-8x9square-153	GCAGCAAGCGAACAGGAAAAACGCTCATGGACTGAGAGAGTT
2Scafs-8x9square-154	TTTTCATTTTCCAGTGAGACGGGCAATTTT
2Scafs-8x9square-155	TTCGGTTTTTCGCTGAGGCTAGGAAGCCCGATTTTAAGACTTCAAATATCGTTTT
2Scafs-8x9square-156	TTTTAAGCGGCTCACCTTTATTTTGCATCAACAGGCCACATTTT
2Scafs-8x9square-157	TTTTACCAACCAGAACGTGAATTTTAAAGCGTCCTCTGCG
2Scafs-8x9square-158	GAATAATTTTGGCTTGCCCTGGGCTTGAGATTTTTGGTTTAATTTGATTT
2Scafs-8x9square-159	GCGTGTTGAATTACCTTATGCCAACTTTAATCATTGAGCGAA
2Scafs-8x9square-160	TAAGAATTTTCTGGCTCATTATACCATTTT
2Scafs-8x9square-161	TTTTGTCAGGACGTTGGGAAGTTTTAAAAATCTACTCAGT

2Scafs-8x9square-162	GGCGCATTTTTAGGCTGGCTCCGGATATTCATTTTTTACCCAAATAACAA
2Scafs-8x9square-163	GAACTAACGGCAACGTAACAAAGCTGCTCATGTTAATAAAAC
2Scafs-8x9square-164	CATTATTTTTTACAGGTAGAAAGATTTTTT
2Scafs-8x9square-165	TTTTCATCAGTTGAGATTTAGTTTTGAATACCACAGACCA
2Scafs-8x9square-166	TGTTACTTTTTTAGCCGGAACGAACTGACCATTTTACTTTGAAAGACGCC
2Scafs-8x9square-167	CAGATACATAAGGACAGATGAACGGTGTACATTCAACTAATG
2Scafs-8x9square-168	AAAAGGTTTTAATTACGAGGCATAGTTTTT
2Scafs-8x9square-169	TTTTAAGAGCAACACTATCATTTTTAACCCTCGTTCTCCA
2Scafs-8x9square-170	GACCCCTTTTCAGCGATTATTTGTATCATCGTTTTCCTGATAAATAAAAT
2Scafs-8x9square-171	GATAAAAACCTGTGTCGAAATCCGCGACCTGTACCAGACGAC
2Scafs-8x9square-172	AGCGAGTTTTAGGCTTTTGCAAAAGATTTT
2Scafs-8x9square-173	TTTTAGTTTTGCCAGAGGGGGTTTTTAATAGTAAATCTTT
2Scafs-8x9square-174	GTTTCCTTTTATTAAACGGGAACCTAAAACGTTTTAAAGAGGCAACAATA
2Scafs-8x9square-175	GGATAGCGTCAAGAATACACTAAAACACTCAATGTTTAGACT
2Scafs-8x9square-176	CTGCGGTTTTAATCGTCATAAATATTTTTT
2Scafs-8x9square-177	TTTTCATTGAATCCCCCTCAATTTTATGCTTTAAAAGGAA
2Scafs-8x9square-178	ACGAGAATGAAGGACTAAAGACTTTTTCATGCAGTTCAGAAA
2Scafs-8x9square-179	AATCAATTTTAAATCAGGTCTTTACCTTTT
2Scafs-8x9square-180	TTTTCTGACTATTATAGTCAGTTTTAAGCAAAGCGTTTGC
2Scafs-8x9square-181	TAAAGGCCGCTGATTGCATCAAAAAGATTAAGTGCAGGGAGT
2Scafs-8x9square-182	TTTTGGAAGCAAACTCCAACATTTTGGTCAGGATTATATA
2Scafs-8x9square-183	TTTTCGTTTTAATTCGAGCTTTTTTCAAAGCGAACCAGACCTTTT
2Scafs-8x9square-184	TCATAGCACTACGAAGGCACCTAAAATACGTAATGCTGGAGG
2Scafs-8x9square-185	AATAGTCAATCATAAGGGAACCGAGGCGCAGACGGTCACGCA
2Scafs-8x9square-186	CGATAAAAGTACAACGGAGATACCAAGCGCGAAACAACCAAC
2Scafs-8x9square-187	GAAGTGAACGAGTAGTAAATTGACGAGAAACACCAGTCCGCA
2Scafs-8x9square-188	ACTGATTAATCTTGACAAGAAGACCTTCATCAAGAGAGCAGT
2Scafs-8x9square-189	CGCATAACCGAGAGAGTACCTTTAATTGCTCACCATCGCCCA
2Scafs-8x9square-190	GCTTTCAAACCAGTCCTTGACATGTGACTCATATCTCTCGTT
2Scafs-8x9square-191	CTCTACTTTTCATGAACAAAGAACGTGCCAATTTTGCATATTAAGTCACC
2Scafs-8x9square-192	CATCCAACGCTTCATGAACTTAATCCACTGTCCACTTCTCCT
2Scafs-8x9square-193	GGCATTTTTTAACACCATCCGTCAGTTTTTGTTTTACAGAATCGTGCTTC
2Scafs-8x9square-194	CGAAAACACCCACCACCGGTCGCAAAGTAAGATAGTTGATGG
2Scafs-8x9square-195	TCGAGCTTTTTGCGCAAGGATAGGTCTTTT
2Scafs-8x9square-196	TTTTGAATTTTCTCATTTTCCTTTTGCCAGCAGTCGGAGT
2Scafs-8x9square-197	CAGTCGGGAGACACTTCGATTTAATTCGTAAAGTAGTGTTAA
2Scafs-8x9square-198	AGTAGTTTTTAATTCCTGCTTTATCATTTT
2Scafs-8x9square-199	CCAAGTATCGTGCCAAGAAAATTGGTATCAGGGTTATACAAA
2Scafs-8x9square-200	TTTTAGTGCCAAGCTTGCATTTTTGCCTGCAGGTAACAC
2Scafs-8x9square-201	GGATCCCCGGGAGAGGTGAGGCGGTCAGTATTCGACTCTAGA

2Scafs-8x9square-202	AAACATTTTTCGCCATTAAAACCACCAGCAGTTTTAAGATAAAACTACCG
2Scafs-8x9square-203	AGCTCGTTTTAATTCGTAATCATGGTTTT
2Scafs-8x9square-204	TTTTTCATAGCTGTTTCCTGTTTTTGTGAAATTGCCCTA
2Scafs-8x9square-205	ACAATTCCACACTTTAATGCGCGAACTGATAGTTATCCGCTC
2Scafs-8x9square-206	AATACCTAAGAACGTCAGTGTTTTCTTCTGCGTCAGGAACGA
2Scafs-8x9square-207	CAGCAGCAAAAATACCGACCGTGTGATAAATACGCTGAGAGC
2Scafs-8x9square-208	AATCTATTTTAAGCATCACCCAATATCTGGTTTTTCAGTTGGCAATTTTC
2Scafs-8x9square-209	AAAGGAATTGGACAAAGAACGCGAGAAAACTATCAACAGTTG
2Scafs-8x9square-210	TCACTCTATCAAACCCTCAATTTGCTGAACCTCAAACTTCCG
2Scafs-8x9square-211	GGTTATTTTTCTAAAATATCGCCGTCAATAGTTTTATAATACATTGCTTA
2Scafs-8x9square-212	AAGTATTAGAAGACTACCTTTTTAACCTCCGTGAGGATTTAG
2Scafs-8x9square-213	TTGCCTAACTAATAGATTAGATTTAGGAGCACTAACTTAGTA
2Scafs-8x9square-214	CAAACATTTTATTCGACAACAATTTTAAAAGTTTTTTTGAGTAACGATTA
2Scafs-8x9square-215	GCGGAACAAATGAAAACATAGCGATAGCTTAATTATCATTTT
2Scafs-8x9square-216	CCTCAATGCCCGAACGTTATTTCGTATTAAATCCTTGTAAGG
2Scafs-8x9square-217	CACCAGTTTTAAGGAGCGGATGGCAATTCATTTTTCAATATAATCAATAA
2Scafs-8x9square-218	GATTATACTTATAAATCAATATATGTGAGTGCTGATTGTTTG
2Scafs-8x9square-219	CCAGCACTGATTATCAGATGAATTATCATCATATTCAGGAAG
2Scafs-8x9square-220	TAATGGTTTTAAGGGTTAGAAACAGAAATAATTTTAGAAATTGCGATTAA
2Scafs-8x9square-221	GTTTAACGTCAACAAACATCAAGAAAACAAATAGATTTTCAG
2Scafs-8x9square-222	CCAACGATTATTTGCACGTAAACCTACCATATCAAAGCGTCC
2Scafs-8x9square-223	AATATATTTTCAGTAACAGTTTCGCCTGATTTTTTGCTTTGAATAATCGC
2Scafs-8x9square-224	GGGACAAGAAACAATAACGGAACCTTTTACATCGGGTAAAAA
2Scafs-8x9square-225	ATAAACTTTTGTGACGATGAGTAAAAATGTCTTTTTACAGTAGAGAGAAT
2Scafs-8x9square-226	TGTCAACAAGTCAATAGCAAGGCCACGACGCCTCTTTTAAAA
2Scafs-8x9square-227	GTAACTTTTTTGACTCATGAGGACTCAGATATTTTGTAATCCACGAATGG
2Scafs-8x9square-228	AGAAAGTTTTACGGAGAGCGATCTCGAAGGATTTTGTCGCCAGCGGTTCA
2Scafs-8x9square-229	CGTACGGATTATAACCGGAGTAGTTGAAATGCGGTCTGGAAA
2Scafs-8x9square-230	AAATCGTTTTAAATCATCTTTGAGCTTAATATTTTGAGGCCAAAGGTAAT
2Scafs-8x9square-231	AAGACGTTTTACCAATCTGACCAAGATGGGATTTTAAGGTCATGCTCAGA
2Scafs-8x9square-232	TTGTTACTCGGGCATACGCTCGGCGCCAGTTCAATCCAAACT
2Scafs-8x9square-233	CCCACATTTTAAGTCCAGCGGAGCACGAGAGTTTTCGGTCAGTAGTGAAT
2Scafs-8x9square-234	ATTAGATTTTCATAATTTATGGCCGAAGCCCTTTTCTGCAATTAAGGTAT
2Scafs-8x9square-235	GGAAAGCGAGAATTGTTGACCACCTACATACCAACAGGAGCA
2Scafs-8x9square-236	ATGATGTTTTAGACAGGCCGCAATGACGGCATTTTGCAATAAACTCAAAG
2Scafs-8x9square-237	ACGAGCTTTTGCCTTTACGCCCTCGCAACGGTTTTCTGCGGACGACTTCC
2Scafs-8x9square-238	AATTCAGCGCCCAGGGCGAGCGCCAGAACGTTGTTTTCCGTA
2Scafs-8x9square-239	GCGCGTTTTTACACGCAAGGGACGCTCGACGTTTTCCATTAATAATTTTT
2Scafs-8x9square-240	ACCTTTTTTTAGACATTACACACGTAATTTTTTTTTGACGCACGTTTCCT
2Scafs-8x9square-241	AACAGACAGCGGCTTTAACCGTAAACGCGAACAATTGATAGA

2Scafs-8x9square-242	AATCGTGATGAACATAATAAGTTTGAATGTTGACGGCTGAAA
2Scafs-8x9square-243	AATATCCCTCAACGCAGCGACTACCATAAACGCAAGCAGAAC
2Scafs-8x9square-244	TGTAGCCGGCAGAAGCCTGAACGGTTAAATCCAAAAAATACT
2Scafs-8x9square-245	AGAATCTGGTTGAACAGCATCTTTCTTACCTATTAGCTGAGA
2Scafs-8x9square-246	ATAGCACAAACAGGCAAAAAAAGACTCAAAGCGAACGCCTTT
2Scafs-8x9square-247	TAGTCGTTTTGAACCGAAGATTTAGGGTCGGTTTTCATCAAAAGCGCATT
2Scafs-8x9square-248	CAACAGAAACCAGCCGTTTGAGCTTGAGTAAAATATCAGCAC
2Scafs-8x9square-249	TTGGTATTTTAAATACTGACAACCTGATTAGTTTTCGGCGTTGACGTCGC
2Scafs-8x9square-250	ATCTGAATGCATTCAATATCTGGTTGAACGGCAGATGTATCC
2Scafs-8x9square-251	GTCGTATTTTACCCAGCTTGGTAAGTTTTT
2Scafs-8x9square-252	TTTTTGGATTAAGCACTCCGTTTTTGGACAGATTTGGTCA
2Scafs-8x9square-253	TTCTTTGATTTGTCATTGTGAGCATTTTCATCTGCGAGTCAT
2Scafs-8x9square-254	GTTGCGTTTTGCTCATTCTGATTCTGTTTT
2Scafs-8x9square-255	ACTCCTTTTGCTGATGAACTAACCGCTGATTCTGCGCAAGAG
2Scafs-8x9square-256	CGTTGGTGTAGTTTTATGGGCGCATCAGCA
2Scafs-8x9square-257	CATCTGCCAGTGAGCCATTTGGGAATTAGAGCCGTAACCGTG
2Scafs-8x9square-258	TATTGATTTTCGGAAATTATAATTATCACCGTTTTTCACCGACTTTTGAG
2Scafs-8x9square-259	GGGACGTTTTACGACAGTATCGGCCTTTT
2Scafs-8x9square-260	TTTTTCAGGAAGATCGCACTTTTTCCAGCCAGCTGTAAA
2Scafs-8x9square-261	GCTTCTGGTGCCAACCGATTGAGGGAGGGAAGTTCCGGCACC
2Scafs-8x9square-262	TCATTACAAGAAGTCCTTTACCCTGCAACGTACCTTAAGGTG
2Scafs-8x9square-263	GCAAGGCCGGAAATCTCCAAAAAAAAGGCTCCATTACCATTA
2Scafs-8x9square-264	TCACCATTTTATGAAACCATAATCAAGTTTGTTTTCCTTTAGCGTGAGAA
2Scafs-8x9square-265	GCGTTTTCATTTTCAACAGTTTCAGCGGAGTCAGACTGTAGC
2Scafs-8x9square-266	CAAGCAAATCAGTAGCGACAGCGATAGCAGCACCGTACTTAT
2Scafs-8x9square-267	TTTTCGTTTTGTCATAGCCCCAAAATCACCGTTTTGAACCAGAGCCGTCT
2Scafs-8x9square-268	CCGCCTCCCTCGTAACGATCTAAAGTTTTGTCACCACCGGAA
2Scafs-8x9square-269	GAAATGCCATCTTTTCATAATCCTTATTAGCGTTTGCAGCAG
2Scafs-8x9square-270	CCGCCATTTTCCCTCAGAACCGCCACCAGAATTTTCCACCACCAGCACCA
2Scafs-8x9square-271	GCATTGACAGTACCGTAACACTGAGTTTCGTAGCCGCCGCCA
2Scafs-8x9square-272	GATTCTCACCACCCTCAGAGCCGCCACCCTCAGAGCCAAATC
2Scafs-8x9square-273	TGAGGCTTTTAGGTCAGACGCCTCATTAAAGTTTTCCAGAATGGACCTCA
2Scafs-8x9square-274	CTGAATTTACCGCCACCCTCAGAACCGCCACAAGCGCAGTCT
2Scafs-8x9square-275	TCTCTTCACAAACAAATAAATATTGGCCTTGATATTTTTGAG
2Scafs-8x9square-276	CAGTAATTTTGCGTCATACAAATAAGTTTTATTTTACGGGGTCAGCCGGA
2Scafs-8x9square-277	ACAGTGCCCGGAGGGTTGATATAAGTATAGCTGCCTTGAGTA
2Scafs-8x9square-278	AAGCAACAGGAGTGTACTGGTTGGCTTTTGATGATAGCATCT
2Scafs-8x9square-279	ACAGTTTTTTAATGCCCCCTTGAAAGTATTATTTTAGAGGCTGAGAAGGA
2Scafs-8x9square-280	AAACCTTATTATTCTGAAACAGCCTATTTCGGAACCGCTGTT
2Scafs-8x9square-281	TGGCGCTTTTATAATCTCGGGCTTGGAAAGATTTTTTGGTGTTTTGAGGG

2Scafs-8x9square-282	AGGCAGTCGGCCATAATAGACGCAACGCGAGATAAACATCAT
2Scafs-8x9square-283	GAAGGATTTTCGTCAATAGTATCATGGCGACTTTTCATTCAAAGGCAGTA
2Scafs-8x9square-284	GACTCCTTTTTTCTGTTGATCATTTTGTGCATTTTTATACCTGGTACGGG
2Scafs-8x9square-285	TGCCCGGCGTCTTTCGTATTCTGGCGTGAAGATAAACGTTAT
2Scafs-8x9square-286	CCTGATTTTTTCAGCGAAACAGACCAAACCATTTTTGAAACCAACTCGCC
2Scafs-8x9square-287	GACTGATTTTATGCCAGCAATCTCATTTTGCTTTTATCTCGGCAAAATAA
2Scafs-8x9square-288	TAATCTCTTTTCTCTTTCTGATTGTCCAGTTCTGGAGACAAA
2Scafs-8x9square-289	CCACCATTTTGCAAGAGCAGTAAATCACCTCTTTTACTTAAGTGGGCATT
2Scafs-8x9square-290	TTAGTATTTTAGCTCTTTTTCGGCGTCAACCTTTTATACCAGCAGTGGCG
2Scafs-8x9square-291	AATTTAGACAAGGAAGCATCAGCACCAGCACCGCCTCCAAAC
2Scafs-8x9square-292	ATGCCTTTTTACAGTATTGTCACCGGAGGCGTTTTGCTTTTTGACGCTCC
2Scafs-8x9square-293	CAAGCATTTTTTAAGCTCAGCAAGATAATCATTTTCGAGTATCCTCACCC
2Scafs-8x9square-294	ATACCAGCATTTCCTTTATCAGCGGCAGACTTGGCAGATTTA
2Scafs-8x9square-295	TTAGCCTTTTATAGCACCAGTAGGAACATTATTTTGAGCCTTGAATGCCA
2Scafs-8x9square-296	CCAAGTTTTTCCAACCAAATCAGAAACGGCATTTTGAAGTGCCAGCAGCT
2Scafs-8x9square-297	GGAATACGGCCTCATCAGGGTAAACAAAACTAGGGGAGTTTA
2Scafs-8x9square-298	TACGCACATCACCTTGAATGCTATCGGTAGCAAGCAGTATGT
2Scafs-8x9square-299	GTTACCCAATAGCACCAAACAAAGCAATACCGCCAGAGAAGG
2Scafs-8x9square-300	GCAAGAAGTAGCGGTAAAGTTCAATCCGCGGCATTTAACAAT
2Scafs-8x9square-301	ACAAAGGGTATAATAACCACCCACACAGTCCTTGACTCAGAG

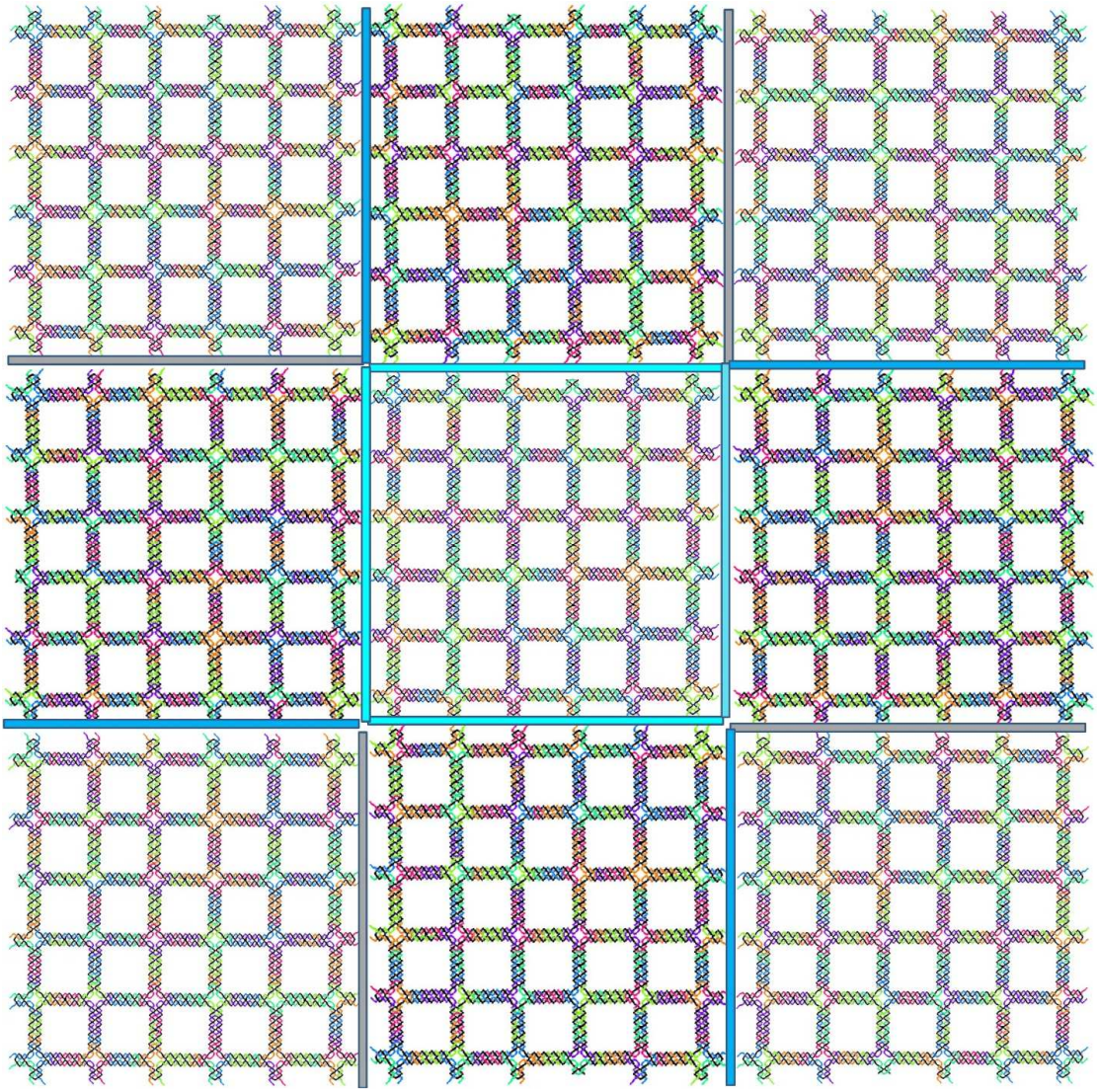


Fig. S70. Design pattern for the 3 × 3 square origami array. The black strand is the scaffold DNA, and other colorful strands are the staple strands. The white-washed panels indicate backward-facing origami.

Table S6. Sequences of the 3 × 3 square origami array. Three types of square origami are modified from the square Platonic tiling (fig. S67 and Table S3). The 43 staple strands that have their 5' or 3' ends exposed on the termini of the outward arms were deleted from the original design, and 43 new helper strands with sticky ends (5 nt) overhangs were added. The 43 strands that are exchanged are numbered 37-40, 54, 55, 65, 69, 70, 73, 74, 77, 78, 93, 98, 110, 111, 114, 119, 188-211 on the original design.

Number	Sequence
CenterSquare-1	CGGGGTAGAGCTTAATTGCTGTTTTAATATAATGCAAGTTACCAGATTTTAGGAA
CenterSquare-2	TCGCAATTTTATGGTCAATATAAAGATTCAATTTTAAGGGTGAGACAGAG
CenterSquare-3	AGCCCAAGTACGGTGTTTTTCTGGAAGTTTCATTCGCATA
CenterSquare-4	ACGTCCATATAACAGTTGATTTTTTCCCAATTCTGCCACA
CenterSquare-5	AGTGTAGGTCGACTCTTTTTAGAGGATCCCCGGGTGCATA
CenterSquare-6	ACGTCACCGAGCTCGAATTCGTTTTTAATCATGGTGGAAA
CenterSquare-7	CGGGGGGTTTTCCCAGTCACGTTTTACGTTGTAAAGCCGCGCTTAATTTTTGCGC
CenterSquare-8	TCCGAGGCGGTTTGCGTTTTTATTGGGCGCCAGGGCAGGT
CenterSquare-9	CTTAATGGTTTTTCTTTTCACTTTTCAGTGAGACGCTGAG
CenterSquare-10	AAGAAATTTTAATCTACGTTACAGGTAGAAATTTTGATTCATCAGTTGAGATAGT
CenterSquare-11	TGGTGCAGTCGGGAAACCTGTTTTTCGTGCCAGCTTAGCC
CenterSquare-12	AACCCGCTCACAATTCTTTTCACACAACATACGAGCTTAA
CenterSquare-13	AACGTTTGCGTTGCGCTTTTTCACTGCCCGCTTTCAATAA
CenterSquare-14	GAAGGCTTTTTTATCCGGTAATCAGAAAAGCTTTTCCCAAAAACAGGAAGATAGT
CenterSquare-15	TGGTGATTAATGCCGGAGAGGTTTTGTAGCTATTTAGAGC
CenterSquare-16	ATCCTGCCTGAGAGTCTTTTTGGAGCAAACAAGAGCAGGT
CenterSquare-17	CTTAAAATCGATGAACGGTAATTTTTCGTAAAACTCCCGA
CenterSquare-18	TTTGTTGATATTCAACTTTTCGTTCTAGCTGATAAAATAA
CenterSquare-19	TCGGTCCGGAAGCATAAAGTGTTTTTAAAGCCTGGGGCCC
CenterSquare-20	TCCTCATTGTATAAGCAAATATTTTTTTAAATTGTAAACGGTACT
CenterSquare-21	CGGGGTTAATATTTTGTTAAATTTTATTCGCATTAATATA
CenterSquare-22	GAACGCTTTTCATCAAAAATAATTCGCATA
CenterSquare-23	ACGTCGCGTCTGGCCTTCCTGTTTTTAGCCAGCTTACAAA
CenterSquare-24	GGATTCTTTTTCCGTGGGAACAAACCTTAA
CenterSquare-25	TCGGTGGCGGATTGACCGTAATTTTTGGGATAGGTGCGTA
CenterSquare-26	CTGCCATTTTGTTTGAGGGGACGACAATAA
CenterSquare-27	TGGTGGACAGTATCGGCCTCATTTTGGAAGATCGCTTTTA
CenterSquare-28	GAAACCTTTTAGGCAAAGCGCCATTCAGGT
CenterSquare-29	CTTAACGCCATTCAGGCTGCGTTTTCAACTGTTGGACGTA
CenterSquare-30	CGCTATACGCCAGCTGTTTTGCGAAAGGGGGATGTATAGT
CenterSquare-31	TCCTCGCTGCAAGGCGATTAATTTTGTTGGGTAACGCCAGGTACT
CenterSquare-32	TCCTCTTTTGATAAGAGGTCATTTTTTTTTGCGGATGGCTGTACT

CenterSquare-33	ACCGAGATTAGAGAGTTTTTACCTTTAATTGCTCCATAGT
CenterSquare-34	CTTAAGCGTTTTAATTCGAGCTTTTTTCAAAGCGAAACTG
CenterSquare-35	GCCCGATTTTAAGACTTCAAATATCCAGGT
CenterSquare-36	TGGTGGTCTTTACCCTGACTATTTTTTATAGTCAGAAGTT
CenterSquare-37	TGACCATTTTTAAATCAAAAATCAGAATAA
CenterSquare-38	TCGGTGAATCGTCATAAATATTTTTTCATTGAATCTAGCA
CenterSquare-39	TGGATATTTTGCGTCCAATACTGCGCTTAA
CenterSquare-40	ACGTCAATAGCGAGAGGCTTTTTTTTGCAAAAGAAACGAT
CenterSquare-41	CCAGACTTTTGACGATAAAAACCAAGCATA
CenterSquare-42	CGGGGTACATAACGCCAAAAGTTTTGAATTACGAGTTGGG
CenterSquare-43	TCCTCATTTAGGAATACCACATTTTTTCAACTAATGCAGAGTACT
CornerSquare-1	TTTTTAGAGCTTAATTGCTGTTTTAATATAATGCAAGTTACCAGATTTTAGGAA
CornerSquare-2	TCGCAATTTTATGGTCAATATAAAGATTCAATTTTAAGGGTGAGACAGAG
CornerSquare-3	AGCCCAAGTACGGTGTTTTTCTGGAAGTTTCATTCTTTT
CornerSquare-4	TTTTCATATAACAGTTGATTTTTTCCCAATTCTGCCACA
CornerSquare-5	AGTGTAGGTCGACTCTTTTTAGAGGATCCCCGGGTTGTTA
CornerSquare-6	CGTGCACCGAGCTCGAATTCGTTTTTAATCATGGTGGAAA
CornerSquare-7	GAAATGGTTTTCCCAGTCACGTTTTACGTTGTAAAGCCGCGCTTAATTTTTGCGC
CornerSquare-8	TCCGAGGCGGTTTGCGTTTTTATTGGGCGCCAGGGTTCAC
CornerSquare-9	TCACATGGTTTTTCTTTTCACTTTTCAGTGAGACGCTGAG
CornerSquare-10	AAGAAATTTTAATCTACGTTACAGGTAGAAATTTTGATTCATCAGTTGAGTATAA
CornerSquare-11	CTGAACAGTCGGGAAACCTGTTTTTCGTGCCAGCTTAGCC
CornerSquare-12	AACCCGCTCACAATTCTTTTCACACAACATACGAGTATAT
CornerSquare-13	AACGTTTGCGTTGCGCTTTTTCACTGCCCGCTTTCTGTTC
CornerSquare-14	GAAGGCTTTTTTATCCGGTAATCAGAAAAGCTTTTCCCAAAAACAGGAAGTTTT
CornerSquare-15	TTTTATTAATGCCGGAGAGGTTTTGTAGCTATTTAGAGC
CornerSquare-16	ATCCTGCCTGAGAGTCTTTTTGGAGCAAACAAGAGTTTT
CornerSquare-17	TTTTAATCGATGAACGGTAATTTTTCGTAAAACTCCCGA
CornerSquare-18	TTTGTTGATATTCAACTTTTCGTTCTAGCTGATAATTTT
CornerSquare-19	TTCTTCCGGAAGCATAAAGTGTTTTTAAAGCCTGGGGCCC
CornerSquare-20	TTTTATTGTATAAGCAAATATTTTTTTAAATTGTAAACGTTTT
CornerSquare-21	TTTTTTAATATTTTGTTAAATTTTATTCGCATTAATATA
CornerSquare-22	GAACGCTTTTCATCAAAAATAATTCTTTT
CornerSquare-23	TTTTGCGTCTGGCCTTCCTGTTTTTAGCCAGCTTACAAA
CornerSquare-24	GGATTCTTTTTCCGTGGGAACAAACTTTT
CornerSquare-25	TTTTGGCGGATTGACCGTAATTTTTGGGATAGGTGCGTA
CornerSquare-26	CTGCCATTTTGTTTGAGGGGACGACTTTT
CornerSquare-27	TTTTGACAGTATCGGCCTCATTTTGGAAGATCGCTTTTA
CornerSquare-28	GAAACCTTTTAGGCAAAGCGCCATTTTTT
CornerSquare-29	TTTTCGCCATTCAGGCTGCGTTTTCAACTGTTGGACGTA

CornerSquare-30	CGCTATACGCCAGCTGTTTTGCGAAAGGGGGATGTTTTT
CornerSquare-31	TTTTGCTGCAAGGCGATTAATTTTGTTGGGTAACGCCAGGTTGT
CornerSquare-32	TCAACTTTTGATAAGAGGTCATTTTTTTTTGCGGATGGCTTTTT
CornerSquare-33	ACCGAGATTAGAGAGTTTTTACCTTTAATTGCTCCAACTT
CornerSquare-34	AAGGAGCGTTTTAATTCGAGCTTTTTTCAAAGCGAAACTG
CornerSquare-35	GCCCGATTTTAAGACTTCAAATATCGTCAT
CornerSquare-36	GATAAGTCTTTACCCTGACTATTTTTTATAGTCAGAAGTT
CornerSquare-37	TGACCATTTTTAAATCAAAAATCAGGGTGA
CornerSquare-38	TTCGTGAATCGTCATAAATATTTTTTCATTGAATCTAGCA
CornerSquare-39	TGGATATTTTGCGTCCAATACTGCGAGACT
CornerSquare-40	ATCGCAATAGCGAGAGGCTTTTTTTTGCAAAAGAAACGAT
CornerSquare-41	CCAGACTTTTGACGATAAAAACCAATACAG
CornerSquare-42	CTGTCTACATAACGCCAAAAGTTTTGAATTACGAGTTGGG
CornerSquare-43	GACAAATTTAGGAATACCACATTTTTTCAACTAATGCAGAATAAT
EdgeSquare-4	TTTTCATATAACAGTTGATTTTTTCCCAATTCTGCCACA
EdgeSquare-5	AGTGTAGGTCGACTCTTTTTAGAGGATCCCCGGGTTATGC
EdgeSquare-6	GACGTACCGAGCTCGAATTCGTTTTTAATCATGGTGGAAA
EdgeSquare-7	CCCCGGGTTTTCCCAGTCACGTTTTACGTTGTAAAGCCGCGCTTAATTTTTGCGC
EdgeSquare-8	TCCGAGGCGGTTTGCGTTTTTATTGGGCGCCAGGGACCTG
EdgeSquare-9	TTAAGTGGTTTTTCTTTTCACTTTTCAGTGAGACGCTGAG
EdgeSquare-10	AAGAAATTTTAATCTACGTTACAGGTAGAAATTTTGATTCATCAGTTGAGACTAT
EdgeSquare-11	CACCACAGTCGGGAAACCTGTTTTTCGTGCCAGCTTAGCC
EdgeSquare-12	AACCCGCTCACAATTCTTTTCACACAACATACGAGTTAAG
EdgeSquare-13	AACGTTTGCGTTGCGCTTTTTCACTGCCCGCTTTCTTATT
EdgeSquare-14	GAAGGCTTTTTTATCCGGTAATCAGAAAAGCTTTTCCCAAAAACAGGAAGTTTT
EdgeSquare-15	TTTTATTAATGCCGGAGAGGTTTTGTAGCTATTTAGAGC
EdgeSquare-16	ATCCTGCCTGAGAGTCTTTTTGGAGCAAACAAGAGTTTT
EdgeSquare-17	TTTTAATCGATGAACGGTAATTTTTCGTAAAACTCCCGA
EdgeSquare-18	TTTGTTGATATTCAACTTTTCGTTCTAGCTGATAATTTT
EdgeSquare-19	ACCGACCGGAAGCATAAAGTGTTTTTAAAGCCTGGGGCCC
EdgeSquare-20	TTTTATTGTATAAGCAAATATTTTTTTAAATTGTAAACGACAAC
EdgeSquare-21	ATTTCTTAATATTTTGTTAAATTTTATTCGCATTAATATA
EdgeSquare-22	GAACGCTTTTCATCAAAAATAATTCTAACA
EdgeSquare-23	GCACGGCGTCTGGCCTTCCTGTTTTTAGCCAGCTTACAAA
EdgeSquare-24	GGATTCTTTTTCCGTGGGAACAAACATATA
EdgeSquare-25	AAGAAGGCGGATTGACCGTAATTTTTGGGATAGGTGCGTA
EdgeSquare-26	CTGCCATTTTGTTTGAGGGGACGACGAACA
EdgeSquare-27	TTCAGGACAGTATCGGCCTCATTTTGGAAGATCGCTTTTA
EdgeSquare-28	GAAACCTTTTAGGCAAAGCGCCATTGTGAA
EdgeSquare-29	TGTGACGCCATTCAGGCTGCGTTTTCAACTGTTGGACGTA

EdgeSquare-30	CGCTATACGCCAGCTGTTTTGCGAAAGGGGGATGTTTATA
EdgeSquare-31	TTGTCGCTGCAAGGCGATTAATTTTGTTGGGTAACGCCAGAGTAC
EdgeSquare-32	GTTGATTTTGATAAGAGGTCATTTTTTTTTGCGGATGGCTTTTT
EdgeSquare-33	ACCGAGATTAGAGAGTTTTTACCTTTAATTGCTCCAAGTT
EdgeSquare-34	TCCTTGCGTTTTAATTCGAGCTTTTTTCAAAGCGAAACTG
EdgeSquare-35	GCCCGATTTTAAGACTTCAAATATCATGAC
EdgeSquare-36	TTATCGTCTTTACCCTGACTATTTTTTATAGTCAGAAGTT
EdgeSquare-37	TGACCATTTTTAAATCAAAAATCAGTCACC
EdgeSquare-38	ACGAAGAATCGTCATAAATATTTTTTCATTGAATCTAGCA
EdgeSquare-39	TGGATATTTTGCGTCCAATACTGCGAGTCT
EdgeSquare-40	GCGATAATAGCGAGAGGCTTTTTTTTGCAAAAGAAACGAT
EdgeSquare-41	CCAGACTTTTGACGATAAAAACCAACTGTA
EdgeSquare-42	GACAGTACATAACGCCAAAAGTTTTGAATTACGAGTTGGG
EdgeSquare-43	GAGGAATTTAGGAATACCACATTTTTTCAACTAATGCAGAATTAT

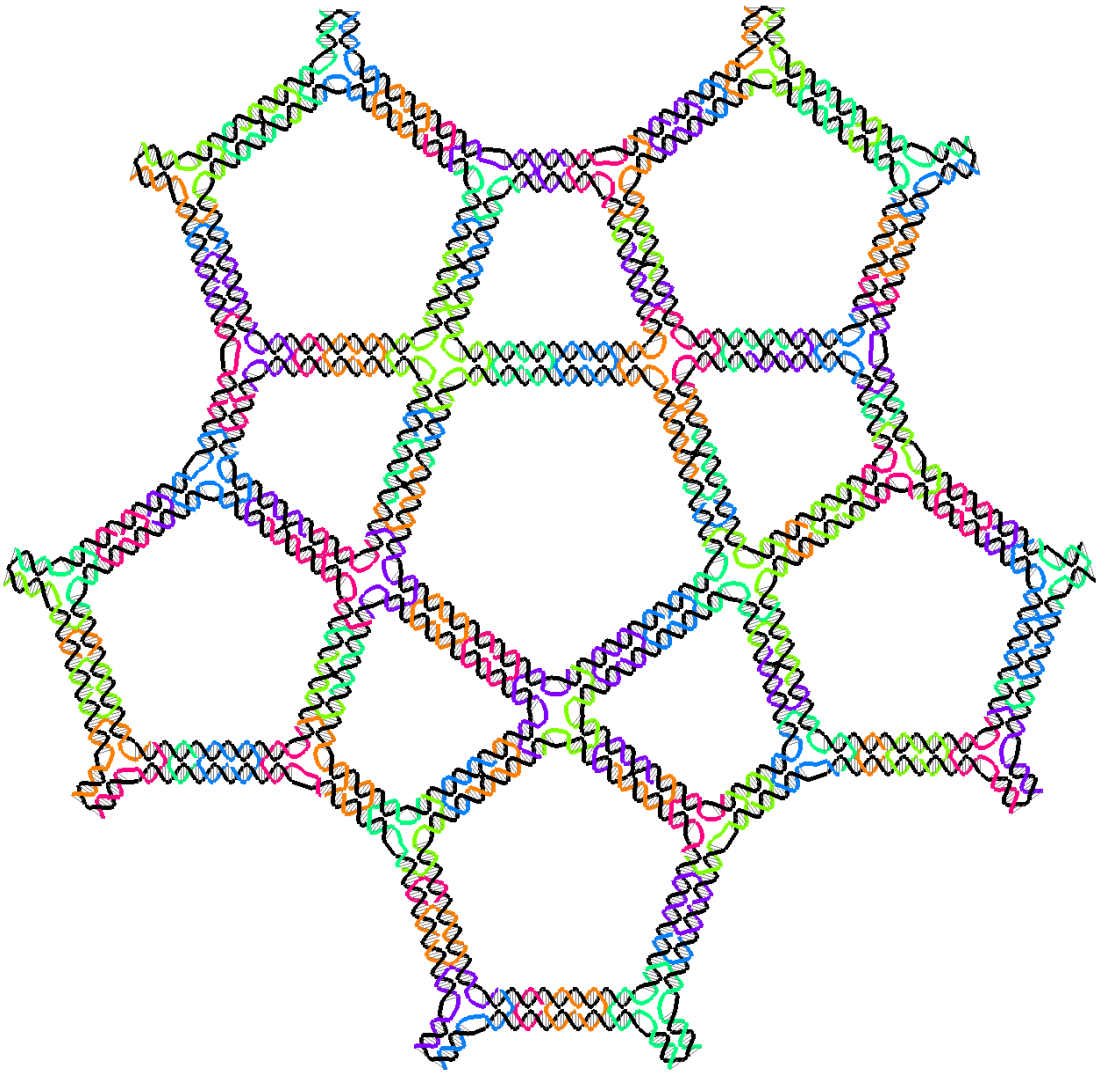


Fig. S71. Design pattern for the star shape

Table S7. Sequences of the star shape

Number	Sequence
1--scf-5star--	TAAAGGTTTTCCGCTTTTGCAAAAAAAGGCTTTTTCCAAAAGGAGCGAGG
2--scf-5star--	TAGAAATTTTGGAACAACTAGGTAGCAACGGTTTTCTACAGAGGCGTTTC
3--scf-5star--	AATAATAATTGAAAGACAGCATCGGAACGAGAAGGAATTGCG
4--scf-5star--	CCCTCAGCAGCTTTTCACGTTGAAAATCTCCAGGGATCGTCA
5--scf-5star--	ACCGAGTTTTGAAACGCAATACAAAGTCAGATTTTGGGTAATTGATGAGT
6--scf-5star--	TAAGCCTTTTCAATAATAAGTACCGAAGCCCTTTTTTTTTAAGAAAGGAA
7--scf-5star--	ACAAGAATGCGCTAATATCAGAGAACAACTTT
8--scf-5star--	CATTAATTTTACGGGTAAAATCTTTCCAGACTTTTGTTAGTAAATGAGAA
9--scf-5star--	CAACAGTTTATGGGATTTTGCTAAGATAACCC

10--scf-5star--	TCAGCGGAGTGAATTTTCTG
11--scf-5star--	GAGAATTTTTAACATAAAAATATTACGCAGTTTTTATGTTAGCAACATAT
12--scf-5star--	ATTAGACGGGACTGGCATGATTAAGACTCCTCAGGGAAGCGC
13--scf-5star--	TACCCAAAAGAAGAATTAACTGAACACCCTGAAATAACGGAA
14--scf-5star--	AAAAGATTTTAACGCAAAGACAATAGAAAATTTTTTCATATGGTTACAGA
15--scf-5star--	CCAATCTTTTCAAATAAGAAGGAGGGAAGGTTTTTAAATATTGACCGTCA
16--scf-5star--	TTTAACGTCAGGCGACATTCAACCGATTGAGACGATTTTTTG
17--scf-5star--	AAAGACAAAAGAAAATGAAAATAGCAGCCTTTTACCAGCGCC
18--scf-5star--	TGAATTTTTTTCTTAAACAGAACAACCATCGTTTTCCCACGCATAGGAGT
19--scf-5star--	CTTGCTTTCCTTTAATTGTATCGGAGTTACAA
20--scf-5star--	CCGACTTTTTTGAGCCATTTTGAATCTTACCTTTTAACGCTAACGTTATC
21--scf-5star--	AATAAACACAGAGCCTAATTTGCCTTTATCAG
22--scf-5star--	GCCATATTATAGCGTCTTTC
23--scf-5star--	TCCACATTTGACAGCCCTCAAACGAAAGAGTTTGCAAA
24--scf-5star--	CACCAACCTAATAGTTAGCG
25--scf-5star--	AGTTTTGTCGTACGTAATGCCACTACGAAGGTAACGATCTAA
26--scf-5star--	AAAAGGTACTTTTTCATGAGGAATTTGAGGA
27--scf-5star--	AATAAGTTTAGAATATAAAACGACGACAATTTTAAACA
28--scf-5star--	CTAAAGAAAGTAATTCTGTCCAGGTACCGAC
29--scf-5star--	TAAGTCCTCGCTGAGGCTTGCAGACCGATAT
30--scf-5star--	AATTTATTTATGGTTTGAAGCAGAACGCGCTTTCTGTT
31--scf-5star--	TATCAATCCCATCCTATTTATTTACGAGCACCTA
32--scf-5star--	ATTCGGTGAACAAGAAAAATAATACAATAGA
33--scf-5star--	GTATTCTATGCGCCGACAATGACCTTGATAC
34--scf-5star--	AATCAGTTTATATAGAAGGTAGCGAACCTCTTTCCGAC
35--scf-5star--	CGATAGTAGAACGCGAGGCGTTTCTTATCCG
36--scf-5star--	GGGTTTTTTTGCTCAGTACTATACCAGTCATTTGGACG
37--scf-5star--	TTGGGGAAACATGAAATTTGTATTAAGAGTAGCG
38--scf-5star--	AGATAGCCGAGAAGGATTAGGATGCTGAGAC
39--scf-5star--	TCCTCAAGAACAAAGTTACCAGAAAGTAAGC
40--scf-5star--	TCAGGGTTTATAGCAAGCCTTTCGTCACCATTTGTACA
41--scf-5star--	ACAATGAATACCGTAACACTGAGCAATAGGA
42--scf-5star--	ACCCATGATAGCAATAGCTATCTAGCAAGAA
43--scf-5star--	ACAGTGCCACATAAAGGTGGCAAACGTAGAA
44--scf-5star--	ACTAAAAGTTTTAACGTTTGGGTCAGTGCGCCCC
45--scf-5star--	CTGCCTTTTATTTCGGAACTACGTTAATAATTTAACGA
46--scf-5star--	AATACATCGTATAAACAGTTAATCTTGAGTA
47--scf-5star--	AACCACCATTATTTTGTCACAATCACCACGG
48--scf-5star--	CACCACTTTCCTCAGAGCCCATTGACAGGATTTGGTTG
49--scf-5star--	AATAAGTCCAGAGCCGCCGCCAGGCCACCAG

50--scf-5star--	AGCCACCAAAGGTGAATTATCACGGAAATTA
51--scf-5star--	TCATACTTTTCATAATTTTCAAAATCACCCAGAG
52--scf-5star--	CCGCCATTTCCCTCAGAACCAATACTGCGGTTTAATCG
53--scf-5star--	TTCATTACCGGAACCGCCTCCCTGGAACCAG
54--scf-5star--	TTAAATTTTCAAGATTAGTACCATTACCATTTTTAGCA
55--scf-5star--	AATTTTATCCGGGAATTAGA
56--scf-5star--	CACCAGTAGCTGCTATTTTGCACCCAGCTACGCCAGCAAAAT
57--scf-5star--	AATCCTTTTCATTAAAGCCACTATCATAACTTTCCTCGTTTACCAGAC
58--scf-5star--	TTCAACTAATGCAGATTTTACATAACGCCTTCCA
59--scf-5star--	GTAAGCTTTGTCATACATGATCAGTTGAGATTTTTTAGGAATACCACA
60--scf-5star--	GAATTTACCGAAAAGGAATT
61--scf-5star--	TAAGAGCAACAGAATGGAAAGCGCAGTCTCTACGAGGCATAG
62--scf-5star--	AGGCATAATAGTAAAATTTTGTTTAGACTAGAGC
63--scf-5star--	GACGATAAAAACCAAATTTATAGCGAGAGAAATA
64--scf-5star--	ATTCACAAACGCTTTTGCAA
65--scf-5star--	CCAGAGGGGGGGTCAGACGATTGGCCTTGATAAGAAGTTTTG
66--scf-5star--	TACTGGTAATCGGAACAACA
67--scf-5star--	AGAAAGATTCGCTTTTGATGATACAGGAGTGTTATTACAGGT
68--scf-5star--	AGGCCCTTCTGTAAATTTTCGTCGCTATTAAGCC
69--scf-5star--	AAAGATTTTTAAGAGGAAGCTTTTTTAATGTTTGAAAC
70--scf-5star--	AGTACGCACCGTAATCTTTAGTAGCGACAATCAA
71--scf-5star--	TAAATCAAAAATCAGGTTTTCTTTACCCTGCGCG
72--scf-5star--	TTTTCATTTTCGGCATTTTAACAGTTCAGATTTAAACGAGAATGACCA
73--scf-5star--	CGTTTGCCATAATATTCATT
74--scf-5star--	AAATGCTTTACGGTCATAGCCCCCTTATTAGGAATCCCCCTC
75--scf-5star--	TCAGACTGTAGACTATTATA
76--scf-5star--	AGCGGATTGCGAATCAAGTTTGCCTTTAGCGGTCAGAAGCAA
77--scf-5star--	ATCGATAGCAATAAATCAAT
78--scf-5star--	AATAACCTTGGGAAACGTCACCAATGAAACCATATGTGAGTG
79--scf-5star--	AAGACAAAGAACGCGATTTGAAAACTTTTTCCTT
80--scf-5star--	ATCATTTTTCCAAGAACGGGTAAATGCTGATTTTGCAAATCCAATCGC
81--scf-5star--	TAGGTCTGAGAGACTATTTCCTTTTTAACCAAGCAAGCCGTTTTTTTT
82--scf-5star--	ATTTTAGTCAATAGTGTTTAATTTATCAAAATCA
83--scf-5star--	TTGCGCCCTTAGAATCTTTCTTGAAAACAAAGCA
84--scf-5star--	GCCCAATAGCTAGCGATAGC
85--scf-5star--	CGCTGAGAAGCATCGTAGGAATCATTACCGCTTAGATTAAGA
86--scf-5star--	TCATCGAGAACTCCGGCTTA
87--scf-5star--	ATAACTATATGTATTAAACCAAGTACCGCACGGTTGGGTTAT
88--scf-5star--	CGGCTGTCTTTCAAATATAT
89--scf-5star--	TCATCTTCTGATGTAGAAACCAATCAATAATTTTAGTTAATT

90--scf-5star--	AGAATCCAACTTTGAATTTAGAGGACAGAAGCAT
91--scf-5star--	CCATGTTACTTAGCCGTTTGAACGAGGCGATTAT
92--scf-5star--	ACCAAGTTTCGCGAAACAAAATTGTGTCGATTTAATCCGCGACCTGCT
93--scf-5star--	TTATACAAATTCTTACTTTCAGTATAAAGATCGC
94--scf-5star--	CATATTTTTTAACAACGCCAAAAAGCCTGTTTTTTAGTATCATATGCG
95--scf-5star--	ACATGTGTGATAAATATTTAGGCGTTAAACCAGT
96--scf-5star--	CATTTTCGAGTAAGAATAAA
97--scf-5star--	TAATTACTAGAACATGTAATTTAGGCAGAGGCACCGGAATCA
98--scf-5star--	AATTGAGACCAACGCTCAACAGTGTATCATC
99--scf-5star--	GCCTGATAAGTACAACGGAGATTTAGGGCTT
100--scf-5star--	ACCCCCAGCGCAGACGGTCA
101--scf-5star--	ACCGAACTGAACACTAAAACACTCATCTTTGATCATAAGGGA
102--scf-5star--	TTAATTTCAACTTTAATTTTCATTGTGAAATAAG
103--scf-5star--	TATAGCTTTCCGGAATAGGGAGTAGTAAATTTTTGGGCTTGAGATGGT
104--scf-5star--	CGTAACAAAGCTGCTCTTTATTCAGTGAACCACC
105--scf-5star--	CTCAGATTTACCGCCACCCAACCGGATATTTTTCATTACCCAAATCAA
106--scf-5star--	AACTAACAGACCAGGCTTTGCATAGGCTGCATTT
107--scf-5star--	CCACCACCCTGCTGACCTTC
108--scf-5star--	TCTTGACAAGTCAGAACCGCCACCCTCAGAGATCAAGAGTAA
109--scf-5star--	TTTAGTACCGTAAGGCTTGC
110--scf-5star--	ACACCAGAACTGTATCACCGTACTCAGGAGGCCTGACGAGAA
111--scf-5star--	GAGGGTTGATTTACCTTATG
112--scf-5star--	ACTGGCTCATCAGGCGGATAAGTGCCGTCGACGATTTTAAGA
113--scf-5star--	ATACCGACCGTTCAGCTAAT
114--scf-5star--	TGAACGGTGTCAACGCCTGT
115--scf-5star--	AAGAAAAATCCTATTATTCT
116--scf-5star--	GGATAGCGTCCGCCACCCTC
117--scf-5star--	AATTAATTTTGGAGGTTTTG

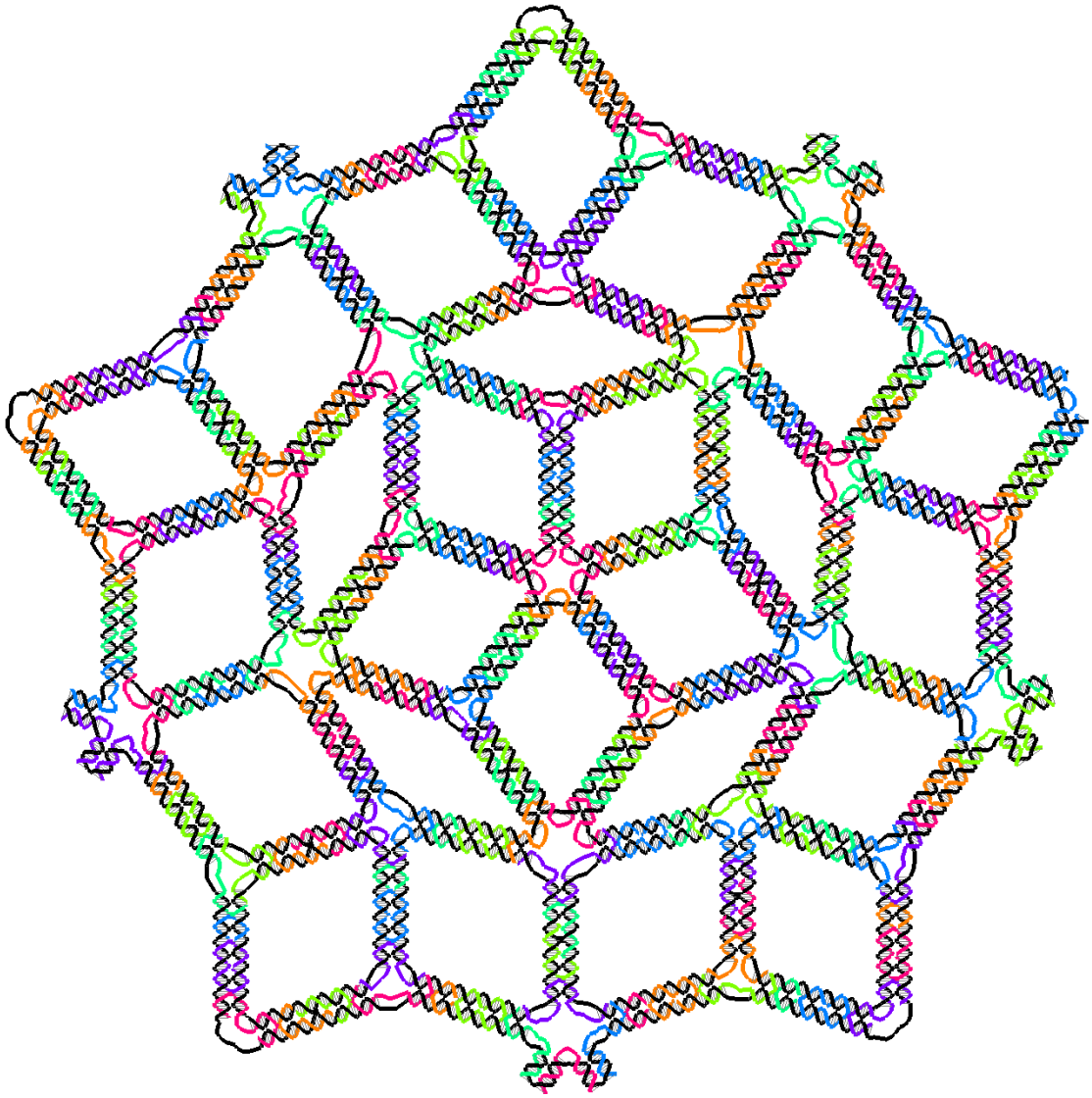


Fig. S72. Design pattern for the 2D Penrose tiling

Table S8. Sequences of the 2D Penrose tiling

Number	Sequence
Penrose-1	GGACGACGACTGGAAATACCTTTTTACATTTTGACACCGTGCATCT
Penrose-2	AAAACGCTCAAGTATCGGCCTCAGGAAGATCTTGCAACAGGA
Penrose-3	GCACTCCAGCCCGCCAGCCA
Penrose-4	TATTACAGCTTTCCGGTTTTACCGCTTCTGTAGTA
Penrose-5	CCAGGCAAAGCTTGTAGCAATACTTCTTTGATGTGCCGGAAA
Penrose-6	GCCATTCGCCAAATTAACCG
Penrose-7	ATAACATTTTCACTTGCCTGTAATATCCAGTTTAACAA
Penrose-8	CCTTGCTGGAGTAGAAGAACTCAATAAGAAC
Penrose-9	GCGAGGCGGCTTATCCGGTATTCACTATCGG
Penrose-10	GCAAGCTTTTCGTTTTTATTCAAGCAAATCATTTTGATATAGAAGTTTTA
Penrose-11	GCGAACTTTTCTCCCGACTTGGATTAGGATTTTTTTTTGGTTTTGCTCAACAA
Penrose-12	CTCAAGAGAAGCGGGAGGTT
Penrose-13	AATCAAGATTAGTATTAAGAGGCTGAGACTCTTGAAGCCTTA
Penrose-14	TATTCTTTTTTGAAACATGAAAGTTGCTATTTTTTTTGCACCCAGCTACGAG
Penrose-15	TTTATAATATCTTACCAACGCTAACAATTTT
Penrose-16	ATCCTGACAGTGAGGCCACCGAGGAAGTGTT
Penrose-17	CGTCTTTTTTTCCAGAGCCTATATTTATCCCATTTTTATCCA
Penrose-18	AGCCATATATTTGCCAGTTACAACGGAACGA
Penrose-19	GGCGCAGACTCCATGTTACTTAGCAATAAAC
Penrose-20	AATAATTGAGTAACAGTTTTTGCCCGTATACCTAT
Penrose-21	GGTCAGTGCCGAAACGATTT
Penrose-22	TCAAAAATGAACTGGTAATAAGTTTTAACGGTTTGTTTAACG
Penrose-23	AGTGTAAATAGCAGCCTTTTTTACAGAGAGCGGGA
Penrose-24	AAACACTCGGGAAGCGCATTAGAAATAACAT
Penrose-25	AAAAACAATCTTTGACCCCCAGCATACACTA
Penrose-26	GAATTATTTACTGAACACCCTTTTGATGATTTTACAGG
Penrose-27	TCATACATGGCTGAACAAAGTCAGAGGGTAATCCAGTAAGCG
Penrose-28	TTGAGCGCTAAATTTACCGT
Penrose-29	GAAAGCTTTTGCAGTCTCTGATATCAGAGAGTTTTATAAC
Penrose-30	CCACATGTTAGCAAACTTTTGTAGAAAATACACCG
Penrose-31	TTACGCAGTAAGAATTGAGT
Penrose-32	ATAAGAGCAATGGCATGATTAAGACTCCTTATAAGCCCAATA
Penrose-33	GAATACTTTCCAAAAGAACGAAACAATGAATTTATAGC
Penrose-34	AATAGTAAGCAGATAGTTTTCGAACAAAGTTAACG
Penrose-35	AATGCCACCCTTTTTAAGAAAAGCTATCTTA
Penrose-36	CCGAAGCTACGAAGGCACCAACCAAATACGT
Penrose-37	ATTCGGTCGAGGAAACGCAATAATACCAGAA

Penrose-38	GGAAACCGCTGAGGCTTGCAGGGACCGATAT
Penrose-39	TTATCACCGTCATACATAAA
Penrose-40	ATAAAAGAAAAAATTATTCATTAAAGGTGAAGGTGGCAACAT
Penrose-41	AGGTAATTTATATTGACGGCGCAAAGACACTTTCACGG
Penrose-42	AATAAATATGGTTTACTTTTAGCGCCAAAGAGGGA
Penrose-43	ACCGATAGATCAATAGAAAATTCGTTTATTT
Penrose-44	TGTCACATTGCGCCGACAATGACAGCTTGAT
Penrose-45	TCAACAGTTCAACCGATTGAGGGACAAAAGG
Penrose-46	GCGACATTTCAGCGGAGTGAGAAAACAACTT
Penrose-47	ACTTGATTTTGCCATTTGGGTTTCATAATCATTTTAAATCACCGGGAATG
Penrose-48	TTTGCCATCTAATTAGAGCC
Penrose-49	CAGTAGCACCGTCATAGCCCCCTTATTAGCGAGCAAAATCAC
Penrose-50	TTCATCTTTGGCATTTTCGATTACCATTAGTTTCAAGG
Penrose-51	CCGGAACCGTAATCAGTTTTAGCGACAGAAGCGTT
Penrose-52	CTTTCCAGACCATCGATAGCAGCAACGTCAC
Penrose-53	CAATGAAACGTTAGTAAATGAATTTTGTCGT
Penrose-54	AACCGCCAGCGTCAGACTGTAGCTCAAGTTT
Penrose-55	GCCTTTACCCTCAGAGCCACCACACCCTCAG
Penrose-56	ATTAAAGCCAAACCAGAGCC
Penrose-57	CGCCTCCCTCTTCACAAACAAATAAATCCTCACCACCGGAAC
Penrose-58	ACGATTTTTGGCCTTGATAAGAGCCGCCACTTTCCTCA
Penrose-59	GAACCGCCACCAGAACTTTTACCACCAGAGGTCAG
Penrose-60	ATCACCGTACCACCCTCAGAGCCGCCACCCT
Penrose-61	CAGAGCCACTCAGGAGGTTTAGTATAGGTGT
Penrose-62	ATGCCCCCAGGAGGTTGAGGCAGCCGCCGCC
Penrose-63	AGCATTGACTGCCTATTTCGGAAAACAGTTA
Penrose-64	GGATAAGTGCCCCAAGTACCGCACTCATCGAGAGTACCAGGC
Penrose-65	GTCGAGAGGGGGGTATTAAA
Penrose-66	CCCATCTTTTTTAATTTACGACTTTCCTTATCTTTTTTTCCAAGAACTTGAT
Penrose-67	ATAAGTTTTTTATAGCCCGGAACCGCCACCCTTTTTCAGAA
Penrose-68	CCGCCCCTCATTTTCATTTTTGGGATAGCAAAATAT
Penrose-69	CAAGAAAAATGCCCAATAGGAACCCATGTACTAAGTCCTGAA
Penrose-70	CGTAACACTGCAACAATAGA
Penrose-71	GAACGCTTTTGCCTGTTTATAGTTTCGTCACTTTTTTTCAAACTACAATTTAG
Penrose-72	CAACATGTAACGCCTGTAGC
Penrose-73	AGCCCTCATAAATCGCCATATTTAACAACGCATTCCACAGAC
Penrose-74	TAGTATTTTTTATATGCGTTATCAACAGTAGGTTTTTCTTAATTGAGGTTAG
Penrose-75	CGTAACTTTTTGATCTAAAGTTTTCTGTATGGTTTTGATTT
Penrose-76	TGCTATAGAAAGGAACTTTTTAACTAAAGGACTGTT
Penrose-77	TAGAAAAAGCATTGCGAATAATAATTTTTTCATCATAATTAC

Penrose-78	ACGTTGAAAAAAACACCGGA
Penrose-79	GGCGTTTTTTAAATAAGAATTCTCCAAAAAATTTTTTTTCCAAAAGGATCCAA
Penrose-80	CTGATGCAAAGCCTTTAATT
Penrose-81	TCAGCTTGCTTTATATAACTATATGTAAATGGTATCGGTTTA
Penrose-82	AGTCAATTTTTAGTGAATTTATTAACCTCCGGTTTTTTTAGGTTGGGTTCGA
Penrose-83	GGTGAATTTTTTTTCTTAAACAACAACCATCGTTTTCCCAC
Penrose-84	GCATAAGTTAAAGGCCTTTTTGCTTTTGCGGAGAAG
Penrose-85	TAAGACGCTGGATCGTCACCCTCAGCAGCGAATAGCTTAGAT
Penrose-86	AAGACAGCATAAACATAGCG
Penrose-87	CCCTTATTTTGAATCCTTGACGGAACGAGGGTTTTTTTACGGCTACAGTTTAA
Penrose-88	CTTATGCGATAGGCTTTGAG
Penrose-89	TTTTCATGAGACTTTAATCATTGTGAATTACGACTAAAGACT
Penrose-90	GAGATGTTTTTTTTAATTTCAGAAGTTTCCATTTTTTTAAACGGGTATAAAA
Penrose-91	CGAAAGTTTTAGGCAAAAGAGATTATACCAATTTTTGCGCG
Penrose-92	AAACATCAGTGAATAATTTTTGCTTGCCCTGGGCTT
Penrose-93	AGCTGCTCATAAGTACAACGGAGATTTGTATCAACGTAACAA
Penrose-94	CATCGCCTGATTACCCAAAT
Penrose-95	AAGAACTTTTCGGATATTCATAAATTGTGTCTTTTTTTCCGCGACCTGCGGTC
Penrose-96	AATCATTTTTAAGGGAACCGACGGTGTACAGTTTTACCAGGCGCATTGAC
Penrose-97	ACAGATGAAACTGACCAACTTTGAACGTGCT
Penrose-98	TTCCTCGTTTTGACGAGCACGTATAAAGAGG
Penrose-99	CGCGTATTTCTATGGTTGCTAGAATCAGAGTTTCGGGA
Penrose-100	GCTAACCCAGTCACGATTTTGTTGTAAAACCAGGG
Penrose-101	CCAGGGTTTTACAGGAGGCC
Penrose-102	GTTGGGTAACGGATTAAAGGGATTTTAGACAGAGGCGATTAA
Penrose-103	CAGCTGGCGAAAGGGGTTTTTTATGTGCTGCAGAACG
Penrose-104	GTACGCTTTTTTAGAATCCTGATAAAAGAGTCTTTTTTTTCCAT
Penrose-105	CACGCATTCAGGCTGCTTTTTTCAACTGTTGGGAAGG
Penrose-106	GCCAAGCTTGCGCCGCGCTTAATGCGCCGCTAGACGGCCAGT
Penrose-107	ATGCCTGCAGCACCACACCC
Penrose-108	TACCGAGCTCAGCGGTCACGCTTTTTGCGCGTAACGTCGACTCTAG
Penrose-109	TGGCAAGTGTGAATTCGTAA
Penrose-110	GCTGTTTCCTAAGGAGCGGGCGCTAGGGCGCTCATGGTCATA
Penrose-111	GAGTAATCTAGGCTGGCTGACCTCCGGCGAA
Penrose-112	CGTGGCGAAGCTTGACGGGGAAAGTCATCAA
Penrose-113	GGAGCCTTTCCCGATTTAGGAAAGGAAGGGTTTAAGAA
Penrose-114	AGCGAGTGTGAAATTGTTTTTATCCGCTCATAAAG
Penrose-115	ATCGGAACCCCAATTCCACA
Penrose-116	TAAAGCACTAACAACATACGAGCCGGAAGCATCGAGGTGCCG
Penrose-117	GGGGTAAAGTGTAAAGTTTTTTCTGGGGTGCCTAATG

Penrose-118	TCACTGCCCGCTTTCCTTTTTTGTCGGGAAACGTCAA
Penrose-119	AGGGCGTTTTTTAAAACCGTCTCCCAAATCAAGTTTTTTTTTTT
Penrose-120	GGACTCCAACCTGTCGTGCC
Penrose-121	TTAAAGAACGTAGCTGCATTAATGAATCGGCCGAGTCCACTA
Penrose-122	CACCAGAACTACGTGAACCATCAATCAGGGC
Penrose-123	GATGGCCCACGAGTAGTAAATTGACGAGAAA
Penrose-124	GAACTGTTTTGCTCATTATAGTTAATAAAACTTTTGAACTAACGGATTTT
Penrose-125	AAATCTACCCAGTCAGGACGTTGCCGAGATA
Penrose-126	GGGTTGAGAAATCAAAAGAATAGCGGAAGAA
Penrose-127	CGGCAATTTAATCCCTTATTGTTGTTCCAGTTTTTTGG
Penrose-128	AACAAAACGCGCGGGGTTTTGAGGCGGTTTGAAAT
Penrose-129	TGGTGGTTCCGCGTATTGGG
Penrose-130	ATCCTGTTTGACGCCAGGGTGGTTTTTCTTTTGCAGGCGAAA
Penrose-131	GATTGCCCTTGGTCCACGCTGTTTTGTTTGCCCCACACCAGTGAGA
Penrose-132	CAGCAAGCCACCGCCTGGCCCTGAGATACAT
Penrose-133	AACGCCAACATTCAACTAATGCAGAGAGTTG
Penrose-134	CTATTAATTAAACAACATTA
Penrose-135	AAGATTCATCCCTTGCTTCTGTAAATCGTCGTTACAGGTAGA
Penrose-136	ATATGTTTTGAGTGAATAAAGTTGAGATTTTTTAGGAA
Penrose-137	TACCAAAGGAATTACGTTTTGGCATAGTAATCAAT
Penrose-138	AGTACATAAAGAGCAACACT
Penrose-139	TTAATGGAAACATCATAACCCTCGTTTACCAGATTACCTTTT
Penrose-140	TTTGAACGACGATAAATTTTTTACCAAAATAGCGAGA
Penrose-141	GTAATAGTAAAATGTTTTTTTTAGACTGGATAAATTA
Penrose-142	CCTGAGTTTTTTAAAAGAAGATTACATTTAACATTTTTTTTTCA
Penrose-143	TATTCATTTCGCGTCCAATA
Penrose-144	TCATAAATATCAAAATCGCGCAGAGGCGAATCTGCGGAATCG
Penrose-145	ATAGGTCAGAAAACAAAATTAATGATGAAAC
Penrose-146	AAACATCATGAGAGACTACCTTTTCAAAATC
Penrose-147	TCGCAATTTTGACAAAGAACTAATTTCATCTTTTTTCTGACCTAAAATAA
Penrose-148	ATTTTAGTGCGAGAAAACTTTTTATAACGGA
Penrose-149	TTCGCCTGTACATCGGGAGAAACACAAATAT
Penrose-150	CAGTAATTTCAGTACCTTTATTGCTTTGAATTTTACCA
Penrose-151	AGTTATCATTGAATCCTTTTCCTCAAATGCATATA
Penrose-152	CGTCAGATGATTTAAACAGT
Penrose-153	TTCAGGTTTAATCAGAAAACGAGAATGACCATTGCGTAGATT
Penrose-154	CTGACTATTAGTAAAACAGAATTTTATAAAGAAATAAATCAAAAAT
Penrose-155	CAAAGCGGATTCCATATCAAAATTATTTGCACTAGTCAGAAG
Penrose-156	GCATCAAAAATTAGAACCTA
Penrose-157	GTGTGATAATTTAATGGTTTGAAATGATTGT

Penrose-158	TTGGATTATCATCAATATAATCCTACCGACC
Penrose-159	ATCAGATTTTGATGGCAATTACTTCTGAATTTTAATGG
Penrose-160	AAGGGGATTAAGAGGATTTTGCCCGAAAGATGATT
Penrose-161	ATCATATTCCCTTCAAATAT
Penrose-162	GCGGAATTATCCGCGTTTTAATTCGAGCTTCAACCAGAAGGA
Penrose-163	AAACCAAGCGAACCAGTTTTTTCCGGAAGCAAACTCC
Penrose-164	GCTCCTTTTGATAAGATTTTTTGTCATTTTTGGTATT
Penrose-165	AAATCCTTTTTTTTGCCCGAACCATTTTGCGGATTTTTTCAAAG
Penrose-166	AGAGCTTAATTTTACAAACAATTCGACAACTCCGGATGGCTT
Penrose-167	GCTGAATATAGTATTAGACT
Penrose-168	CTTACCAGTTTGAGTAACATTATGTTATTAA
Penrose-169	TTTTAAAAGTATAAAGCCAACGCTACAAATT
Penrose-170	GCAGAGTTTTGCATTTTCGAAAGGTAAAGTATTTTATTCTGTCCAATGCA
Penrose-171	ACCGACAAGCCAGTAATAAGAGAAGAGCCGT
Penrose-172	CAATAGATACAACTAATAGATTATATAAAGT
Penrose-173	ATCTTTTTTAGGAGCACTAAATACATTTGATTTGGATT
Penrose-174	TAGAAATGCTGTAGCTTTTTAACATGTTTTAAAAT
Penrose-175	AAGGTTATCTAAATATGCAACTAAAGTACGGAGGAATTGAGG
Penrose-176	TGTCTGGAAGAACAGTTGAA
Penrose-177	CAAAAACAGGTATCTGGTCAGTTTTTTGGCAAATCTTTCATTCCAT
Penrose-178	AAGCAAATATTAAATATCAAACCCTCAATCAAAAGATTGTAT
Penrose-179	TAAATTGTAAGCTGAACCTC
Penrose-180	TTCAGCTAGACGACGACAATAAAGAGCCAGC
Penrose-181	AGCAAATGAACAGTGCCACGCTGACAACATG
Penrose-182	TATTAATTTCACCGCCTGCAAAAATCTAAATTTGCATC
Penrose-183	ACCTTACGTTAATATTTTTTTGTTAAAATTGTCAG
Penrose-184	AGGTGAGGCGCGCATTAAAT
Penrose-185	AGATAAAACAGTTTTGTTAAATCAGCTCATTTCACCAGCAGA
Penrose-186	CGAACTTTAACCAATATTTTTTGAACGCCATCAAAAA
Penrose-187	TTCATCAACATTAAATTTTTTTTGAGCGAGTAATTTT
Penrose-188	TGAATGTTTTTTCTATTAGTCTCCATTAAAAATTTTTTTCCGAA
Penrose-189	CACAGACAATACAACCCGTC
Penrose-190	GGGAACAAACTGAAAGCGTAAGAATACGTGGGGATTCTCCGT
Penrose-191	TGACCGGCGGATTGACTTTTGTAATGGGATGGCAG
Penrose-192	TATTTACATTAGGTCACGTTGGTGTAGATGGCTGAAATGGAT
Penrose-193	GCGCATCGTAGCTCAATCGT
Penrose-194	ATTCACTTTCAGTCACACGAGAGATAGAACTTTCCTTC
Penrose-195	TGGCCAACACCAGTAATAAAAGGTTACCGCG
Penrose-196	CCCAATAGTTCATCGTAGGAATCAGACATTC
Penrose-197	AAACCAATAGCCCTAAAACATCGTTAATGCG

Penrose-198	CGAACTGATCAATAATCGGCTGTGCATGTAG
Penrose-199	AACAGGTCAGGATTAGTTTTTAGAGTACCTTTAATT
Penrose-200	TAATTCGCGTCTGGCCTTTTTTTCCTGTAGCCAGCT
Penrose-201	GCGATCGGTGCGGGCCTTTTTTCTTCGCTATTACGC
Penrose-202	AGTGAGCTAACTCACATTTTTTTAATTGCGTTGCGC
Penrose-203	GGCTTTTGCAAAAGAATTTTTGTTTTGCCAGAGGGG

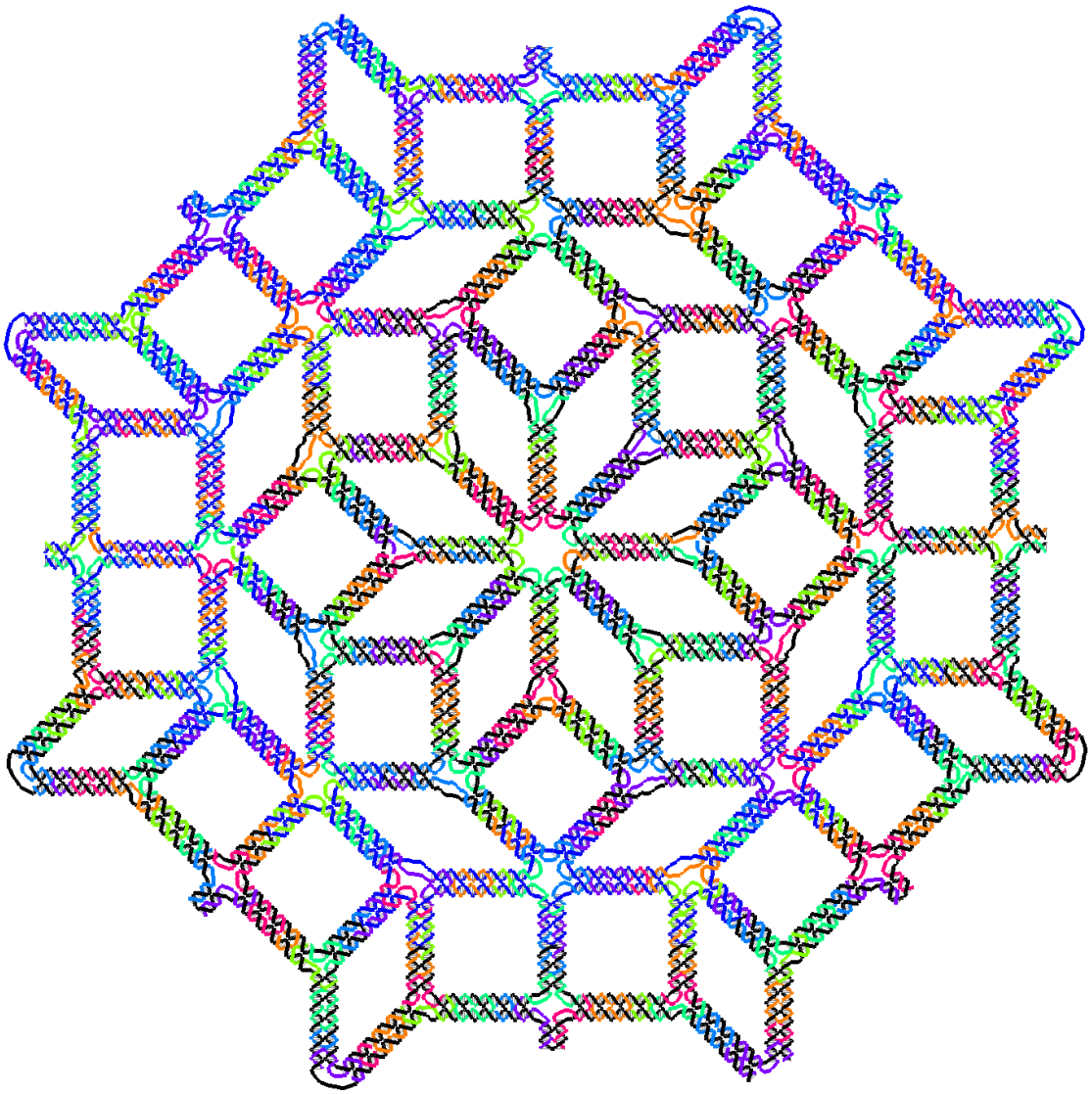


Fig. S73. Design pattern for the 2D quasicrystalline pattern with 8-fold symmetry

Table S9. Sequences of the 8-fold quasicrystal 2D pattern

Number	Sequence
1--QC8fold--	AAGAAGTTTTTATGATGAAACACAATTTCATTTTTTTTGAATTACCTTGTGA
2--QC8fold--	AGAGCAAAAATTAATTACATTTAAAACATCA
3--QC8fold--	AAGAATTTTTGAGTTAAGCCCAATAGCTATCTTTTTACCG
4--QC8fold--	AGAAAACAGAAACAATGAAATAGCAATAATA
5--QC8fold--	GTACCTTTTTTTACATCGGGCTGACCTAAATTTTTTAATG
6--QC8fold--	AGAACGTTTTTCGAGAAAACTGAATACCAAGTTTTTTTACAAAATCGAGCAA
7--QC8fold--	TGATTGCTTTTTTTCAAATATATTTTAGTTAACGGATTCGCC
8--QC8fold--	ATTTCATCTTAGAAACAATA
9--QC8fold--	GTGAATTTTTTAACCTTGCTTTAGAATCCTTGTTTTTAAAACATAGCATAGT
10--QC8fold--	AAAAATGTACATAAATCAATATATTTTTAAT
11--QC8fold--	AAGAAATTTTGATTTTTTGTACAGAGAGAATTTTTACATA
12--QC8fold--	GGAAACAGAAAATAGCAGCCTTTTTAACGTC
13--QC8fold--	GAATTTTTTTTATCAAAATCACTTAGGTTGGGTTTTTTTATATAACTGACAA
14--QC8fold--	ACTCATCACGCTGAGAAGAGTCAGATAGCTT
15--QC8fold--	GAACGGTTTTTATTAAACCAGTTTTTATTTTTTTTATCGT
16--QC8fold--	AGATTAAGGAGAACAAGCAAGCCAGTACCGC
17--QC8fold--	ATCAAGATTAATTAATTTTCCCTCTGTAAAT
18--QC8fold--	CTTGCGTTTTGAGGTTTTGACACCCAGCTACTTTTATTTT
19--QC8fold--	CGTCGCTATTAGTTGCTATTTTGAGCCTTAA
20--QC8fold--	AAGCCAAGCAAATCCAATCGCAAATATGTAA
21--QC8fold--	CGTTATTTTTCAAATTCTTACTTAATTGAGATTTTTCGCC
22--QC8fold--	ATGCTGATCGCTCAACAGTAGGGCCAGTATA
23--QC8fold--	TGTTCAGCCTTTTTAACCTCCGGTAGGTCTG
24--QC8fold--	GTCCAGTTTTCGACGACAATCCTGTTTATCATTTTCAATA
25--QC8fold--	AGAGACTACTAATGCAGAACGCGAAACAACA
26--QC8fold--	CAGTATGTCATTTCAATTACCTGCGCAGAGG
27--QC8fold--	GGCATGTTTTTTAAGACTCCAATACATACATTTTTAAGGT
28--QC8fold--	CGAATTATTTAGCAAACGTAGAATTATTACG
29--QC8fold--	CCACTATTTTTTAAAGAACGAAAGTTACCAGTTTTTAAGGAAACCGTTTGC
30--QC8fold--	CCCAGCTTTTAGGCGAAAATTCCCTTATAAATTTTTTTTTAATAGCCCGAAGAGT
31--QC8fold--	GGCAATTGCCCTTCACTTTTCGCCTGGCCCGAACT
32--QC8fold--	ATACCCAAAATGAGAGAGTTGCAGCAAGCGGATAATAACGGA
33--QC8fold--	TCCACGCTGGAGGAAACGCA
34--QC8fold--	AAGCCCTATCAGGGCGTTTTATGGCCCACTCCCAC
35--QC8fold--	GAAAAACCGTCTTTTTAAGAAAAGTAAGCAGCGTCAAAGGGC
36--QC8fold--	ATAGCCGAACTGGACTCCAA
37--QC8fold--	TAAGCAGTCCGAAATCGGCAAAACCTGTTTG
38--QC8fold--	AAGCAGTTTTTATCAGTGACGCACGTTCTTGGTTTTTTCAGT

39--QC8fold--	ATGCATGTCAATAGATTTTTGTGGTAGAAGGAGGA
40--QC8fold--	ATGGTGGTCTTGCAGACCCATAAAATTAGCA
41--QC8fold--	TCAGTTCGTTCCAGTTTGGAACAGATAGGGT
42--QC8fold--	ACGCGGACCAGAAAACTTTTTGGCCTAACGCAGAT
43--QC8fold--	GCCCAGTTTTTAGATTAGAGCTGTTGCTTGGATTTTTAGATTGGTGTTCCTT
44--QC8fold--	TGAGTGTTCATCAACATCATAGCACGTTTGG
45--QC8fold--	CCGTTTTTTTGAGCTTGAGTATAAGAGGTTTTTTTTTTTTAAATGAAGAATATGA
46--QC8fold--	GAAGAGTTTTCCATACCGCTTCAGCACTAACTTTTTCTTGCGAGTCACCAG
47--QC8fold--	ATGAAGTAATACATTAGAAATATCCTTTGCAGGTAACGCTGC
48--QC8fold--	GTAGCGCCAAATAACATCAT
49--QC8fold--	TGTCAGCGTCAAGCATTTGGCGCATAATCTCTAAAGGACGGT
50--QC8fold--	GGAAACCTGCGCATGACAAG
51--QC8fold--	GATTTGGTTTTCATAGTGGAGGCAAAGCCTC
52--QC8fold--	TCAAACTTTTGCTGAATAGCCTCCAGCAATCTTTTGAA
53--QC8fold--	TACGCGACATTGGTAAAATACTGATTTCTTT
54--QC8fold--	GTTTGCTGTTTCGTCCCCTTCGGTCTCTAAA
55--QC8fold--	GTAACCTTGTCTTCTCGTGGCGGTGGTCTTTTTATAGT
56--QC8fold--	AACCATTATGAACTAAGTCAACCGATTCTGC
57--QC8fold--	GTTATTGCGATGGGCATTTTCTGTAACCATACTGC
58--QC8fold--	AATTAGGGTCTACTCATCGCGTTTTTAATATCCTTAGCGGCGATTGC
59--QC8fold--	TTGCGGCAAAAAGGCCACGTATTTTGCAAGCCAGCAGCCAGC
60--QC8fold--	TATTTAACTGAGAGGGCGTT
61--QC8fold--	CACTCTGCACCGCATGTTTTAATGAAGACGATTCA
62--QC8fold--	CCGCATAAAGATCCTTAATACCTTTCTTTTTGTAGGAAGTGT
63--QC8fold--	GGGGTAATTAAACGCTACCT
64--QC8fold--	TAGCATCAACAGGCCATTTTTAACCAACCAGTTGTG
65--QC8fold--	CCAATTTTTTTATCCATTAACACGTCAGAAGCTTTTTGCCTT
66--QC8fold--	CATTGTAGCAAACGTGAAAAAGCGTCCTGCGTTGGGGGAGCA
67--QC8fold--	TGTAGCGAACTAATATCAAG
68--QC8fold--	AAGCATTTTTTGGGATTATCAGTGAGAGTGTCTTTTTAAAAC
69--QC8fold--	GATAAAAATACTGATATTTTTCAGTCGGCGTGTGAA
70--QC8fold--	GTCGCATTGCGCCATTAGCTGTACCATACTCAGCATGAGCCT
71--QC8fold--	AGGCACACAAACCAACCATC
72--QC8fold--	CTGTTGTTTTTTAAGCAAGCAGCTCAGGAACATTTTTAAGAA
73--QC8fold--	CTCTAATAACGCGAGCAGTAGACTTTCCATA
74--QC8fold--	ATAGACGCCGGTCGTCAGCCAACTAAAACGC
75--QC8fold--	TCATTAGCCTTGCGACTTTTTCTCGGCAGCAACGCA
76--QC8fold--	CGGTGTTTTTTCAGACCAGGCATCTTGACAAGTTTTTACCGG
77--QC8fold--	ATGCAGCACCTTCATCAAGAGTAGCATAGGC
78--QC8fold--	AATCTAAACATCCTTCTTTTATAGAAATTTGGTCG

79--QC8fold--	AAAGGAAGTGCCATACAAAACAGCACGCGGC
80--QC8fold--	AAAACACACCCGCCGCTTTTGCTTAATGCGCAAAT
81--QC8fold--	GGCAAGTGGAAGAAAGCGAAAGGGTGGCGAG
82--QC8fold--	GAGGTGTTTTCCGTAAAGCATAGAGCTTGACTTTTTTTTTGCCGGCGAACAGCGG
83--QC8fold--	GCGCTATTTTGGGCGCTGGCAACACCCTGAATTTTTCAAAGTCAGAGGGTC
84--QC8fold--	GAATTAACTGAAGTGTAGCG
85--QC8fold--	CGTAACCACCAGGGAAGCGCATTAGACGGGAGTCACGCTGCG
86--QC8fold--	AAGTTTTTTGGGGTAATTGA
87--QC8fold--	AGAGAGATAAACGTGAACCATCACCCAAATCGCGCTAATATC
88--QC8fold--	TGTTTATAAGGGAGCCCCCGATTCTAAATCG
89--QC8fold--	GAACCCTAAGGTCTGTTGAACACGCACAGAA
90--QC8fold--	CCAGCATTTTTATATCGGTATCAGCACGCTCCTTTTTAAGCATTAAGTATCC
91--QC8fold--	TGGCTGAGCAAGATAATCACGAGCTCAGGAA
92--QC8fold--	TTTCCTTTTTTTATCAGCGGCACCACTGACCCTTTTTTCAGC
93--QC8fold--	GCATCAGCACAAGTCAAAGC
94--QC8fold--	TAAGGTACTGTCAACCATACCAGCAGAGGAAACCTTTAGCGT
95--QC8fold--	ATTCTCTTTTAAATCCGGCGAATCTCTTTAGTTTTTTTTTAGGCGGAAAAATTCT
96--QC8fold--	GGTCTTTCGTCGAACAAGCG
97--QC8fold--	ATAGTGCCATTCTCATTTTGTGCATATACCTCAAGAGTAAAC
98--QC8fold--	GGCGTGTTTTAAGTCGCCGATTTGCATCTCGTTTTTGCAATCTCTTTTTTG
99--QC8fold--	CATTATTTCTTTTTGAGTCTCATCTGAATGC
100--QC8fold--	ATAAATATTCCATTACCCAA
101--QC8fold--	CAGCAATCACAGGTAGAAAGATTCGGAACAA
102--QC8fold--	TAATAATTAACGAACTAACATCAGTTGAGTTTTATTTA
103--QC8fold--	GGAATCCAAAAGGAATTTTTCGAGGCATAGTACGT
104--QC8fold--	ACGACCAATGCAGATACATAACGACCACATT
105--QC8fold--	CAACTAATATCACGAAAATAGTCAGAACCAT
106--QC8fold--	AAGAAAAATCTAAGAGCAACACTATCATAACAGGACGTTGGG
107--QC8fold--	CCTCGTTTACTATACCAGTC
108--QC8fold--	GAGAGGCTTTCGATTTTAAGATTTTTACTGGCTCATCAGACGACGAT
109--QC8fold--	TTACCTTATGTGCAAAAGAA
110--QC8fold--	GGGGGTAATATTCAACTTTAATCATTGTGAAGTTTTGCCAGA
111--QC8fold--	TTAATGTAAAATGTTTTTTTACTGGATAGCATTCA
112--QC8fold--	AAAGCTGCTCGTCCAATACTGCGGAATCGTCATCAACGTAAC
113--QC8fold--	AGTTCAGAAAACGAGATTTTTTGACCATAAAATGAA
114--QC8fold--	ATATTATTGAATCCCCTTTTTTCAAATGCTTTAAAC
115--QC8fold--	GTGAATTTTTAAGGCTTGCCTTGGGCTTGAGTTATGGT
116--QC8fold--	TCCAGTTGAGAACGAGTAGTAAACTGACGAG
117--QC8fold--	AAACACCCATTTTAGTAAGCTCTTCTGATTG
118--QC8fold--	ATTTACTTTTATTGGCAGATATATAGAAGGCTTTTTTTATCCGGTAAACTC

119--QC8fold--	ATCCTGTTGTAGCAATTTTTACTTCTTTGACCCGA
120--QC8fold--	AGGAACATTCTGGCCATTTTACAGAGATAGTCCAA
121--QC8fold--	AAACTATTTTTCGGCCTTGCATTGCAACAGGTTTTTTTTTCTCATGGAAAGGATT
122--QC8fold--	ACGGTCAAATATTACCGCCAGCCTGGTAATA
123--QC8fold--	TAACATTTGAATTATGTTTTGCGAGAAATAAGAAA
124--QC8fold--	TTTTGACGATTAAACTCCTAAGCAAAGTCTG
125--QC8fold--	AAACATGCTCAATCGTCTGAAATTACCTACA
126--QC8fold--	ACCTACTTTTTCGCGCTTCGCCCGGCGTACGGTTTTTGAAGGACGTCCACCA
127--QC8fold--	TCCAGAACCATTAAATTTAACCTTGCGATAA
128--QC8fold--	CAGATGACTATTCCACTTTTTGCAACAACTGGTGG
129--QC8fold--	GCCAGAATACTGGTCATAATCATGAACGGAC
130--QC8fold--	TGGAAACCCTGAGAAGTGTTTTTAACGGTAC
131--QC8fold--	CGAATATTTTTAGTACGCGTTGCCTACAGTATTTTTTGTTATCGGTAGGCTT
132--QC8fold--	CATCACCCATCTTGCAAATC
133--QC8fold--	GTTCCTGAATGAATGGCAGATTTAATACCAGACCAGAAGGCG
134--QC8fold--	GGAACATTTTTTAGAGCCTTGAATGGGAAGCTTTTTTTTTAAGGTGATAAAACTT
135--QC8fold--	CAAATCAAGCGCAGGAGAAA
136--QC8fold--	GCATAACGATAGACTTGCCACCAAGTCCAACCATACGAAGGC
137--QC8fold--	AGAATCAGAATCACCAGAACGGATAAACTTC
138--QC8fold--	TTAGACGAGCGGGAGCTAAACAGTCCTCGTT
139--QC8fold--	ATCAGATTTTAACGGCAGAATTTACCAGCTTTTTTTTAGCCATAGCGGTTA
140--QC8fold--	TGTTACTTACCTTCAAGAAGTCCGTGCCAGC
141--QC8fold--	CTGTAGTTTTTATTCCACAGAGTTTTGTCGTCTTTTTTTCCA
142--QC8fold--	CACAGTCCGCGTAACGATCTAAACAGCCCTC
143--QC8fold--	ATAGTTATTGACGGTATAATAACAATAGTCA
144--QC8fold--	GACGTAAAATTTTTAGTTTTTACCCTCATATATTTT
145--QC8fold--	CTGCAACGTAGCCGGAACGAGGCCTGCTCCA
146--QC8fold--	TGTCGATTAATCCGCGACGCAGACGGTCATTTTATCAT
147--QC8fold--	AAGGGAGTCAGAAGCATTTTGCGGATTGCAAATTG
148--QC8fold--	TGACTATTATAACCGAACTG
149--QC8fold--	AAGAGGACAGTCAAAAATCAGGTCTTTACCCACCAACTTTGA
150--QC8fold--	TCGCCTGATATCAAAAAGATTAAGAGGAAGCGATTTGTATCA
151--QC8fold--	CCGAAAGACTGTACAACGGA
152--QC8fold--	TCAAAGCGAATTATACCAAGCTTTTTGCGAAACAAATCAAATATCGC
153--QC8fold--	CCCCCAGCGACCAGACCGGA
154--QC8fold--	ACAGGTCAGGCACTAAAACACTCATCTTTGAAGCAAACTCCA
155--QC8fold--	GAATAATTAGAGAGTATTTTTTTAATTGCTTTCCA
156--QC8fold--	ATGAGGAAGTCCTTTTGATAAGAGGTCATTTAAGACTTTTTC
157--QC8fold--	TTGCGGATGGTTGAGGACTA
158--QC8fold--	AGGCTCTTAGAGCTTATTTTTTTGCTGAATATAATG

159--QC8fold--	CTGTAGCTCAACATGTTTTTTTTAAATATGCCCGCT
160--QC8fold--	GAGTTAAAGGAACTAAAGTACGGTGTCTGGAAGGCTTGCAGG
161--QC8fold--	AGTTTCATTCTCGGTCGCTG
162--QC8fold--	TATATCATATAACAGTTTTTATTCCCAATTTTCTT
163--QC8fold--	CGAGGTGAATCTGCGAACGAGTAGATTTAGTCAGCTTGCTTT
164--QC8fold--	TTGACCATTAATCGGTTTAT
165--QC8fold--	AAAGCCTCAGCCAAAAGGAGCTTTTTCTTTAATTGTGATACATTTCG
166--QC8fold--	AAAAAAGGCTAGCATAAAGC
167--QC8fold--	TACCAAAAACTTTCACGTTGAAAATCTCCAATAAATCGGTTG
168--QC8fold--	AATTTATTATGACCCTTTTTAATACTTTTGTAAAC
169--QC8fold--	GGGATTTTGCCGGGAGAAGCCTTTATTTCAAATTTTCTGTAT
170--QC8fold--	CGCAAGGATATAGTAAATGA
171--QC8fold--	AAATGCAATGCCTGAGTTTTTAATGTGTAGGAACGC
172--QC8fold--	AACTTTTTTTCAACAGTTTCAGGAATTGCGATTATAAT
173--QC8fold--	TTCAGCGAGAAAGGAACAACTAAAGCGGAGT
174--QC8fold--	GAGAATAAACCAATCCGCGGCATTAACCTGA
175--QC8fold--	AAACAGTTCTTGATACCGGCCCACGCATATTTTACCGA
176--QC8fold--	ATAAATCAATGACAACAACCATCATAGTTGC
177--QC8fold--	GCCGACACCTCACTTAAGTGGCTCACCAAAC
178--QC8fold--	TTTGCGTTTTTGATCGTCACCGGTAGCAACGGTTTTTTACAG
179--QC8fold--	ATCACCTTCAGCATCGGAACGAGCTCAGCAG
180--QC8fold--	CGAAAGAGAATGCCACCGGAGGCGCAAGCAC
181--QC8fold--	TTAAACTTTTGGGTAAAATAAACGAAAGAGGTTCAAAA
182--QC8fold--	CAAAACTAAAGGCACCAACCTAACGTAATGC
183--QC8fold--	CACTACGGGGGCGGCCTCATCAGACCAGAAA
184--QC8fold--	TTTGACTTTTTGCCTCCAAACAATCGCGAGTGTTTTTGTCGG
185--QC8fold--	GTCATGATTGAATTTAGACATGGCGCCACCATTTATCACGAA
186--QC8fold--	GCAAGAGCAGCACTCAATCT
187--QC8fold--	AGCGGTTTTTAAAGTTAGACTTTTTACCGCTTTTTTTTTTTATAACCTCAAAGCA
188--QC8fold--	ATACCGTTTTCCAGCAATAGGGAGACAAATATTTTTATCTCTTTAATTAGT
189--QC8fold--	AAAATTAAAACAAACCATGAAACCAACATAACCTCAGCGGCA
190--QC8fold--	ACGTTATTGCTTGGTCAACC
191--QC8fold--	TCATGGTTTTTGACCATTCAAGCCGAAGAAGCTTTTTTGGAG
192--QC8fold--	AGCAAATCAGTCACCAGTCA
193--QC8fold--	ATAAAAGGGATCATTACCGCGCCCAATAGCACACGACCAGTA
194--QC8fold--	TGAGTAGAAGTTCTAAGAAC
195--QC8fold--	TAGCGAACCTTTAGTAATAACATCACTTGCCGCGAGGCGTTT
196--QC8fold--	CAAATTAACCGAATCTTACCAACGCTAACGATGTCCATCACG
197--QC8fold--	GCGTCTTTCCTAAAAGAGTC
198--QC8fold--	CAGTGATTTTGGCCACCGAGAGAGCCTAATTTTTTTTGCCAGTTACACGAG

199--QC8fold--	GGTTGCTTTGAAAATAAACA
200--QC8fold--	TATCCCAATCCCGCTACAGGGCGCGTACTATGCCATATTATT
201--QC8fold--	CACGTATTTTTAACGTGCTTGAGGCCGATTATTTTTTTTTTTTAGACAGGATAAT
202--QC8fold--	GTTTGAAATAAAGAAATTTTTTGCGTAGATTAACA
203--QC8fold--	ATATTATTAAATCCTTTTTTTGCCCGAACGATATG
204--QC8fold--	CAACAGCTGACATATAAAAGAAACGCAAAGAAGTGAGACGGG
205--QC8fold--	CACCACGGAATCTTTTCACC
206--QC8fold--	GCGCCATTTTGGGTGGTTTTTAAGTTTATTTTTTTTTGTCACAATCAGACA
207--QC8fold--	AAAGGGTTTTCGACATTCAAAACCTGTCGTGTTTTTTTTTCATTAATGAAATTGG
208--QC8fold--	TGGCGAGGAGAGGCGGTTTGCGTTCGGCCAA
209--QC8fold--	CGCGCGGGAAAGCTCAGTCTCAGTCGTCATT
210--QC8fold--	TAGTCTTTTTATCAGAAAAAAAGGAAGTGAG
211--QC8fold--	GATAAAGCAAATGAAATTTTAATCTAAAGCATTCT
212--QC8fold--	AACCAGCAATGCGCGAACTGATAATGGCTAT
213--QC8fold--	CAGACATTTTATATTTTTGAGCCCTAAAACATTTTTTTTTTAAAAATACCTGAGG
214--QC8fold--	AAGCGCAGAAGATAAAACAGAGGGAACGAAC
215--QC8fold--	CACCAGCAGTCTCTGAATTTACCAGAATGGA
216--QC8fold--	CGGTCATTTTGTATTAACACAATTTACGAGCTTTTTATGTAGAAACTGGCA
217--QC8fold--	ATCCCATCCTCGCCTGCAAC
218--QC8fold--	GAGAGCCAGCGTCCTGAACAAGAAAAATAATAGTGCCACGCT
219--QC8fold--	AGGTAAAGTAATCACCTTGCTGAACCTCAAAGTACCGACAAA
220--QC8fold--	TATCAAACCCAGAATATAAA
221--QC8fold--	TTGAGGTTTTATTTAGAAGTCATTTTCGAGCTTTTTCAGTAATAAGTCAAT
222--QC8fold--	CAATATTTTTCTGGTCAGTTGTTATCTAAAATTTTTTTTTAGGAGCACTATACAT
223--QC8fold--	ATTGGCCAAAGGAATTGAGGAAGGGCAAATC
224--QC8fold--	AACAGTTGTTGATATTCACAAACGTCAGACG
225--QC8fold--	CATAATCAGCCGTCAATAGATAAACAACTAA
226--QC8fold--	TAGATTAGAAAATCACCGGAACCCATCTTTT
227--QC8fold--	TAGGCAGAGGATTAGACTTT
228--QC8fold--	GACAACTCGTTAACAACGCCAACATGTAATTACAAACAATTC
229--QC8fold--	TAAGAATACGCAATCAATAA
230--QC8fold--	TCCTTATCATAACCCTTCTGACCTGAAAGCGTCGGCTGTCTT
231--QC8fold--	AGCGGATTTTTGCAGTCCAAAATTGTTCCAAGTTTTTATCGGCAACAACCAG
232--QC8fold--	AACAGATGAAAAATATCAACCACGCTTTATC
233--QC8fold--	AATACCATACAAACTCATCACGATTCTCAGT
234--QC8fold--	ATGGCAACTGGTAGCTTTTTTTAAGCGGCTCACCTT
235--QC8fold--	ACAGCCATATCGTCAACATA
236--QC8fold--	TATCGAACTCTATTTAGCCACATAGAAACCACATATCACCAT
237--QC8fold--	AAGCCCCATATCTGACTTTTTTTGTTAACGAACGC
238--QC8fold--	GCAGCAAGGTAAGCATTGGGGATTGAGAAAGCTAGACCTTTA

239--QC8fold--	AGTAGAAATGTCTTTAGCTC
240--QC8fold--	GTTGTTCCATCCACAAGCCTCTTTTTAATAGCAGGTCGATAAAACTC
241--QC8fold--	GTCATGGAAGTTAAGAGCCT
242--QC8fold--	TTAACTTCTGCCGATACGCTCAAAGTCAAAATTCCGAAAGTG
243--QC8fold--	CAGGGAATTTTTCGACTTTTATCAGAAATAAATCA
244--QC8fold--	TTATCAAGATTTAATCGTGC
245--QC8fold--	GCATGGTCAACAAGCAGTAGTAATTCCTGCTCAAGAAAAGCG
246--QC8fold--	GCAGTCCACTTCGATTTTTTTAATTCGTAAATATAA
247--QC8fold--	GTTCATAGGTCGAATTTTTTTTCTCATTTTCCGCCA
248--QC8fold--	TGCGCAAGGACCATAAACGT
249--QC8fold--	ACATAAAAAGCAAAGTAAGAGCTTCTCGAGCGACGATGAGGG
250--QC8fold--	GGCGTCGTTAGTTGATTTTTCGAAAGGTCGTAAAA
251--QC8fold--	TTGACAGAATCCATCTCGAAGGAGTCGCCAGCGCGTCAGTTT
252--QC8fold--	CGATAACCGGCCTCATCCAA
253--QC8fold--	AGCCACTTCTAGTAGTTGAAATTTTTTGGTAATAAGCCTTGACGAAC
254--QC8fold--	CTAAACCAGTACGACCAATC
255--QC8fold--	GAAGCCAAGATGAACAAAATGTGACTCATATTGACCAGCAAG
256--QC8fold--	GGGGCTGTCAACAAGATTTTATCTCTACCATGGGA
257--QC8fold--	TCTTTTAAAACGAAGCCCCT
258--QC8fold--	TGTTGACCACGACTCAGATAGTAATCCACGCGCAATTAAAAT
259--QC8fold--	TTACCTATTAGTGGTTTTTTTAACAGCATCGCTACA
260--QC8fold--	GAGCGTTGTTCAGTAATTTTTTTGACTCATGATTTC
261--QC8fold--	AACGTACGGACCAGAACGTT
262--QC8fold--	GACATTACATTAGAGGCCAAAGCGGTCTGGATTTTACCTTTA
263--QC8fold--	GCGCGGGCAGAAGCCTTTTTATGAGCTTAACACTC
264--QC8fold--	AATCCAAAACTACACGCAAGGTAAACGCGAAATCTTCGGTTA
265--QC8fold--	CAATTCAGCGAATCGAAATC
266--QC8fold--	CTCGTCAGAAGCTTTAACCGGTTTTTACGCTCGACGGGTCAGTAGCA
267--QC8fold--	GCACGAGAGCCCATTAATAA
268--QC8fold--	AATTCAGCGCCAAGCCTCAACGCAGCGACGATGTTTTCCGTA
269--QC8fold--	AAACTCAAAGTCCAGCTTTTACCATAAACGCTTCC
270--QC8fold--	GTAAACGTAGCGAGGGTATCCCACAACAGGA
271--QC8fold--	GCAGGAATAATATTTTGTTAAAATTTAAATT
272--QC8fold--	AAAACAGGAAGATTGTTTTTTTAAGCAAATATTCGC
273--QC8fold--	GAACCCCCCGGTTGATTTTTTATCAGAAAAGCCCCA
274--QC8fold--	ATCATATGTATATTATTCTG
275--QC8fold--	TATTAAGAGGAATCGTAAAACTAGCATGTCAAAACATGAAAG
276--QC8fold--	TCGAGAAACAAGAGAATTTTGATGAACGGTCTGAG
277--QC8fold--	AGTCTGGAGCAGGGTTGATATAAGTATAGCCCATTGCCTGAG
278--QC8fold--	CGGAATAGGTGCTATCAGGT

279--QC8fold--	ATCTACAAAGGTATCACCGTATTTTTCTCAGGAGGTCCGGAGAGGGT
280--QC8fold--	TAAATTAATGTTAGTACCGC
281--QC8fold--	CCGCCACCCTATATTCAACCGTTCTAGCTGACACCCTCAGAA
282--QC8fold--	ACCGTACAGTCAAATCTTTTCATCAATATGCAGAA
283--QC8fold--	AAGGCCGGAGAACACTGAGT
284--QC8fold--	TACAAACTACTAAAGATTCAAAAGGGTGAGATTCGTCACCAG
285--QC8fold--	CCGCCATTCCCTCAGAGCCAATAGGAACCTTTTCATGT
286--QC8fold--	CAAACAGGCAGGGATAGCAAGCCCACCACCC
287--QC8fold--	TCATTTTCAAAAAATTTAGGGTCAAGCGAAC
288--QC8fold--	ACTCCTTTTTCAAGAGAAGGAGGCGGATAAGTTTGCCG
289--QC8fold--	AAACAACCGTTTTGCTCAGTACCATTAGGAT
290--QC8fold--	TAGCGGGTGATTAGCGGCGTTGAACCAACAG
291--QC8fold--	ATTAAATTTTTTTTTGTTAAAGCCCCCTGCCTTTTTTTTTCG
292--QC8fold--	TTTTAACCAATGTGCCCGTATAAACAGTTAATTCAGCTCATT
293--QC8fold--	AGGAACGCCATTGAGTAACA
294--QC8fold--	ATGATGTTAGACAGGCCGCAATGACGGCATTTTGCAAT
295--QC8fold--	CCGTGGGAGATGAACATAATAAGTTTGAATG
296--QC8fold--	TTGACGGACAAACGGCGGATTGACGGATTCT
297--QC8fold--	CTTCCGTTTTCACGTAATTTAACGTCAGTGTTTTTCCT
298--QC8fold--	ATGGGCGCTTCTGCGTCAGTAAGTTGACGCA
299--QC8fold--	CGTTTTCATCGTAACCGTGCATCTGGTGTAG
300--QC8fold--	TACCAATTTTTGACGAGCGCCCTGCGGACGACTTTTTAGGGC
301--QC8fold--	GAAACCAGAGTACCTCGCAACGGTTTACGCT
302--QC8fold--	TGCCTTTGCAAAGCGCCATTCGCTGGTGCCG
303--QC8fold--	AAGGTCTTATGCGGCATAATTTATCCTCATTTTAGTAA
304--QC8fold--	TGCAAGGCTGAATATTAGACATACGCTCGGC
305--QC8fold--	GCCAGTTGATTAAGTTGGGTAACGGATGTGC
306--QC8fold--	ATGTCTTTTTACAGTAGAGTGACGGAGAGCGTTCCAAC
307--QC8fold--	AAACGACGGACGCAATGGAGAAACAATAGCA
308--QC8fold--	AGGCCACGCCAGTGCCAAGCTTGACGTTGTA
309--QC8fold--	CCAGTATTTTTTGTTAACAGTCATGAACTTAATTTTTCCACT
310--QC8fold--	ATTGTTATATTAACACCATCCTTCGGGAGAG
311--QC8fold--	GAGTGGCCCGCTCACAATTCCACTGTGTGAA
312--QC8fold--	GCGTGATTCATTCAGAAGGGGATTTTATTTTTTGGTAT
313--QC8fold--	CAAATGCTTAGGTAATAAGA
314--QC8fold--	TAACCCTGAAAACGAACCATAAAAAAGCCTCCGATATTCAAA
315--QC8fold--	CCTGCATTTTTACGAAAAGAATCCATCTGCTTTTATGG
316--QC8fold--	GGGAGGGTGAGCTTGATGCGGTTCAGAATCT
317--QC8fold--	CTTCCAAGTCAATCCTGACGGTTATACAATT
318--QC8fold--	CTTTCCAGAATGTTTTTGAG

319--QC8fold--	GGAAACCATACAATACCATCAGCTTTACCGTATGGCAGCAAC
320--QC8fold--	CTAGACTTTTAAATTAGAGCACGAGCATCATTTTTTTTTTAAGCTCATTACGCTT
321--QC8fold--	TAAAATTTTTAGTTGTTATAAAGATTTGGAGTTTTTGCATGAAAACATTTC
322--QC8fold--	ATCCACGGGGGTTAGCCTCGGTACGGGTGCC
323--QC8fold--	TAATGAGTAAAGTGTAAAGCCTGGGTCAGGC
324--QC8fold--	ATACGATTTTTCCGGAAGCATGAGCTAACTCATTTTTCATTA
325--QC8fold--	TAGAGCTTTTTCAGCAAAATCAATCATGGTCATTTTTAGCTGTTTCCACAAC
326--QC8fold--	TCGAATTCGTACCAGTAGCA
327--QC8fold--	AGCAAGGCCGGAGGATCCCCGGGTACCGAGCCCATTACCATT
328--QC8fold--	CTGCAGTTTTGTCGACTCTAGAAACGTCACCTTTTTTTTTCCATCGATAGTTACG
329--QC8fold--	CCAGCTTTTTGGCGAAAGGGGCCAGGGTTTTTTTTTCCCAGTCACGCATGC
330--QC8fold--	CTCTTCGCTACAGCACCGTA
331--QC8fold--	CAGAATCAAGGGAAGGGCGATCGGTGCGGGCATCAGTAGCGA
332--QC8fold--	AGGCTGTTTTTGCAACTGTTGTTTGCCTTTAGTTTTTCGTCA
333--QC8fold--	CACCACTTTTTCGGAACCGCCGCCAGCTTTCCTTTTTGCACCGCTTCCATTC
334--QC8fold--	TCGCACTCCATCCCTCAGAG
335--QC8fold--	AGAACCGCCAACAGTATCGGCCTCAGGAAGACCGCCACCCTC
336--QC8fold--	GTTTGATTTTGGGGACGACGCCCTCAGAGCCTTTTTTTTTTCAGAGCCGCTGTGA
337--QC8fold--	GCGAGTTTTTAACAACCCGTCCGTAATGGGATTTTTTAGGTCACGTTGCCA
338--QC8fold--	CAACATTAAACACCAGAACC
339--QC8fold--	CGCCGCCAGCCTTCCTGTAGCCAGCTTTCATACCACCAGAGC
340--QC8fold--	AATAATTTTTTCGCGTCTGGCATTGACAGGAGTTTTTGTTGA
341--QC8fold--	AGTAAGTTTTTCGTCATACATAAGTTTTAACGTTTTTGGTCAGTGCCTCAAA
342--QC8fold--	CTGGTAATGGCTTTTGATGATACACATTACC
343--QC8fold--	AGCATTAAAATGAAGAAAACCACGGAGTGTA
344--QC8fold--	GTATCCTTTTATCTGAATGCCCGTCAAACTATTTTTTTTTATAACGTTGAGTCGG
345--QC8fold--	AACCGATTTTAGAAGACTCAGGCATCAAAAGTTTTTCAATATCAGCCAGAT
346--QC8fold--	CGGGAGGGTACGATGTAGCT
347--QC8fold--	TAAAACAGGTAGGATAAACATCATAGGCAGTTTAGGTGTCTG
348--QC8fold--	GGCAGAAATAAATCCTTTTTCATTAAAGCCGTTCC
349--QC8fold--	GACTGAGCCCCCTTATTTTTTAGCGTTTGCAGAGC
350--QC8fold--	GGAGCGGAGGCATTTTCGGTCATTAGCGCGT
351--QC8fold--	TTTCATCATTATCATCATATTCCCACCAGAA
352--QC8fold--	ATTGCGAAATTATTCATTTTTTAAAGGTGAGGAAT
353--QC8fold--	TGATTGTTGACTTGAGCCATTTGATTATCAC
354--QC8fold--	CGTCACCTGGATTATACTTCTGATATAATCC
355--QC8fold--	AATATTGACGGTTGCGCTCA
356--QC8fold--	GAGGGAAGGTACTGCCCGCTTTCCAGTCGGGACCGATTGAGG
357--QC8fold--	TTAACGTGGTTTACCAGCGCCAAAATAGAAA
358--QC8fold--	ATTCATATCAGATGAATATACAGTTTCAGGT

359--QC8fold--	CGTAAAACAGAAATACCGACCGTGTGATAAAAATTATTTGCA
360--QC8fold--	TAAGGCGTTAACCATATCAA
361--QC8fold--	GGAAGGTTTTGTTAGAACCTAATAAGAATAATTTTTACACCGGAATCATTT
362--QC8fold--	TGCGGATTTTACAAAGAAACTGATTATCAGATTTTTTTTTAATTCATCAAATAAT
363--QC8fold--	GTAACATTATCATAATTACT
364--QC8fold--	GTTTAGTATCTTATTAATTTTAAAAGTTTGAAGAAAAAGCCT

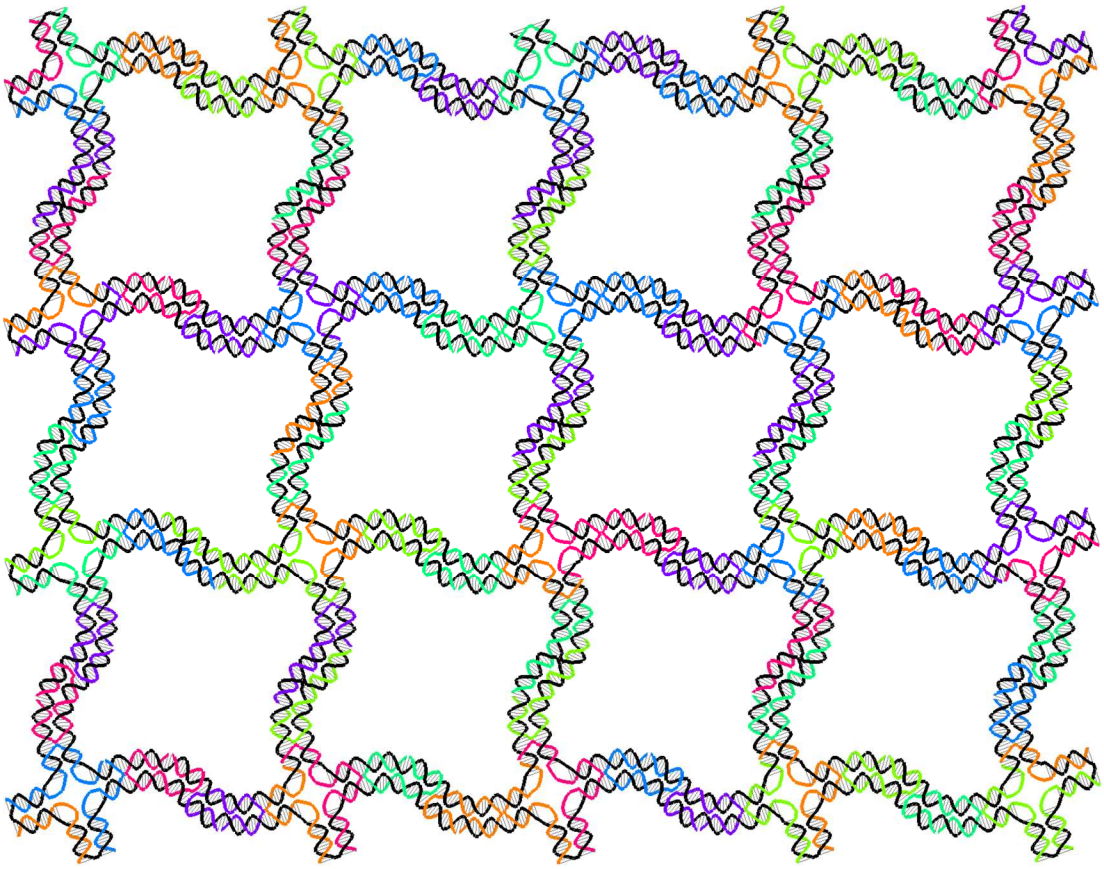


Fig. S74. Design pattern for the waving grid

Table S10. Sequences of the waving grid

Number	Sequence
wavinggrid-1	CTGAACTTTTACCCTGAACAGGAGGTTTAGTTTTTACCGCCACCCTCAGA
wavinggrid-2	TGTATCACCGTACTCAAAGTCAGAGGGTAATTGAGCGATAGCCCGGAATAGG
wavinggrid-3	TAACCCACAAGAATTGGTCGAGAGGGTTGATATAAGTCTAATATCAGAGAGA
wavinggrid-4	TGCTCAGTACCAGGCGTTTTGATAAGTGCCAGTTAAGCCCATTTTATAATAAGAGCCCTG
wavinggrid-5	CTTAGGTTTTTTGGGTTATATCATATTCCTGTTTTATTATCAGATTAGAA
wavinggrid-6	CCTACCTTTTATATCAAAATACGCTGAGAAGTTTTAGTCAATAGTTCCGG
wavinggrid-7	TGAATAATGGAAGGGTGATGGCAATTCATCAATATAATCCTGATAAGTTTATTT

wavinggrid-8	GCCGCCTTTTGCCAGCATTGGGTGGCAACATTTTTATAAAAGAAAGAAAA
wavinggrid-9	TTCATATTTTTGGTTTACCAGGAACCGCCTCTTTTCCTCAGAGCCCCAGA
wavinggrid-10	TGTCACAATCAATACGCAAAGACACCACGGAATTGTTTGGATTATACTTC
wavinggrid-11	ATCGGTTTATCAGCTTTTTTGCTTTCGAGGGCTGTCTTTCCTTTTTTATCATTCCATCGT
wavinggrid-12	AGGAATTTTTCATTACCGCGAGGCTCCAAAATTTTGGAGCCTTTAATTGT
wavinggrid-13	AGCCGTTTTTATTTTCAAGAACGGGTATTAAACCAAGTACCGAGGTCTGAGAGA
wavinggrid-14	CTACCTTTTTAACCGAATTTATCAAAATCATCACTCATCGAGAACAAGCA
wavinggrid-15	TTTTAATTTTAAGTTTGAGTAAAACTTTTTCTTTTAAATATATTTCCGAC
wavinggrid-16	AGACAAAGAACGCGAGAACATTATCATTTTGCGGAACGCAAATCCAATCGCA
wavinggrid-17	AGGAGCGGAATTATCATAACTATATGTAAATGCTGATAAAGAAACCACCAGA
wavinggrid-18	TGTAAATTTTTCGTCGCTATGTCAGATGAATTTTTATACAGTAACACAAA
wavinggrid-19	GATTTTCAGGTTTAACTAATTAATTTTCCCTTAGAATTAAAGAAATTGCGTA
wavinggrid-20	GATAGCTTAGATTAAGTATTTGCACGTAAAACAGAAACCTTGAAAACATAGC
wavinggrid-21	AGAAAAAGCCTGTTTATTATCTAAAATATCTTTAGGAGGAATCATAATTACT
wavinggrid-22	TATACCAAGCGCGAAATTTTCAAAGTACAAGTTGAAAGGAATTTTTTGAGGAAGGGTATC
wavinggrid-23	ATATGCTTTTGTTATACAAAAACACTCATCTTTTTTTGACCCCCAGCGAT
wavinggrid-24	AGATTAGAGCCGTCAACGTTAAATAAGAATAAACACCGCACTAACAACTAAT
wavinggrid-25	CGTGTGTTTTATAAATAAGGTAGATAATACATTTTTTTGAGGATTATTAA
wavinggrid-26	CCTTTGCCCGAACGTTTAGAAGTATTAGACTTTACAAACAATTCACCGACTTGA
wavinggrid-27	CGGAAATTTTTTATTCATTAAGAGCCAGCAATTTTAATCACCAGTCCATC
wavinggrid-28	GCCATTTGGGAATTAAGGTGAATTATCACCGTCGACAACTCGTATTAAAT
wavinggrid-29	AATGGTTTGAAATATAGTTAATTTCATCTTCGTTCAGCTAATGCAGAACG
wavinggrid-30	CGGTCGCTGAGGCTTGTTTTCAGGGAGTTAGACAAAAGGTATTTTAAGTAATTCTATAGA
wavinggrid-31	TAAGTCTTTTCTGAACAAGACATCGCCCACGTTTTCATAACCGATATATT
wavinggrid-32	CGCCTGTTTATCAACAGTCCAGACGACGACAATAAACAACATTGACCTAAATTT
wavinggrid-33	TAGTAATTTTTAACATCACTTCCAGAACAATTTTTATTACCGCCAATAAA
wavinggrid-34	CCTTGCTGGTAATATGCCTGAGTAGAAGAACAGCCACCACCCTCAGAGCC
wavinggrid-35	GATAGCTTTTAGCACCGTAATTTCGGTCATATTTTGCCCCCTTATATTGA
wavinggrid-36	CCTATTTTTTTCGGAACCTAGAGAAGGATTATTTTGGATTAGCGGGGTTT
wavinggrid-37	GCCACCAGAACCACCAGCCACCCTCAGAACCGCCACCCTCAGTCAAACTATCGG
wavinggrid-38	AAAATACATACATAAAACAGGAGGTTGAGGCAGGTCATGTTAGCAAACGTAG
wavinggrid-39	TATTCACAAACAAATAAAGACTCCTTATTACGCAGTAGACGATTGGCCTTGA
wavinggrid-40	AAGAACTTTTTGGCATGATTAATCCTCATTATTTTAAGCCAGAATAGGAG
wavinggrid-41	TGGCTTTTGATGATACGGAAAGCGCAGTCTCTGAATTTACCGCGGTACGCCAGA
wavinggrid-42	CTACGTGAACCATCACTTTTCCAAATCAAGCTAAACAGGAGTTTTGCCGATTAAATTTTA
wavinggrid-43	TAATCATTTTGTGAGGCCACAAACCGTCTATTTTTCAGGGCGATGGCCCA
wavinggrid-44	ATCCTGAGAAGTGTGGGATTTTAGACAGGAATTCCAGTAAGCGTCATACA
wavinggrid-45	TGTACTTTTTGGTAATAAGTAGTAAGCAGATTTTTAGCCGAACAACCCAA
wavinggrid-46	AGCCCTTTTTAAGAAATTTAACGGGGTCAGTGCCTTGTAGCTATCTTACCGA

wavinggrid-47	ATAAACAGTTAATGCCCAAGAAACAATGAAATAGCAAAGTAACAGTGCCCGT
wavinggrid-48	ATAATAACGGAATAAGTTACCAGAAGGAAACTAACGGATTCGCCTGATTG
wavinggrid-49	ATCGCGTTTTCAGAGGCGAATAATTACATTTTTTTAACAATTTCAGCTTC
wavinggrid-50	CTTTGAATACCAAGTTAGTACCTTTTACATCGGGAGAAACAACGAGGAAACGCA
wavinggrid-51	AACCAGAGCCACCACCGCGCCAAAGACAAAAGGGCGAAATCAAAATCACCGG
wavinggrid-52	GGGAGGGAAGGTAAATTAGCGTTTGCCATCTTTTCATCATTCAACCGATTGA
wavinggrid-53	GCGTTTTCATCGGCATTCAGTAGCGACAGAATCAAGTTTGCCGTCACACGACCA
wavinggrid-54	CATTGTGAATTACCTTTTTTATGCGATTTTTCGTCTGAAATTTTTGGATTATTTAATTCT
wavinggrid-55	GGCCAATTTTCAGAGATAGATGAGATGGTTTTTTTAATTTCAACTTTAAT
wavinggrid-56	GTAATAAAAGGGACCATTGGCAGATTCACCATTTAGCGTCAGACTGTAGC
wavinggrid-57	AAGCATTTTTCACCTTGCTGCATGTTACTTATTTTGCCGGAACGAGGCGC
wavinggrid-58	AGACGGTCAATCATAATTTTGGGAACCGAACAGAAGATAAATTTTACAGAGGTGAATCTA
wavinggrid-59	CAGCAGCAAATGAAAAGGCGGTCAGTATTAACACCGCCTGCACAAGGCCGGAAA
wavinggrid-60	CGTCACCAATGAAAAGCACCATTACCATTAGACAGTGCCACGCTGAGAGC
wavinggrid-61	ATCAAGAAAACAAAATTTATTCATTTCAATTACCTGAGCAAAAACAGGGAAGCG
wavinggrid-62	ACCGCCACCCTCAGAATTTTCCGCCACCCTAATGAAAATAGTTTTCAGCCTTTACATTAA
wavinggrid-63	CATTAGACGGGAGAAGAGAGAATAACATAAAAGAAGATGATGAAACAAAC
wavinggrid-64	GTGAGTGAATAACCTTTTTGAATTACCTTTTTTAATGGAAACTTGAAGCCTTAA
wavinggrid-65	CAGCGGAGTGAGAATATTTTGAAAGGAACACGAGGCGTTTTTTTTAGCGAACCTCCTATT
wavinggrid-66	TTGCACTTTTCCAGCTACAATTTTGCTAAACTTTTAACTTTCAACAGTTT
wavinggrid-67	ATCAAGATTAGTTGCCGACTTGCGGGAGGTTAGTACATAAATCAATATAT
wavinggrid-68	GGGCGCTTTTGTACTATGGTCCCCGATTTAGTTTTAGCTTGACGGGGAAA
wavinggrid-69	AAAGCGAAAGGAGCGGTTTTGCGCTAGGGCCTACA
wavinggrid-70	CGCGCTTAATGCGCCGGCTGGCAAGTGTAGCGGTCACGCTGCAGTATTAAGAGG
wavinggrid-71	CTGAGACTCCTCAATTATTCTGAAACATGAAGCGTAACCACCACACCCGC
wavinggrid-72	GAACCCTAAAGGGAGCTGCTTTGACGAGCACGTATAATAAAGCACTAAATCG
wavinggrid-73	AGAATCAGAGCGGGAGTTTTTTGGGGTCGAGGTGCCGCGTGCTTTCCTCGTT
wavinggrid-74	ACGAACTTTTTAACGGAACAGGGTTGAGTGTTTTTTGTTCCAGTTTTGAT
wavinggrid-75	AACGTCAAAGGGCGAACGAGTAAAAGAGTCTGTCCATAAGAACGTGGACTCC
wavinggrid-76	TTGTAGCAATACTTCTTGGAACAAGAGTCCACTATTACACGCAAATTAACCG
wavinggrid-77	TCAACGTAACAAAGCTTTTTGCTCATTCAGTGGCT
wavinggrid-78	ATTAGTTTTTCTTTAATGCGGCTGGCTGACCTTTTTTCATCAAGAGTAAT
wavinggrid-79	AGTAGTAAATTGGGCTACCCTTCTGACCTGAAAGCGTAGAAACACCAGAACG
wavinggrid-80	AGACAATATTTTTGAATGAATAAGGCTTGCCCTGACGAAGAATACGTGGCAC
wavinggrid-81	ACATTTTGACGCTCAAAAGAACTGGCTCATTATACCACTCATGGAAATACCT
wavinggrid-82	AGAAAAATCTACGTTAGCCATTGCAACAGGAAAAACGGTCAGGACGTTGGGA
wavinggrid-83	CCGAACGAACCACCAGCTGACCAACTTTGAAAGAGGATCGCCATTAAAAATA
wavinggrid-84	CAGACCAGGCGCATAGCGAACTGATAGCCCTAAAACACAGATGAACGGTGTA
wavinggrid-85	AATCCGCGACCTGCTCAACCTCAAATATCAAACCCTCGATAAATTGTGTCGA

wavinggrid-86	AGTTGGCAAATCAACACGGAGATTTGTATCATCGCCTAATCAATATCTGGTC
wavinggrid-87	TTACAATTTTAATAAACAGCACCGTAACACTTTTTGAGTTTCGTCACCAG
wavinggrid-88	TTTGTTTAACGTCAAACAGAGCCACCACCCTCATTTTAATAAGAAACGATTT
wavinggrid-89	CAATAGGAACCCATGTCATATTATTTATCCCAATCCACAGGGATAGCAAGCC
wavinggrid-90	TTCCATTAAACGGGTATTTTAAATACGTAACATAT
wavinggrid-91	TTAACATTTTACGCCAACATCGAGGGTAGCATTTTACGGCTACAGAGGCT
wavinggrid-92	CAAAAGAATACACTAATTCTTACCAGTATAAAGCCAACTAAAACGAAAGAGG
wavinggrid-93	CTTAATTGAGAATCGCTGCCACTACGAAGGCACCAACCGCTCAACAGTAGGG
wavinggrid-94	CTTGACAAGAACCGGATTTTTATTCATTACCCAAA
wavinggrid-95	GCCGGCGAACGTGGCGTTTTAGAAAGGAAGGGAAG
wavinggrid-96	TACAAACTACAACGCCTTTTTGTAGCATTCCACAG
wavinggrid-97	TTGAGGACTAAAGACTTTTTTTTTCATGAGGAAGT
wavinggrid-98	CGGTATTCTAAGAACGACTAAAGGAATTGCGAATAATTATAGAAGGCTTATC
wavinggrid-99	ACAGCCCTCATAGTTATTTTGCGTAACGATGCCAG
wavinggrid-100	AATTTTCTGTATGGGATTTTATCCTGAATCTTACCAAAGACGTTAGTAAATG
wavinggrid-101	TTCCAGAGCCTAATTTCTAAAGTTTTGTCGTCTTTCCCGCTAACGAGCGTCT
wavinggrid-102	AAAATCTCCAAAAAAACCCAATAGCAAGCAAATCAGAAATTTTTTCACGTTG
wavinggrid-103	GAGAATATAAAGTACCAAGGCCGCTTTTGCGGGATCGTCGAGCCAGTAATAA
wavinggrid-104	AAAGACAGCATCGGAAGTAATTTAGGCAGAGGCATTTTCACCCTCAGCAGCG
wavinggrid-105	AACCAATCAATAATCGTGAATTTCTTAAACAGCTTGATTACGAGCATGTAGA
wavinggrid-106	CGACAATGACAACAACAAAATAATATCCCATCCTAATTACCGATAGTTGCGC

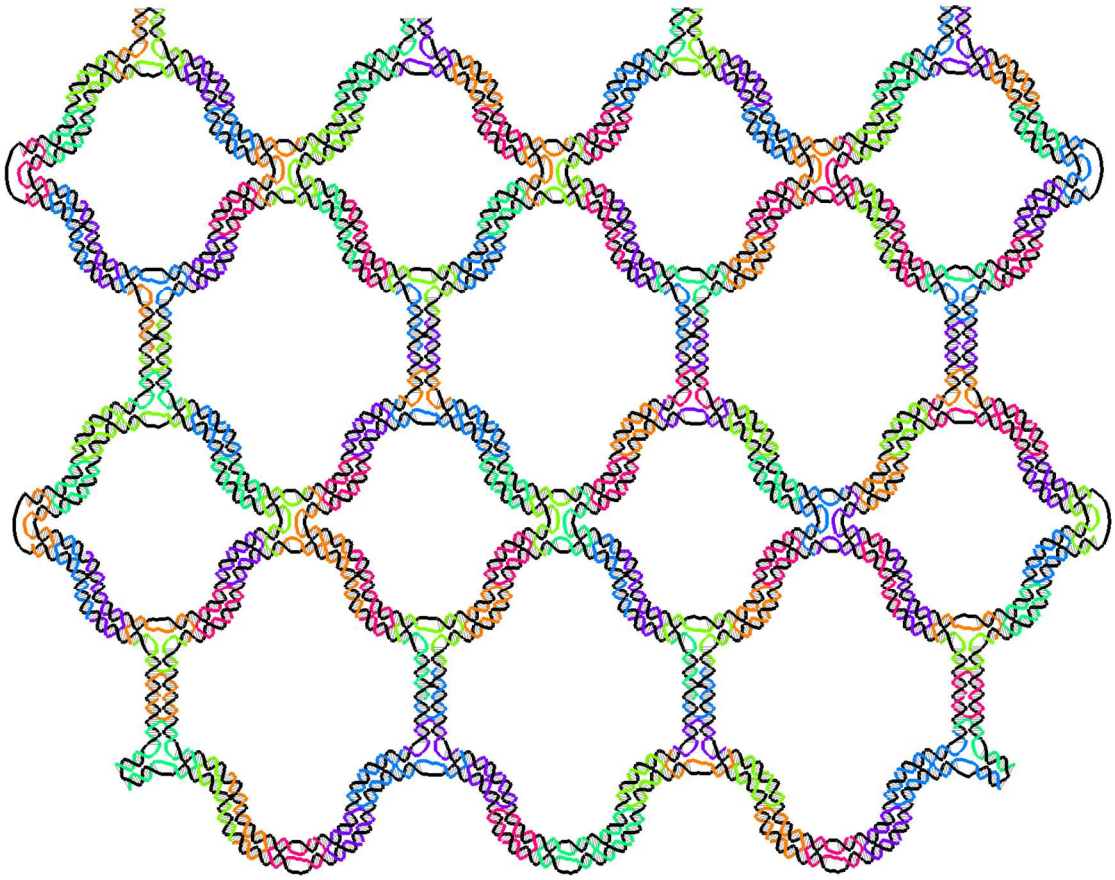


Fig. S75. Design pattern for the fishnet

Table S11. Sequences of the fishnet

Number	Sequence
fishnet-1	CAGAGCTTTGGGAGCTAAAAGAATCCTGAGTTTTTAAGTGTTTTTGGATG
fishnet-2	GTGGCACAGTTTTACAATATTTTGCTCAATCGTCTTTTTAAATGGATTAGATAG
fishnet-3	AACCCTTTTTTCTGACCTGAAAGCGTTACATCG
fishnet-4	CATTCTGGCCAACAGATTTACATTGGCAGATTCACCAGTCACAGGATTTTAGACA
fishnet-5	GGAACGGTACGCCCAGGAGGCCGATTAAAGCGACCAGTAATAAAAGGGA
fishnet-6	GCTTAGTTTAGCTTAATTGCCCAAATCAAGTTTTTTTTTTTGGGGAGAAT
fishnet-7	AGCAATTTTTTTTCTTTGATTGTTTTAATTCGTTTTAGCTTCAAAGCGAAC
fishnet-8	CAGACCGGAAGCAAACTTTTTCCAACAGGTGTTGT
fishnet-9	CTTTAATTGCTCCTTTGTCCATCACGCAAATTAACCCAGGATTAGAGAGTAC
fishnet-10	GAACGTTTTTTACTCCAACGTTGTCTGGAAGTTTTTTTCATTCCAT
fishnet-11	ACTACGTGAACCATCACTGAATATAATGCTGTAGCTCTCAGGGCGATGGCCC
fishnet-12	TGCAACTAAAGTACGGCAAAGGGCGAAAAACCGTCTAAACATGTTTTAAATA
fishnet-13	CGCTTAATGTTTTCGCCGCTACAGAGCTTGACGGTTTTTAAAGCCGGCGAGCGG
fishnet-14	CACTAAATCGGAACCGTATAACGTGCTTTCCTCGTTTCGAGGTGCCGTAAAG

fishnet-15	TGCTTTGACGAGCACCTAAAGGGAGCCCCCGATTTAGGGCGCGTACTATGGT
fishnet-16	GGAGAAACAGTAACAGTACCTTTAAGAATAC
fishnet-17	CGTCAGTTTTATGAATATACAATAACGGATTTTTTCGCCT
fishnet-18	TTAAACCAAGGCGTTTTAGCGAACCTCCCGAAAGAACGGGTA
fishnet-19	CTTTCCTTTTTTATCATTCCCTTGCGGGAGGTTTTTTTTG
fishnet-20	ATTCTATTTTAGAACGCGAGTACCGCACTCATTTTTCGAGAACAAGCAAG
fishnet-21	CATTGAATCCCCCTCATTTTAATGCTTTAAATATC
fishnet-22	AAACCCTTTTTAATCAATATCAATACTGCGGATTTTATCGTCATAAATATT
fishnet-23	AATACCTTTGAACGAACCACCGCCTGCAACTTTTTAGTGCCACGCACTAT
fishnet-24	TATAGTTTTCAGAAGCAAAGTAATATCCAGTTTTTAACAATATTATTAAA
fishnet-25	GTCAGTATTAACACCAGCAGAAGATAAAACTATTAATTTTAAAAGTTTG
fishnet-26	AGGAAAGACTTTACAATTTTTAATTCGACAATGCGG
fishnet-27	AACAAATTTTGAAACCACCAGATACATAACGTTTTCCAAA
fishnet-28	ATTATCATCAATACCACATTCAACTAATGCAGAAGGAGCGGA
fishnet-29	AGTAACATTATCATTTCTCGTATTAAATCCTTTGCCCGAACGTAGAGGTGAGGCG
fishnet-30	CAGTTGTTTTAGATTTAGGATATTCCTGATTTTTTATCAG
fishnet-31	ATGAAAAATCTAAAGAAAAATCAGGTCTTTACCCTGTGAGAGCCAGCAGCAA
fishnet-32	GAATGACCATAAATCCATCACCTTGCTGAACCTCAAACAGTTCAGAAAACGA
fishnet-33	ACTATCGGCCTTGCTGGCGGATTGCATCAAAAAGATTAGTAGAAGAACTCAA
fishnet-34	AGACTTCAAATATCGCAGTAATAACATCACTTGCCTGAAGAGGAAGCCCGAA
fishnet-35	CAGGAAAAACGCTCAGATAGCCCTAAAACATCGCCACCGCCAGCCATTGCAA
fishnet-36	TTTAATGCGCGAACTTGGAAATACCTACATTTTGACTGAATGGCTATTAGTC
fishnet-37	CCCGGGTTTTTCCGAGCTCGAAGGAAGATCGCTTTTACTCCAGCCAGCTTT
fishnet-38	TTAAATGTGAGTTTTCGAGTAACAATTAAA
fishnet-39	TCTTTTCACTTTTCAGTGAGACGTTAATTGCGTTTTTTTGCTCACTGCCGAGGC
fishnet-40	CCTTCACCGCCTGGCTAATGAGTGAGCTAACTCACAGGCAACAGCTGATTGC
fishnet-41	AAAGCCTGGGGTGCCCCTGAGAGAGTTGCAGCAAGCCGGAAGCATAAAGTGT
fishnet-42	TTGGTGTTTTAGATGGGCGAATTCCACACATTTTTACATACGAGCGGTCC
fishnet-43	ACGCTGTTTGTTTGCCCCAAATCCCTTATATTTTTAATCAAAAGATCACG
fishnet-44	CCGAAATCGGCAAGCAGGCGAAAATCCTGTGCGAAAGGAGCGGGCGCTA
fishnet-45	TCACGCTTTTTGCGCGTAACCACCACTATGTGA
fishnet-46	GTGAATAAGTACATAAATCAATAACCCGCCG
fishnet-47	GGGCGCTGGCAAGTGTAACGTGGCGAGAAAGGAAGGGAAGAAATTGATGGTGGTT
fishnet-48	TTTTTTTTTTAATGGAAACACCTTGCTTCTGTTTTTAAAT
fishnet-49	TTGAGTGTTGTTCCAGATTGACCGTAATGGGATAGGATAGCCCGAGATAGGG
fishnet-50	TGGGAACAAACGGCGGTTTGGAACAAGAGTCCACTACCCGTCGGATTCTCCG
fishnet-51	ATTGTTATCCGCTCACCATCGTAACCGTGCATCTGCCTGTTTCCTGTGTGAA
fishnet-52	CGACAGTATCGGCCTCATTCGTAATCATGGTCATAGCAGTTTGAGGGGACGA
fishnet-53	GAGTTAAGCCCAATAATAAAGGTGGCAACATATAAAATAACCCACAAGAATT
fishnet-54	AAGCCTGTTTATCAACTTTTTTAGATAAGTCGCTGT

fishnet-55	GAAAAATCGCCATATTTTTTTACAACGCCAACTTAG
fishnet-56	GTTGGGTTTTTTATATAACTAAGACAAAGAATTTTCGCGA
fishnet-57	CGTCGGAGAATAACATTTTTTAAACAGGGAATTACC
fishnet-58	GATTGTCATTAAAGGTTTTTTATTATCACCGTTTAA
fishnet-59	ATGATGTAAGCGTCATTTTTTATGGCTTTTGTTCAT
fishnet-60	GTAGAAAATACATACATAAGAGCAAGAAACAATGAACAGTATGTTAGCAAAC
fishnet-61	ATTAAGACTTTTTCCTTATTACGATAGCAATAGCTTTTTTCTTACCGAAGCAAT
fishnet-62	AATAACTTTTGGAATACCCAAAAGAATTGCTTT
fishnet-63	CGAGGTGATATCGGTTTATCAGCCTGGCATG
fishnet-64	AAGGAGTTTTCCTTTAATTGATTTCTTAAACTTTTAGCTT
fishnet-65	AGGAAACCGAGGAAACGCCCTTTTTAAGAAAAGTAAGCAGATATGCCAGTTACAA
fishnet-66	AATAAACAGCCATCTTTCCAGAGCCTAATTGCCGAACAAAGTTACCAGA
fishnet-67	AGCAAATTTAGAAGATGATTTGAGCGCTAATTTTTTATCAGAGAGAGAAA
fishnet-68	CGCAAATTTGACACCACGGTTACCAGCGCCTTTTTAAAGACAAAAACCTG
fishnet-69	AAATTCATATGGTAATAAGTTTATTTTGTCCGGTCATAGCCCCCTTATT
fishnet-70	AGCGTTTGCCATCTTTACTGTAGCGCGTTTTCATCGGCATTTTACAATCAATAGA
fishnet-71	ATTGAGGGAGGGAAGGGCGAATTATTCATTTCAATTGGGCGACATTCAACCG
fishnet-72	ACAAAATCGCGCAGAGTAAATATTGACGGAAATTATCTTTGAATACCAAGTT
fishnet-73	TGCGTAGATTTTCAGGTCACCGACTTGAGCCATTTGGCAGAAATAAAGAAAT
fishnet-74	AAATCACCAGTAGCACCAAAATTATTTGCACGTAAAAGAATTAGAGCCAGCA
fishnet-75	TTAGAATTTCCTACCATATCATTACCATTATTTTTGCAAGGCCGGCAGAA
fishnet-76	CCACCATTTCCAGAGCCGCCTTGATATTCATTTTTCAAACAAATAAAGGG
fishnet-77	ACCATCGATAGCAGCCACCACCCTCAGAGCCGCCACAAACGTCACCAATGAA
fishnet-78	CGCCACCCTCAGAGCACCGTAATCAGTAGCGACAGACCGCCACCCTCAGAAC
fishnet-79	GGAACCGCCTTTTTCCCTCAGAGATCAAGTTTGCTTTTTTTAGCGTCAGTCATA
fishnet-80	ATCAAATTTTATCACCGGAACCAGAGTCTAAAG
fishnet-81	TTTTGTCGTAGTTAGCGTAACGACCACCACC
fishnet-82	CCACAGTTTTACAGCCCTCATCTTTCCAGACTTTTGTTAG
fishnet-83	TCAGACGATTGGCCGCCAGCATTGACAGGAAGTACCAGGCGGATAAGTG
fishnet-84	GATGGAGAGGCTGAGATTTTTCCTCAAGAGATATAA
fishnet-85	GTATAGTTTTCCCGGAATAGAGTAAATTGGGTTTTCTTGA
fishnet-86	TCATTCAGTGAATAAGTTTTGCTTGCCCTGGTTTA
fishnet-87	GTACCGTTTTCCACCCTCAGATTACCCAAATTTTTTACGTAACAAAGCTGC
fishnet-88	CCGTCGAGAGGGTTGAAGGATTAGGATTAGCGGGGTTTTGCTCGGTTGAGGCAGG
fishnet-89	TACTCAGGAGACGAGAAACACCAGAACGAGTGTGTATCACCG
fishnet-90	AGAATGGAAAGCGCAGATTATACTTCTGAATAATGGAATCCTCATTAAAGCC
fishnet-91	AATCCTGATTGTTTGGTCTCTGAATTTACCGTTCCAGGCAATTCATCAATAT
fishnet-92	CAAAGTCAGAGGGTAAGAAACAAACATCAAGAAAACAACTGAACACCCTGAA
fishnet-93	AACAATTTCATTTGAAGCGCATTAGACGGGAGAATTAAAATTAATTACATTT
fishnet-94	CCTTAGAATCCTTGAGAAAATAGCAGCCTTTACAGACTATTAATTAATTTTC

fishnet-95	TTTAACGTCAAAAATAAACATAGCGATAGCTTAGATAAGAAACGATTTTTTG
fishnet-96	CAACGCTTTTAACGAGCGTATTATTTATCCTTTTTCAATCCAAATTAAGA
fishnet-97	CGCTGATTTGAAGAGTCAAACCGACAAAAGTTTTTGTAAAGTAATCTTAC
fishnet-98	TAAGAGAATATAAAGTTAGTGAATTTATCAAAATCATTTTTCGAGCCAGTAA
fishnet-99	CCTTTTTAACCTCCGGCATGTAATTTAGGCAGAGGCAAGGTCTGAGAGACTA
fishnet-100	CAATAAACAACATGTAGCTACAATTTTATCCTGAATTCTGTCCAGACGACGA
fishnet-101	TGCTATTTTGCACCCTCAGCTAATGCAGAACGCGCCTTAAATCAAGATTAGT
fishnet-102	GGTTTGTTTTCGTATTGGGCGCCAGGTGCAAAT
fishnet-103	CCAATCGCATATGTAAATGCTGAGTGGTTTT
fishnet-104	TAGAAATTTGGAACAACTACGATTATACCATTTTAAACAAAGTAGAGAA
fishnet-105	GATACATGCCACTACGTTTTTGGCACCAACCTCCAA
fishnet-106	TCTCCAAAAAAAAGGCTAAAACGAAAGAGGCAAAAGATTTTCACGTTGAAAA
fishnet-107	CATCTTTGACCCCCAGAAGGAATTGCGAATAATAATTATACACTAAAACACT
fishnet-108	TAAATCTCCATGTTACTTTTTAGCCGGAACGGCATT
fishnet-109	ATTTTGCTAAACAACTGTGTCGAAATCCGCGACCTGGAATTTTCTGTATGGG
fishnet-110	CATCGCCTGATAAATTTTCAACAGTTTCAGCGGAGTCAACGGAGATTTGTAT
fishnet-111	ACCGCCACCCTCAGACTTGACAAGAACCGGATATTCAACCGCCACCCTCAGA
fishnet-112	TTCATCAAGAGTAATGCCACCACCCTCATTTTCAGGCATAGGCTGGCTGACC
fishnet-113	CAAGCCTTTCAATAGGAACACAGATGAACGTTTTAGACCAGGCGGATAG
fishnet-114	CCAACTTTGAAAGAGGCCATGTACCGTAACACTGAGTAGGGAACCGAACTGA
fishnet-115	AACTACAACGCCTGTAAGGCGCAGACGGTCAATCATATTCGTCACCAGTACA
fishnet-116	CCGTTTTTATTTTCATTTTTTTAGGAATCATCCGGT
fishnet-117	AATGACAACAACCATATTAAACGGGTAAAATACGTACGATAGTTGCGCCGAC
fishnet-118	CATGAGGAAGTTTCCCGCCCACGCATAACCGATATAGGACTAAAGACTTTTT
fishnet-119	TCGCTGTTTAGGCTTGCAGGGGTAGCAACGTTTTGAGGCTTTGATTCGG
fishnet-120	GACAGCATCGGAACGAGGAGTTAAAGGCCGCTTTTGCGGGATCAGCAAATCAGA
fishnet-121	TATAGAAGGCTTATTACCGCGCCCAATAGCAGTCACCCTCAGCAGCGAAA
fishnet-122	GGAAGGTTATCTAAAAAGAAGTTTTGCCAGAGGGGGTGTTGAAAGGAATTGA
fishnet-123	CTTTTGCAAATATCTTTAGGATTTTTGCACTAACAAGATAAAAACCA
fishnet-124	GTCAATAGATAATACAACCCTCGTTTACCAGACGACCTAATAGATTAGAGCC
fishnet-125	GAGCAACACTATCATATTTGAGGATTTAGAAGTATTTTACGAGGCATAGTAA
fishnet-126	AGACTGGATAGCGTCCTGGTCAGTTGGCAAATCAACAAATAGTAAAATGTTT
fishnet-127	GGGAAGAAAAAACAGTGCCCGTTTTTTATAAACAGTGCTCATTATAC
fishnet-128	ATCATTGTGAATTACTTCTGAAACATGAAAGTATTATTTAATTTCAACTTTA
fishnet-129	TTTCGGAACCTATTACTTATGCGATTTTAAGAACTGTAATGCCCCCTGCCTA
fishnet-130	GGTCAGTGCCTTGAGTATCTACGTTAATAAAACGAACAATAAGTTTTAACGG
fishnet-131	ATTACAGGTAGAAAGAATGATACAGGAGTGTACTGGTTAACGGAACAACATT
fishnet-132	TAAGGCGTTAGTTTAGTATCATTTTTTATGCGTTATTTGAAATACCG
fishnet-133	TTAGTTAATTTCATCAACAGTAGGGCTTAATTGAGACTTTTTCAAATATATT
fishnet-134	ATAAAGCCAACGCTCTTCTGACCTAAATTTAATGGTACAAATTCTTACCAGT

fishnet-135	ACTAGAAAAAGCCTAATAAGAATAAACACCGTAATTTACGAGCATGTAGA
fishnet-136	AACCAATCAATAATCGCTGAACAAGAAAAATAATATCCCATCCGAATCATAATT
fishnet-137	GCCAGCTGGCAGGGTTTTCCCTTTTTAGTCACGACGCGGTGCGGGCC
fishnet-138	CCGGCACCGCTTCTGGTTTTTGCCGGAAACGGATC
fishnet-139	CGCCATTCAGGCTGCTGCCTGCAGGTCGACTCTAGACAGGCAAAGCGCCATT
fishnet-140	AGTGCCAAGCTTGCAGCAACTGTTGGGAAGGGCGATTTGTAAAACGACGGCC
fishnet-141	AGTTGGGTAACGCCGAAAGGGGGATGTGCTGCCAGCTGCATTAATGAATC
fishnet-142	GGCCAACGCGCGGGGACGCTTTCCAGTCGGGAAACCTGTCGTGCAAGGCGATTA
fishnet-143	CCGAGTAAAAGAGTCTTGATAAGAGGTCATTTTTGCATAATCAGTGAGGCCA

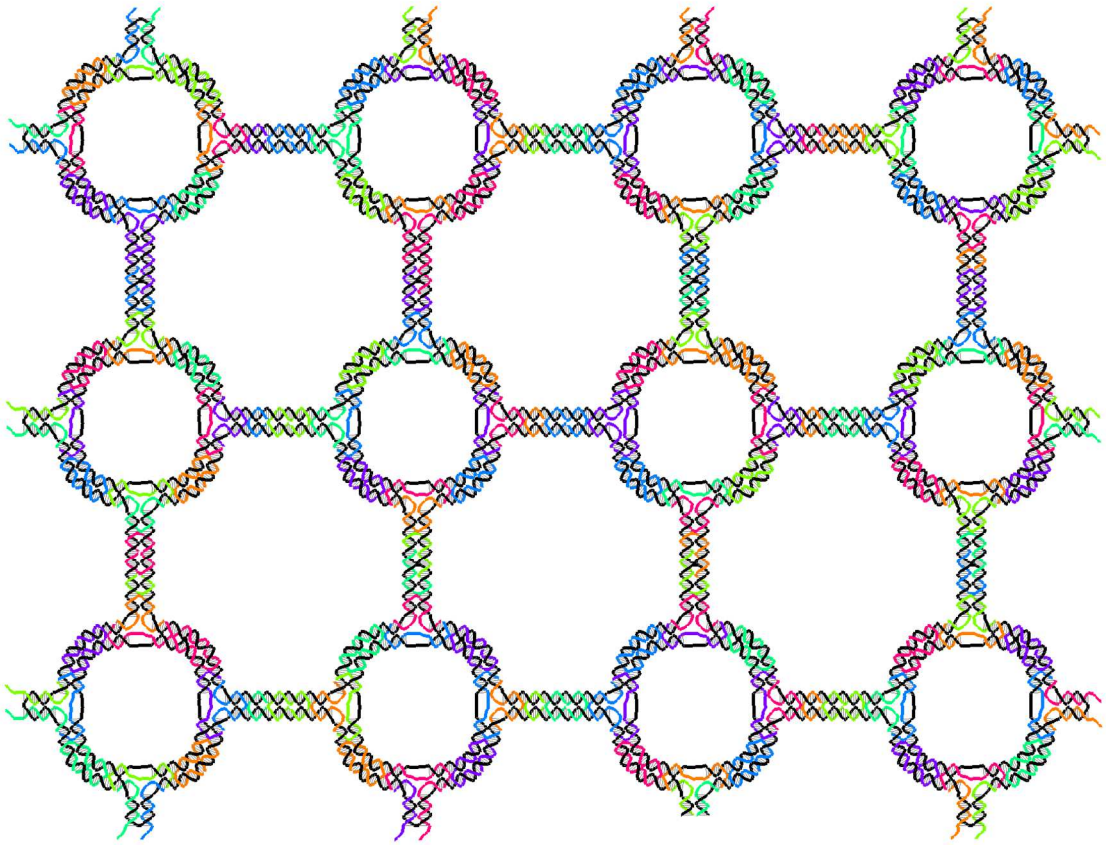


Fig. S76. Design pattern for the circle array

Table S12. Sequences of the circle array

Number	Sequence
spherearray-1	CCTTATTTTTGCGATTTTAAGAACTTTTT
spherearray-2	TTTTGGCTCATTATACCAGTTTTTCAGGA
spherearray-3	ATACCAAGCGCTGAGTAGAA
spherearray-4	TAATAATTTTCATCACTTGCCGAAACAAAGTTTTTACAAC

spherearray-5	GGAGATGCTCCATGTTTTTTTTAGCCGGAACGATTAG
spherearray-6	TAGCAATACTTCTTTGAGGCGCAGACGGTCAATCATAGCAAATTAACCGTTG
spherearray-7	ATCACAGGGAACCGAATTTTTTCCAACTTTGAAACGG
spherearray-8	AACAACTTTTATTATTACAGGTAAAAGAGTCTTTTTGTCC
spherearray-9	GTTAATAAAACGAACTAAGAGGACAGATGAACGGTGTGGAAGAAAAATCTAC
spherearray-10	CGTTGACAGACCAGGCTTTTTTAGGCTGGCTGAATTA
spherearray-11	AACTTTAATCATTGTGACCTTCATCAAGAGTAATCTTAGATGGTTTAATTTC
spherearray-12	GCTTGGACAAGAACCGTTTTTTTTCATTACCCACAAA
spherearray-13	GCTGCTTTTTCATTCAGTGAACGAGTAGTAATTTTATTGG
spherearray-14	CCTGATAAATTGTAACGTAAAATCGTCGAAATCCGCGACCTTTGTATCATCG
spherearray-15	GCGCCAAAACGAGAAACACCAGAATAAGGCT
spherearray-16	TGCCCTGGACAAAAGGGCGACATGTTTACCA
spherearray-17	TCATCAGTTGATATAATCAGTGAGGCCACCGAGTAGAAAGAT
spherearray-18	CGTTGAAAATCTGAAAGCGT
spherearray-19	ATCGGCCTTGCATCTTTGACCCCCAGCGATTGAACTCAAACT
spherearray-20	AATACATTTTCTAAAACACTCTGGTAATATCTTTTCAGAA
spherearray-21	CAATATTGCAGGGAGTTTTTTTGGCCGCTTTTAAAAG
spherearray-22	GCAACAGGAAAAACGCGATATATTCGGTCGCTGAGGCTTACCGCCAGCCATT
spherearray-23	TCACACAACAACCATCTTTTTTACGCATAACCTCATG
spherearray-24	GAAATATTTTCCTACATTTTTTGGCAGATTCTTTTACCAG
spherearray-25	GCGAAAAAAATGGATTATTTACAGACGCTCA
spherearray-26	ATCGTCTGACCGTCTATCAGGGCGTCAAAGG
spherearray-27	GGGACATTCTGGCCAACGATAGTTGCGCCGACAATGACGACCAGTAATAAAA
spherearray-28	CCAAAGTGAATTTCTTTTTTTTAGCTTGATACCAGAG
spherearray-29	ATAGAATTTTCCCTTCTGACCTCCAAAAAAATTTTAGGCT
spherearray-30	TTGTATCGGTTTAACGGCTTAGCATCAGCTTGCTTTCGAGAGGAGCCTTTAA
spherearray-31	CTAAAACGAAAGAGGCGCGGGATCGTCACCCTCAGCATACGAAGGCACCAAC
spherearray-32	GCCACGCGAAAGACAGTTTTTTGGAACGAGGGACAGA
spherearray-33	GGCTTTTTTTGAGGACTAAAGGGTAAAATACTTTTGTAAT
spherearray-34	GACAGAATAAGTTTCCATTAAACGACTTTTT
spherearray-35	CATGAGGCAAGTTTGCCTTTAGCTCAGTAGC
spherearray-36	CCAGGCGGATCCTTGCTGAA
spherearray-37	TTTTTATTAATTTTAAAAGTTTTTTTGAG
spherearray-38	TTAAATTTTTCCTTTGCCCGAACGTTTTT
spherearray-39	GCACAGACAATTGCGAATAATAATTTTTTCAAAGAATACGTG
spherearray-40	AAAATCTTTTTAAAGCATCAAAGTGCCGTCGTTTTAGAGG
spherearray-41	GTTGAGGTTTAGTACCTTTTTTCCCTCAGAACAATGA
spherearray-42	CCGGAATAGGTGTTAGTTACCTCAATCACCGTACTCAGGATATAAGTATAGC
spherearray-43	CTGAGAGCCAGCAGCACGCCACCCTCAGAACCGCCACTGCAACAGTGCCACG
spherearray-44	CCGCCCCTCAGAGCCATTTTTTCCTCATTTTCTCGCC

spherearray-45	TGATAGCCCTAAAACAAGGGATAGCAAGCCCAATAGGCTTTAATGCGCGAAC
spherearray-46	TTAGTAACCCATGTACTTTTTTACACTGAGTTAGAAA
spherearray-47	TCAGCGGAGTGAGAATTCGTCACCAGTACAAACTACACAACTTTCAACAGTT
spherearray-48	CTAAAACGCCTGTAGCTTTTTTCACAGACAGCGCGTA
spherearray-49	ACGATCTTTTTAAAGTTTTGTCTGTATGGGATTTTTTTTG
spherearray-50	CAGAGCCGTTAGTAAATGAATTTTCGTCTTT
spherearray-51	CCAGACGCCGCCAGCATTGACAGACCACCAC
spherearray-52	GGAACATTTTACTAAAGGAATATTTTTGAATTTTTGGCTA
spherearray-53	ATTAAATTTTAATACCGAACGCGGTCAGTATTTTTTAACA
spherearray-54	AAGCCTGATAAAACAGAGGTGAGGAACCACC
spherearray-55	AGCAGAAGGGGTGCCTAATGAGTAAAGTGTA
spherearray-56	AAACCCTCAATTAGCGGGGTTTTGCTCAGTACCTCAAATATC
spherearray-57	AAGAGATTTTAGGATTAGGATCAATATCTGGTTTTTCAGT
spherearray-58	TGGCAAACCTACCATATTTTTTAATTATTTGCTCCTC
spherearray-59	GGAATTGAGGAAGGTTTCTGAATAATGGAAGGGTTAGAATCAACAGTTGAAA
spherearray-60	TTTAGTAATCCTGATTTTTTTTGGATTATACTATCTA
spherearray-61	AAATATTTTTCTTTAGGAGCATAATACATTTTTTTGAGGA
spherearray-62	AGGGGACTTAGAGCCGTCAATAGACTAACAA
spherearray-63	CTAATAGAGACGACAGTATCGGCCCAGTTTG
spherearray-64	CAAACAATTCGACAACGATGATGGCAATTCATCAATAAAGTATTAGACTTTA
spherearray-65	TAACAATTATCATCATTTTTTTCTGATTATCATCGTA
spherearray-66	GGAACAAAGAAACTGAATATCAGACACCAGAAGGAGCGGATTATCATTTTGC
spherearray-67	ATTAAGAGGCTGAGACACGTAAAACAGAAATAAAGAACTGAAACATGAAAGT
spherearray-68	TTATTATTGCGTAGATTTTTTTAGGTTTAACGTACAG
spherearray-69	TAACAGTTTTTACCTTTTACCCTATTTCGGATTTTACCTA
spherearray-70	ACAATAACGGAATAAACAGTTAATGCCCCCTGATCGGGAGAA
spherearray-71	TTTTATTGTATAAGCAAATATTTTTTTAA
spherearray-72	AAAAGCTTTTCCCAAAAACAGGAAGTTTT
spherearray-73	GAGCAATTTTCACTATCATAACCCTTTTT
spherearray-74	TTTTCGTTTACCAGACGACGTTTTATAAA
spherearray-75	CCAGGGCCCTTCACCGTTTTTTGCCCTGAGAGGGTCA
spherearray-76	GAAGCAAACTCCAACAAGTTGCAGCAAGCGGTCCACGAAGCGAACCAGACCG
spherearray-77	CTTCACTGGTTTGCCCTTTTTTAGGCGAAAATATTAA
spherearray-78	GGATTGCATCAAAAAGCCTGTTTGATGGTGGTTCCGAAGTCAGAAGCAAAGC
spherearray-79	ATTATAATCGGCAAAATTTTTTTTATAAATCAGCCGT
spherearray-80	TTTTTGGGGTCGAGGTAAAGAATAGCCCGAGATAGGGTCACCCAAATCAAGT
spherearray-81	AACCATTGAGTGTTGTTTTTTTGTTTGGAACATTAAA
spherearray-82	TTCACCAGTGAGACCACTAAGAGTCGGGCAACAGCTGATTGTGGTTTTTCTT
spherearray-83	GAACGTTTTTGGACTCCAACGATGGCCCACTTTTTACGTG
spherearray-84	AAAGCATTTTCTAAATCGGAGGTCTTTACCCTTTTTGACT

spherearray-85	TCAAAAATCAACCCTAAAGGGAGCCCCCGATATGACCATAAA
spherearray-86	TTAGAGCTTGGAAAACGAGA
spherearray-87	TGCTTTTTTTAAACAGTTCAACGGGGAAAGCTTTTCGGCG
spherearray-88	AACGTGGGCGCTGGCATTTTTTTAGCGGTCACTCAAA
spherearray-89	AGGGAAGAAAGCGGACAGGTTTTAAAAGGAGCGGGCGCTAGGCGAGAAAGGA
spherearray-90	ATTCATTGAATCCCCCGCTGCGCGTAACCACCACACCGGAATCGTCATAAAT
spherearray-91	ACTGCCGCCGCGCTTATTTTTTGCCGCTACAGTGCCA
spherearray-92	TTTGCAAAAGAAGTTTGGCGCGTACTATGGTTGCTTTAAATAGCGAGAGGCT
spherearray-93	AACCAGACGAGCACGTTTTTTTCGTGCTTTCCAGTAA
spherearray-94	AGGAATTACGAGGCATTCGTTAGAATCAGAGCGGGAGATACATAACGCCAAA
spherearray-95	TGCAGCTAAACAGGAGTTTTTTATTAAAGGGAAACGG
spherearray-96	TACGCCTTTTAGAATCCTGATACCACATTCATTTTACTAA
spherearray-97	GATTTAGGAAGAAGTGTTTT
spherearray-98	GAGGGGTTTTGTAATAGTAAAATGTTTTT
spherearray-99	TTTTTTAGACTGGATAGCGTTTTTCCAAT
spherearray-100	GAGGAATTTTGCCCGAAAGACTTCATTTT
spherearray-101	TTTTAATATCGCGTTTTAATTTTTTCGAG
spherearray-102	GGATTATTTTGAGAGTACCTGTTTGCGTATTTTTTGGGCG
spherearray-103	GGGAGAGGCGTTAATTGCTC
spherearray-104	AGGTCATTTTTAATGAATCGGCCAACGCGCGCTTTTGATAAG
spherearray-105	GTCGTGTTTTCCAGCTGCATTGCGGATGGCTTTTTTAGAG
spherearray-106	CTTAAATCCCCGGGTATTTTTTGCTCGAATTCAACCT
spherearray-107	CTGTAGCTCAACATGTCCTGCAGGTCGACTCTAGAGGTTGCTGAATATAATG
spherearray-108	GTAGGCGACGGCCAGTTTTTTTAGCTTGCATGTTTAA
spherearray-109	ATATGCTTTTAACTAAAGTA
spherearray-110	AATGCCTGAGTTTTTAATGT
spherearray-111	GTGAGAAAGGCCGGAGCCCAGTCACGACGTTGTAAAATAAAGATTCAAAAGG
spherearray-112	TTCGCTTAAGTTGGGTTTTTTTCCAGGGTTTTACAGT
spherearray-113	CAAATCTTTTACCATCAATAGATCGGTGCGGTTTTGCCTC
spherearray-114	TGGGAAGGGCTGATATTCAA
spherearray-115	GATAAATTAACCATTCAGGCTGCGCAACTGTCCGTTCTAGCT
spherearray-116	GGCAAATTTTGCGCCATTCGTGCCGGAGAGGTTTTGTAGC
spherearray-117	TATTTGCGAGTAACAATTTTTTTCGGATTCTCAACCA
spherearray-118	AGGCTATCAGGTCATTTTTCATCAACATTAAATGTGATTGAGAGATCTACAA
spherearray-119	AAACTCGCGTCTGGCCTTTTTTTGTAGCCAGCGCCTG
spherearray-120	AGAGTCTTTTTGGAGCAAACAAGAGTTTT
spherearray-121	TTTTAATCGATGAACGGTAATTTTTCGTA
spherearray-122	TGTACCCCGGTTGATAGGAACGCCATCAAAAATAATTAGCATGTCAATCATA
spherearray-123	ATTGTAAATCAGCTCATTTTTTTTAACCAATAATCAG
spherearray-124	TTTGTTAAAATTCGATGGGGTGTAGCATTAAATTTTTGTTAAACGTTAATAT

spherearray-125	CGCTTCTGGTGCCGGACGTGGGAACAAACGGCGGATTCCAGCTTTCCGGCAC
spherearray-126	TCCAGGACCGTAATGGTTTTTTGGTCACGTTGCGCAT
spherearray-127	CGTAACTTTTCGTGCATCTGCTCAGGAAGATTTTTCGCAC
spherearray-128	TGGCGAAAGGGGGTCCACAACAATATGTGCTGCAAGGCGATATTACGCCAGC
spherearray-129	CGCTTTCCAGTCGGGAGTAATCATGGTCATAGCTGTTGTTGCGCTCACTGCC
spherearray-130	ATTGCTCCTGTGTGAATTTTTTTTATCCGCTCCAACA
spherearray-131	TACGAGTTTTCCGGAAGCATGAGCTAACTCATTTTCATTA
spherearray-132	TTTTACAAACATCAAGAAAATTTTCAAAA
spherearray-133	TGAGCATTTTAAAGAAGATGATGAATTTT
spherearray-134	ACGTAGTTTTAAAATACATACATAATTTT
spherearray-135	TTTTAGGTGGCAACATATAATTTTAAGAA
spherearray-136	TTATCGAACAAGCAAGTTTTTTTTTTATTTTCGCGTT
spherearray-137	CATAGCCCCCTTATTAATCGTAGGAATCATTACCGCGATCGGCATTTTCGGT
spherearray-138	TTTTCCCCAATAGCAATTTTTTATCAGATATACCATC
spherearray-139	AACGTCACCAATGAAAGAAGGCTTATCCGGTATTCTAATTAGCAAGGCCGGA
spherearray-140	TTACCAGAACGCGAGGTTTTTTTTAGCGAACCATATT
spherearray-141	TACAAAATAAACAGCCTCCCGACTTGCGGGAGGTTTTGCCTAATTTGCCAGT
spherearray-142	CCAGAGAAGCCTTAAATTTTTTGATTAGTTGCCAGCT
spherearray-143	GGTATTAAACCAATGCACCTATTTGTACCGCACTCATCGAATTCCAAGAACG
spherearray-144	ACAATTTTTTTTATCCTGAATCTTATTTT
spherearray-145	TTTTCCAACGCTAACGAGCGTTTTTCTTT
spherearray-146	ATTTATTTTTCCCAATCCAAATCACCAGTAGTTTTCACCA
spherearray-147	AGCCAGCAAAATAAGAAACGATTTTTTGTTTTTGGGAATTAG
spherearray-148	AACGTCAAAACTTGAGCCAT
spherearray-149	TTATCATTTTCCGTCACCGAATGAAAATAGCTTTTAGCCT
spherearray-150	TTACAAATTAACTGAATTTTTTCTGAACAAAGGTGAA
spherearray-151	TAAAAACAGGGAAAGTTACAACAAGCGCATTAGACGGGAGGAGAGAATAACA
spherearray-152	AAATTATTCATTAAAGTCAGAGGGTAATTGAGCGCTAGGTAAATATTGACGG
spherearray-153	GGGAAATATCAGAGAGTTTTTTCCCACAAGAAAATCA
spherearray-154	AAGTTTATTTTGTCACTTGAGTTAAGCCCAATAATAAAAGACACCACGGAAT
spherearray-155	ACGCAGAGCAAGAAACTTTTTTAAATAGCAATAGCAA
spherearray-156	TATTACGCAGTATGTTAGCTATCTTACCGAAGCCCTTATGATTAAGACTCCT
spherearray-157	CTGGCTTTAAGAAAAGTTTTTTCAGATAGCCGCAGAA
spherearray-158	GGAAACTTTTCGAGGAAACGCAATATTTT
spherearray-159	TTTTATAACGGAATACCCAATTTTAAGAA
spherearray-160	ATAGAATTTTAATTCATATGTCAACCGATTGTTTTAGGGA
spherearray-161	GATAGCTTTTAGCACCGTAAGTCAGACTGTATTTTGCGCG
spherearray-162	TGCCATTTTTCTTTTCATAAAATCGGCTGTCTTTTTTTCC
spherearray-163	ACCAATCAATTCAAAATCAC
spherearray-164	CCACCACCGGCTAATTTACGAGCATGTAGAACGGAACCAGAG

spherearray-165	AAAATATTTTATATCCCATCAACCGCCTCCCTTTTTCAGA
spherearray-166	GCCGCTAACAACGCCATTTTTTGTAATTTAGGCAAGA
spherearray-167	CACCCTCAGAGCCACCTTAATTGAGAATCGCCATATTCACCCTCAGAACCGC
spherearray-168	AGACGTAAAGCCAACGTTTTTTACAGTAGGGCACCCT
spherearray-169	CAGAGCTTTTCGCCACCAGAGAGGTTGAGGCTTTTAGGTC
spherearray-170	CACAAACAAATAAATCTTATACAAATTCTTACCAGTAATTGGCCTTGATATT
spherearray-171	AATAAAAAAGCCTGTTTTTTTTATCATATGCGCTCAT
spherearray-172	TAAAGCTTTTCAGAATGGAATACCGACCGTGTTTTTGATA
spherearray-173	TGGTTTGAAAAGCGCAGTCT
spherearray-174	GTTCCAGTAACATCTTCTGACCTAAATTTAACTGAATTTACC
spherearray-175	ATATTTTTTTTAGTTAATTTGCGTCATACATTTTTGGCTT
spherearray-176	TTGATGCGATAGCTTATTTTTTAAGACGCTGACAAAT
spherearray-177	TGGTAATAAGTTTTAACTTAGAATCCTTGAAAACATAGATACAGGAGTGTAC
spherearray-178	TTACAAATCGTCGCTATTTTTTTTAATTTTCCCGGGG
spherearray-179	TCAGTGTTTTCCTTGAGTAATGCTTTGAATATTTTCCAAG
spherearray-180	TTCGCCTGATCAGTGCCCGT
spherearray-181	GAATTATTCATTTCAAGTGAATAACCTTGCTTCTGTAAAATCGCGCAGAGGC
spherearray-182	TTAATACAGTACATAATTTTTTATATATGTGATTACC
spherearray-183	TTTCATTTGAATTTTAACCCCTTTACCTTTTTTAATGGAATACATTTAACAA
spherearray-184	CGCGAGAAAACTTTTTGAAGAGTCAATAGTGAATTTATCGCAAGACAAAGAA
spherearray-185	TCCAATCAAAATCATATTTTTTTGAGAGACTATCCGG
spherearray-186	CTTAGGTTTTTTGGGTTATATAACTTTTT
spherearray-187	TTTTATATGTAAATGCTGATTTTTGCAAA
spherearray-188	GAATAAACACCGGTAAAGTAAAGGAATCATAATTACTAGAGGCGTTAAATAA
spherearray-189	ATAGATAAGTCCTGAACAGAGGCATTTTCGAGCCAGTGCCTGTTTATCAACA
spherearray-190	AACGCAATAAGAGAATTTTTTTAGTACCGACAAATTC
spherearray-191	TGTCCATTTTGACGACGACAATAAATTTT
spherearray-192	TTTTCAACATGTTCAGCTAATTTTTGCAG

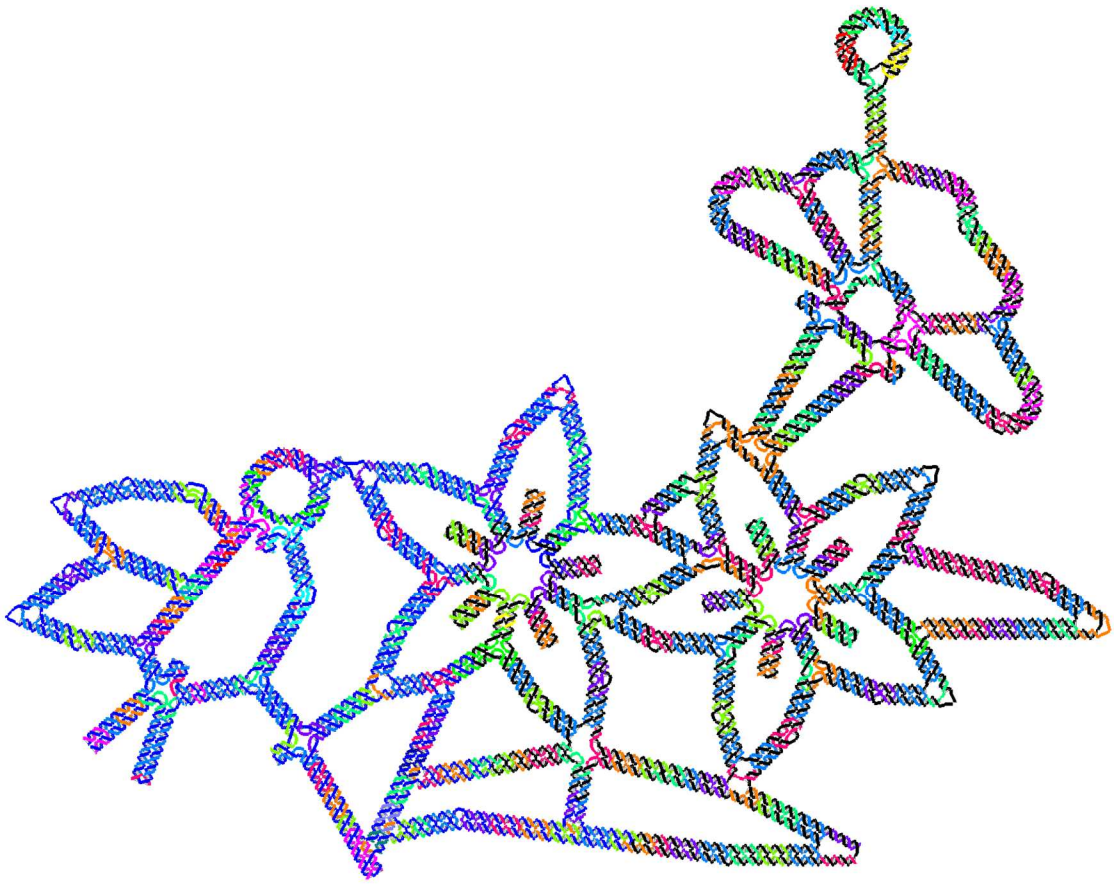


Fig. S77. Design pattern for the flower and bird. The black and dark blue strands are the two scaffold DNAs, and the other colorful strands are the staple strands.

Table S13. Sequences of the flower-and-bird.

Number	Sequence
1--FlowBir--	GTGCCTAATGAGCGTTTTAGCGTTTTAACCTCCCGAAGCTA
2--FlowBir--	CAATTTTTTTTTTTATCCTGAATAACACCAGAACTTTTGAGTAGTAAACGAGCCGGAAG
3--FlowBir--	TTTTGCACCCCTTGCGGGAG
4--FlowBir--	AGTTGCTAGTTTTGAAGCCTTAAACCGAAGCC
5--FlowBir--	TGAGCGCTAATATCAGGTTTAACGTCAAAAAT
6--FlowBir--	CGGTGTTTTTACAGACCAGGGAAAATAGCAGTTCCTTTA
7--FlowBir--	CTTTTTAAAGCAATAGCTATCTTATCAAGATT
8--FlowBir--	GAAAAGTAAGACAATGAAAT
9--FlowBir--	ACAAAGTTACGCCCAATAATAAGAGCAAGAACAGATAGCCGA
10--FlowBir--	CCCACAAGATTTTATTGAGTTAACAGAAGGAAACTTTTCGAGGAAACGCAATA
11--FlowBir--	ATAACGGAATACCCAA
12--FlowBir--	GTCAGAGGGTAATAAGAACTGGCATTTTTGATTAAGACCAACA

13--FlowBir--	ATAAAGGTGGTCCTTATTAC
14--FlowBir--	ATACATACGCAGTATGTTAGCAAAGTACTCAG
15--FlowBir--	GAGGTTTAGAATAGGTGTATCACCCGTAGAAA
16--FlowBir--	TATAAATTTTTTAGAAACGCAATCAATAGAAAATTTTTTTTCATATGGTAACCG
17--FlowBir--	TGTCACAAAGACACCACGGAATAAGCCTATTT
18--FlowBir--	CGGAACCTACAGTTAATGCCCCCTGTTTATTT
19--FlowBir--	CGACATTCTTACCAGCGCCAAAGACCGTTCCA
20--FlowBir--	GTAAGCGTGCAGTCTCTGAATTTACAAAAGGG
21--FlowBir--	ATTGAGTTTTCGGGAGAATTAACTGCACCCTGAACAAA
22--FlowBir--	ATTGACGGAAATTATTAAAACAGGGAAGCGCATTAGAGGAGGGAAGGTAAAT
23--FlowBir--	CAGAGATTGAATAACATACATTAAAGGTGTTTTAATTATCACCCAGC
24--FlowBir--	ATTAGAGCGTCACCGACTTGAGCCGGAACCAG
25--FlowBir--	AGCCACCACATAATCAAAATCACCATTTGGGA
26--FlowBir--	AAAATCATTTTTTCCAGTAGCACATGAA
27--FlowBir--	AAGAGGACAGCATTACCATTAGCAAGGCCGGACCAACTTTGA
28--FlowBir--	AATGAAACCATAATCATAAGGGAACCGAACTGAAACGTCACC
29--FlowBir--	ACCGTAATCAGCGGAACGAGGCGCAGACGGTCCGATAGCAGC
30--FlowBir--	CGCGACCTGCTTATGTTACTTAGTAGCGACAGAATTTCAAGTTTGCCTGTGTCGAAATC
31--FlowBir--	ATCGCCTGATAAATTTTAGCGTCAGACTGTAGCGCGAACGGAGATTTGTATC
32--FlowBir--	TTTTCATCGGCAAACAAAGTAC
33--FlowBir--	TACCAAGCGCGATTTTCGGTCATAGCCCCCTTCCAGCGATTA
34--FlowBir--	ACACTCTTTATCTTTGACCCATTAGCGTTTTTTGCCAT
35--FlowBir--	CTTTTCCGGAACCGCCTTTTCCCTCAGAGCTAAA
36--FlowBir--	AGGCAAAAGAATACACCGCCACCCTCAGAACCGCCACAACCTAAAACGAAAG
37--FlowBir--	ACCACCCTCAGTAATGCCACTACGAAGGCACCCCTCAGAGCC
38--FlowBir--	GAGCCGCCACCGGGTAAAATAC
39--FlowBir--	AGAACCACCACTAGCCGCCGCCAGCATGAGGAAGTTTTTCCATTAAAC
40--FlowBir--	ACTAAAGACTTTTTCATTGACAGGAGGTTGAGGCAGTACAGAGGCTTTGAGG
41--FlowBir--	GGCCTTGATACGGAACGAGGGTAGCAACGGCGTCAGACGATT
42--FlowBir--	AAAGACAGCATTTCACAAACAA
43--FlowBir--	GTCACCTTTTCTCAGCAGCGATAAATCCTCATTTTTTAAAGCCAGTTTTG
44--FlowBir--	CATACATGGCAATGGAAAGC
45--FlowBir--	ATGATATTTTCAGGAGTGTAGCGTAACGATCTTTTTAAAGTTTTGGGATC
46--FlowBir--	ACAGCCCTCATAGTTACTGGTAATAAGTTTTAACGGGTGTAGCATTCCACAG
47--FlowBir--	GAGGCACCAGTACAAATTTCTACAACGCCGTCAG
48--FlowBir--	TGCCTTTTTTGAGTAACAGTAACATGAAAGTTTTATTAA
49--FlowBir--	ATTATTCTGAGCCCGTATAA
50--FlowBir--	CAAGAGAAGGAACCGTAACACTGAGTTTCGTCTGAGACTCCT
51--FlowBir--	AATAGGAACCCATGTTTAGGATTAGCGGGGTTTTGCCAGGGATAGCAAGCCC
52--FlowBir--	CCACCACCCTCATTTTTCAGTACCAGGCGGATAAGTGCCGCCACCCTCAGAG

53--FlowBir--	GGTTGATATAACCTCAGAACCGCCACCCTCAGCCGTCGAGAG
54--FlowBir--	GTACCGCCACGTATAGCCCG
55--FlowBir--	CGCTTTTGCGTCGTCTTTCC
56--FlowBir--	GAGTTAAAGGCAGACGTTAGTAAATGAATTTTGGCTTGCAGG
57--FlowBir--	CGGTCGCTGACTGTATGGGATTTTTTTGCTAAACAGTATCGGTTTA
58--FlowBir--	GCGGAGTGAGAATAGCTCCAAAAGGAGCCTTTAATTACTTTCAACAGTTTCA
59--FlowBir--	AAAAAAAAGGAAAGGAACAACTAAAGGAATTGCGAATAATAATTTTTTCCGATAGTTGCG
60--FlowBir--	CCGACAATGACACTTAAACAGCTTGATACACGTTGAAAATCTCC
61--FlowBir--	CATAACCGATATATTGCTTGCTTTCGAGGTGAATTTACAACCATCGCCCACG
62--FlowBir--	CCTGACGAGACTTACCAACG
63--FlowBir--	CTTTCCAGAGCATTCAGTGAATAAGGCTTGCCTAACGAGCGT
64--FlowBir--	AGTTACAAAAAATCAACGTAACAAAGCTGCTCCTAATTTGCC
65--FlowBir--	ATTATTTATCGAACCGGATATTCATTACCCATAAACAGCCAT
66--FlowBir--	ACCTTCATCAAGAGTATTTTATCTTGACAACCAAT
67--FlowBir--	CGCATAGGCTGGCTG
68--FlowBir--	CCAAATTTTTAAGAAACGATTTTTTAGAGATAA
69--FlowBir--	CATAAAGTGTAAAGCCTGGG
70--FlowBir--	ATGGTTTAATCTCACAATTCCACACAACATATTGGGCTTGAG
71--FlowBir--	TGTAATGTTTCCTGTGTGAAATTTTTGTTATCCGTTCAA
72--FlowBir--	CTTTAATTTTTCATTGTGAACTCGAATTCGTTTTAATCATGGTCATAGCACGTT
73--FlowBir--	GGGTACCGAGTTACCTTATGCGATTTTAAGAAGAGGATCCCC
74--FlowBir--	CTGCATGGTCGACTCTACTGGCTCATTTTCCAGTCAGGAGCATGC
75--FlowBir--	TGCCAAGCTTCGTTGGGAAGAAAAATCTACGACGACGGCCAG
76--FlowBir--	TTAATAAAACGTTCAACATTATT
77--FlowBir--	GATTCATCAGTCCCAGTCACGACGTTGTAAAACAGGTAGAAA
78--FlowBir--	AGTTGGGTAACTGCCAGGGTTTTTGAGATT
79--FlowBir--	TAGGAATACTCAACTAATGCAGATGTCATTTTTGCGGATG
80--FlowBir--	ACATAACGCCATTTGATAAGAG
81--FlowBir--	AATTGCTCCTAAAGGAATTACGAGGCATAGTAGAGTACCTTT
82--FlowBir--	CTATCATAACCACTCCAACAGGTCAGGATTAGAAGAGCAACA
83--FlowBir--	CCGGAAGCAACTCGTTTACCAGACGACGATAAGCGAACCAGA
84--FlowBir--	TAATTTCGAGCTTCAAAAAACCAAAATTTGAGAGGCTTTGCGTTT
85--FlowBir--	TTCAAATATCTGCAAAAGAAGTTTTGCCAGAGCCCGAAAGAC
86--FlowBir--	GGGGGTAATAGTAAAATTTTTTTTTTTTTTAGACTGGATAGC
87--FlowBir--	ATTAAGAGGAAGTCCAATACTGCGGAATCGTCCATCAAAAAG
88--FlowBir--	TTGAATCCCCAGTCAGAAGCAAAGCGGATTGATAAATATTCA
89--FlowBir--	CTGACTATTATCTCAAATGCTTTAAACAGTTCGGTCTTTACC
90--FlowBir--	AAAAATCAAGAAAACGAGAATGACAATTCTGC
91--FlowBir--	GAACGAGTATAACAGTTGATTCCCCATAAATC
92--FlowBir--	AATTGCTGAAGGATGTGCTGCAAGGCGATTAGCTTAGAGCTT

93--FlowBir--	CTGTACGCTATTACGCCAGCTTTTGGCGAAAGGGTATAA
94--FlowBir--	TGCTGTTTTTAGCTCAACATGGAAGGGCGATTTTCGGTGCGGGCCTCTTGCCAG
95--FlowBir--	TGCAACTAAAGATTCAGGCTGCGCAACTGTTGGTTTTAAATA
96--FlowBir--	TACGGTGTCTGTTGACCATTAGACCAGGCAAAGCTGCCATTCGCC
97--FlowBir--	GAAGTTTCATTCCATAGATTTAGTTT
98--FlowBir--	GTGCCGGAAATACATTTCGCAAATGGTCAATCACCGCTTCTG
99--FlowBir--	AACCTGTTTAGTTAAAGAATTAG
100--FlowBir--	AATAAAGCCTGCACTCCAGCCAGCTTTCCGGCAAAATTAAGC
101--FlowBir--	ATAAATTAAATCGGTTGAGTATCGGCCTTCAGGAAGATCCAGAGC
102--FlowBir--	TTATGACCCTCCAGTTTGAGGGGACGACGACTACCAAAAACA
103--FlowBir--	CGTAACTTTCGTGCATCTGGTAATACTTTTTTTTGCGGGAG
104--FlowBir--	AAGCCTTTCATCACCCAAATCAAGGGCCCACT
105--FlowBir--	ACGTGAACATTTCAACGCAAGGATTTTAAAAATTTTTCGCAT
106--FlowBir--	AGAACCCTCATGTAGATGGG
107--FlowBir--	GGATAGTTTTTTGTCACGTTGGTATATTTTAAATTTTTTTGCAATGCCTAATCA
108--FlowBir--	GAGACAGTCAGAGTAATGTG
109--FlowBir--	AGGGTGAGAAAGGCCGTAGGTAAAGATTCAAA
110--FlowBir--	CCATCATTTTTTATATGATATTAGCTGATAAATTTTTTTTAATGCCGGAAGTCT
111--FlowBir--	CAACCGCAAAATCCCTTATAATTTATCAAAAGAAATGGC
112--FlowBir--	TATTAGTTTTTCTTTAATGCCCTGTTTGATGTTTGTGGTTCCGAAATCGGTTCT
113--FlowBir--	AGGCGAAAATGCGAACTGATAGCCCTAAAACTTTGCCCCAGC
114--FlowBir--	ATTAATTTACCGAACGAGCAGCAAGCGGTTCCACGCTGGATCGCC
115--FlowBir--	TGAGAGAGTTACCACCAGCAGAAGATAAAACCCGCCTGGCCC
116--FlowBir--	AGAGGTGAGGCTTTTTTGGTCAGTATTAACACAAGTTTGAGTAACATTACGTTATT
117--FlowBir--	AATTTTAACGCCTGCAACAGTGCCTTTACGCTGAGAGCGCCATATTTATTTACAACGCCA
118--FlowBir--	CCAGCAGCAAAATTGAGAAT
119--FlowBir--	CTCAACTTTTTTAGTAGGGCTTATGAAAAATCTTTTTTTAAAGCATCACATCAA
120--FlowBir--	CAGTTGGCAACTTGCTGAAC
121--FlowBir--	TCAATCAATATCTGGTCTCAAATATCAAACCC
122--FlowBir--	CAGTTGTTTTTTAAAGGAATTGCTTCTGTAAATTTTTTTCGTCGCTATTCGCTG
123--FlowBir--	AATAACCTTGAGGAAGGTTA
124--FlowBir--	CAGAGATTAGTACATAAATCAATATTTTATGTGAGTGTCTAA
125--FlowBir--	AATATCTTTTTTAGGAGCATTTGAATTACCTTTTTTTTTTA
126--FlowBir--	ATGGAAACAGAGCGCATGACAAGTCAGATGCC
127--FlowBir--	TAGATTAAGAAATTAATTTT
128--FlowBir--	AAACATAGCGATAGCTCCCTTAGAATCCTTGA
129--FlowBir--	AGAAGATTTTTTGTCAATAGTGAATTTTCATCAGG
130--FlowBir--	GAAACATTTAAACTAGGGGCGGCCATCAAAAT
131--FlowBir--	CATAGGTCTTTTTTTGAGAGACTATGCAAATCCAATTTTTTTCGCAAG
132--FlowBir--	GTTAGGAATTTCATTAGAGCCCCGCCTCCAAATTTTCAATTTAGACCACCA

133--FlowBir--	TAAATGCTGACCTTTTTAAC
134--FlowBir--	GTTATATAACTATATGCTCCGGCTTAGGTTGG
135--FlowBir--	ACAAAGAAAAATGCAGCAGCAAGATTTTAATCACGAGTTGCT
136--FlowBir--	AGCTCAGGCGCGAGAAAACTTTTTTTTTTTCAAATATATTGTGAT
137--FlowBir--	GATGAATTTTCTAAGTCAACGCACGCTCCCATTTAGCATTA
138--FlowBir--	TACCGACCGTTTAGTTAATT
139--FlowBir--	ATTTAATGGTTTGAAATCATCTTCTGACCTAA
140--FlowBir--	AAATAATTTTTTGGCGTTAAATACCGGAATCATTTTTTTAATTACTAGACAACG
141--FlowBir--	AAGAAGTGAAGTCGCCGACTGTTTAATGCCAGCATCTGT
142--FlowBir--	TGATAATTTTGCAAGCATCTATATACCTGGTTTTCTTTCGTATTCTGGCTAAAC
143--FlowBir--	AGTATAAAGCAAAAGCCTGT
144--FlowBir--	TATACAAATTCTTACCTTAGTATCATATGCGT
145--FlowBir--	ACATGTAATTTAGGCAGAGGCATGATTGCCCTTCA
146--FlowBir--	AGTGATTGGGCAACAGCTTTTCGAGCCATGTAATAAGAGTTCACC
147--FlowBir--	GGTTTTTCTTAATATAAAGTACCGACAAAAGGGCGCCAGGGT
148--FlowBir--	ATGTAGACAGACGACGACAATAAGTAAAGTAATTCTGTCAACCAAT
149--FlowBir--	ACAACATGTTCTTTTTTAGCTAATGCATGCCA
150--FlowBir--	GCTGCATTTTTAATGAATCGGCCAGAACG
151--FlowBir--	AAACCTGTCGGAACGCGCCTGTTTATCAACATTCCAGTCGGG
152--FlowBir--	ATAGATAAGTCTTTACCGCGCCCGCGTTGCGCTCTACTGCCCGCT
153--FlowBir--	CTGAACAAGATTCATCGTAGGAATCA
154--FlowBir--	CCGTTTTTATTAAAATAATATC
155--FlowBir--	TCGAGAACAAGCAAGCCATCCTAAT
156--FlowBir--	CAATAATCGTGCTGTCTTTCTACCGCACTCATTTTACGAGC
157--FlowBir--	CATTTCGGGTATTAAACCAAGCTTATCATTCCAAGAATGTGC
158--FlowBir--	GTAGACTCCTATCTCTTTTTGAGTCTCATTTCAACGCGAGCA
159--FlowBir--	TGTTTTTATAATAGACGTGCATCTCGGCTAATCTCTTTCGATTGG
160--FlowBir--	TGCTTGGAAATGATTGTCCAGTTGCATTTTAAAACCTGCTGT
161--FlowBir--	TTGATTCTCAAAAATACTGACCAGCCGTTTGGTAAGCTCTTT
162--FlowBir--	TTTGATTGGTCATTGGTAATCCGGCGTCTAACCATACCACATTTC
163--FlowBir--	ATCAGCACCACTCAGCACTAACCTTGCGAGTGCAGAGGAAGC
164--FlowBir--	GATTCTGCGTTATCCTTTCCTTTATCAGCGGGCCATACCGCT
165--FlowBir--	GCGCTTTATGAGAAGACAGACTTGCCATCCAAGTCCAATGCAGTA
166--FlowBir--	AAATATCCTTCCAAATCAAGCAACTTATCAGGACGACATTAG
167--FlowBir--	GTGCCAGCCTCCAAACATAAATCACCTCACTAAACGGCAGAA
168--FlowBir--	ATACCTTAGCAATAGCAGCAACGTACCTTTCAAGAAGTCGAAGCA
169--FlowBir--	TTAGCCATAGATGGCGCCACCAGCAAGAGCACTTTACCAGCT
170--FlowBir--	GGCTTTTTGATTGAATGGCAGATTTAATACCCCACCGGAGGC
171--FlowBir--	GCACTTACCTTGAATGAGCATCACCCATTGCCTACAGTGTAGCAA
172--FlowBir--	TCTGTTATCGATTGTTGAACACGACCAGAAAATGTTTATAGG

173--FlowBir--	GACGTTTGGTATCACGTTCTTGGTCAGTATGACTGGCCTAAC
174--FlowBir--	AAAGGACGGTTTTGCATGAAGTACAGTTCCATCATACATCATAGC
175--FlowBir--	AACAATTTCACTAACAACTAATAGATTAGAGTAATTACATTT
176--FlowBir--	CATCTTAAAACAAAATCCGTCAATAGATTAATACATTTAAACAAA
177--FlowBir--	GAAGATGATGGAGGATTTAGAAGTATTAGACCCTGAGCAAAA
178--FlowBir--	TTCGACAACTACCAGAAGGAGCGGAATTATCTTTACAAACAA
179--FlowBir--	TTGCGTTCAAAGAAACCCGTATTAAATCTCTTTGCCCGAATCATT
180--FlowBir--	ATCATATTCCTTTTTGATTATCAGATATCAAAATTATTTTTTTGCA
181--FlowBir--	ATCAATATAAGGAAGGGTTAGAACCTACCATGATGGCAATTC
182--FlowBir--	TGAATAATTCCTGATTGTTTGGATATTATTTA
183--FlowBir--	CATTGGCACAATCGTCTGAAATGGTATACTTC
184--FlowBir--	GATTCACCAGTTTTGACGCT
185--FlowBir--	AAATACCTACATCACACGACCAGTAATAAAAGACGCTCATGG
186--FlowBir--	AGGGAGCCCCCTTGGACATTCTG
187--FlowBir--	TAGAACCCTTAGTCCACTATTAAAGAACGTGGCCAACAGAGA
188--FlowBir--	GAAATTTAAGAATACGGTTGTTCCAGTTTTGGAACAAGCTGACCT
189--FlowBir--	ATATTTTTGATAGCCCGAGATAGGGTTGAGTTGGCACAGACA
190--FlowBir--	ATTGCCTGAGGAGGGTAGCT
191--FlowBir--	CAAAGGCTATCAGGTCATTTTTGAGAGATCTA
192--FlowBir--	GGAGCATTTTTTAACAAGAGAAGTAATCGTAAATTTTTTACTAGCATGTTGTAT
193--FlowBir--	TCGATACGCGCGGGGAGAGGCTTTTTTTTGCGTATTG
194--FlowBir--	ACAGGAAGATCAATCATATG
195--FlowBir--	CAGAAAAGCCCCAAAATACCCCGGTTGATAAT
196--FlowBir--	AAGCAATTTTTTATATTTAAATAATATTTTGTTTTTTTTAAAATTCGCAAATAA
197--FlowBir--	CGCCATCAAATTAAATTTTT
198--FlowBir--	TTTTAACCAATAGGAAGTTAAATCAGCTCATT
199--FlowBir--	TTCGCGTTTTTTTCTGGCCTTCCTTTCATCAACTTTTTTATTAAATGTGTAATG
200--FlowBir--	GGATTGACCGAGCGAGTAAC
201--FlowBir--	CGTGGGAACAAACGGCAACCCGTCGGATTCTC
202--FlowBir--	CACATTAATTAATAGCAAGCAAATCAGATATGTGAGCTAACT
203--FlowBir--	CAAAGGGCGAAGCACTAAATCGGAACCCTAAGACTCCAACGT
204--FlowBir--	GGGGTTTGGTGCCGTAAAAAACCGTCTATTCAGGGCGATTTTTTT
205--FlowBir--	GATTTAGAGCTGCCAGCCATTGCAACAGGAAAA
206--FlowBir--	CAATATTACCTGACGGGGAA
207--FlowBir--	AATATCCAGAAAGCCGGCGAACGTGGCGAGAACCTTGCTGGT
208--FlowBir--	GAAAGCGAAAGTAGAAGAACTCAAACTATCGGAGGAAGGGAA
209--FlowBir--	GAGCGGGCGCTAGGGCGCTGGCAAGT
210--FlowBir--	TAGCGGTCACGCTGCGCGTAACCACC
211--FlowBir--	ACTTGCCTGAGACACCCGCCGCGCTTAATGCGTAATAACATC
212--FlowBir--	GCGCGTACTATTAGCAATACTTCTTTGATTAGCCGCTACAGG

213--FlowBir--	TTAACCGTTGGGTTGCTTTGACGAGCACGTACATCACGCAAA
214--FlowBir--	CCTCGTTAGAACACCGAGTAAAAGAGTCTGTCAACGTGCTTT
215--FlowBir--	TCAGTGAGGCTCAGAGCGGGAGCTAAACAGGTGTTTTTATAA
216--FlowBir--	TCCTGAGAAGAGGCCGATTA
217--FlowBir--	CGCCAGAAAAGGGATTTTAGACAGGGCATACG
218--FlowBir--	CTCGGCGCATGGGAAAGGTCATGCGAACGGTA
219--FlowBir--	CGTAAGTTTAACGTCATTTTTGATGAATATACATTTCAATTA
220--FlowBir--	ATTTTCAGAACAGAAATAAAGAAATAATAAGA
221--FlowBir--	CGACCAATGGAGTAGTTGAAATGGTTGCGTAG
222--FlowBir--	ACCTTTTACATATCGCGCAGAGGCGAATTATTCAGTAACAGT
223--FlowBir--	AAGTTACAAACGGGAGAAAC
224--FlowBir--	TGAATACCAATAACGGATTCGCCTTAACCAGT
225--FlowBir--	AGTGTTAAAGCGGCATGGTCAATAGATTGCTT
226--FlowBir--	TGTCAGCGTCATAAGTGGTAACGCT
227--FlowBir--	TAACATCAAGGTTTTACCTCCAAATAACCCTG
228--FlowBir--	AAACAAATGTTATAGATATTCAAATGAAGAAA
229--FlowBir--	CAAATTAGCATTTGCGGCACAGA
230--FlowBir--	AAGCAGCTTGAACAAAGAAAC
231--FlowBir--	ATGTCAATAGAAAACATAGTGCCATGCTCAGGCAGACCCATA
232--FlowBir--	GCGCAAGAGTTGTGGTAGAA
233--FlowBir--	GGCGGATTTTAAACGAACAAGTCGTCATTTGTTTTGCGAG
234--FlowBir--	GAGGAAGCGGAACGCAAGGTAAACGCGAACAATCAGTCTCAG
235--FlowBir--	CTGCGCGTACGCAGTCCAAATGTTTTTGAGAGTCAGTGTTTC
236--FlowBir--	CAGTAAGAACTGGCAGCAAC
237--FlowBir--	TTCTTCTGCGTGGAAACCATAA
238--FlowBir--	TCCGCACGTAATTTTTGACGCACGTTCATATCCATTACATTACATCACTCCT
239--FlowBir--	TAATGGATAT
240--FlowBir--	GAGCATCATCTTTTTTACCTTTAGA
241--FlowBir--	CATTAGGGTTTAAACGTGACGTATGAATGCGTTGATTAAGCT
242--FlowBir--	CGCATTCATA
243--FlowBir--	GGTCAGGCATAACTTAATCCACTGTTCACCAAGCCTCGGTAC
244--FlowBir--	TAAAATAGTTGCTTAGGGATTTTCGCGCGTACCACGGCGCTT
245--FlowBir--	TACGCGCGAA
246--FlowBir--	TTATTGGTATCTAACACCATCCTTCATG
247--FlowBir--	TGCCAAGAAACAGTCGGGAGAGGAGTGGCATAGGGTTAATCG
248--FlowBir--	ATGAGGGACATAGGGCGAGCGCCAGAACG
249--FlowBir--	GCGGACGACCAAAAAGTAAA
250--FlowBir--	TAGAGTCAATCTTTAGTACCTCGCAACGGCTAATGTCTACAG
251--FlowBir--	TTTACGCTTGCAAGGCCACGACGCAATGGAGAGACGAGCGCC
252--FlowBir--	CGCCAACGGCGTTGACCACCTACATACCAAAAAGACGGAGAG

253--FlowBir--	CAATTAAAATTGTCCATCTCGAAGGAGTCGCCGAAGCCCCTG
254--FlowBir--	AGTAAGGGGCCAGCGATAACCTCGATAACAGTTTTATCCTCA
255--FlowBir--	ACTGTTATCGA
256--FlowBir--	ATTAGACATAACTGACCAGCAAGGAAGCCAAGCAGTTTGAAT
257--FlowBir--	TAAGTGGCTGGTTCAGCATCAGT
258--FlowBir--	AGCTTGAGTAATTATAATCTCGG
259--FlowBir--	AGAAGGCTTATTTAGAACGCGAG
260--FlowBir--	CGCCATTAATAATGTTTTTTTTCCGTAAATCAGTA
261--FlowBir--	CAGTCGGGAGGGTAGTTTTTTTCGGAACCGAAGAAGACTCAA
262--FlowBir--	TAGGGTCGAGCGAACCAAACAGGCCTGAACGG
263--FlowBir--	ACTGGAAATATTCCACTGCAACAAAAAAAATT
264--FlowBir--	CATCAAAAGCTTTAAGAGCCT
265--FlowBir--	AATATCAGCACAATAGCAGG
266--FlowBir--	CCAACAGAAACTTTAACCTGATTAGAGAAAGAGTATTTTCCACAAGCCT
267--FlowBir--	ATTGGGGATTGCGGCGTTGACAGATGTATCCGGAAGCCAAGC
268--FlowBir--	GTTATTATCTGCTTATATCTGAATGCATATGAAGAAAATGATGCG
269--FlowBir--	TCCAAGAGCTCCACCATTACCAGCATTAACCACAGAATCTCT
270--FlowBir--	AAAATATAACGCCGTCAACATACATATCACCGTCAAACTATC
271--FlowBir--	AGTGATTCAGCCTTATGGTTGACGATGTTAGCTTTAGGTAACAGA
272--FlowBir--	GAGTGGTCCCGAAGAAGCTGGAGTGTCTGTAA
273--FlowBir--	AACAGGTGGGCAGATTGCGATAAATGAATCGC
274--FlowBir--	GAACCAGCTTATCAGAAACGTCAGAAG
275--FlowBir--	GATATTACTCATCACGAAAAAAGTTTGTAATTATGGCGAGTAACA
276--FlowBir--	TTAACTTCTCAGAAATAAAAGTCTGAAACATCAATTCATCCA
277--FlowBir--	TAAGCAGAAATGGTCTATAGTGTTATTAATAGATTAAACTCC
278--FlowBir--	CGTCCTTTCGGGGCGGACCTACCGCGCTTCGCTTGGTATTTTT
279--FlowBir--	CGGCAAAAATTCTTCTCGTTCTCTAAAAACCCAACCCCTCAG
280--FlowBir--	GAATGTGCGGCAAAACTTTTGCGTAACCGTAAAA
281--FlowBir--	TTTTTATTTCCGCTTCGGCAAGGCGGTTCCTTTTGAAT
282--FlowBir--	GTTATAACCTAAATCACCAG
283--FlowBir--	TTTTATCACGCGAATAAGTACGCGTTCTTGCCACACTCAATC
284--FlowBir--	AAGTCATGATCGGTCACATTAAAT
285--FlowBir--	AATCATGGTGGGAGGGCAGATATTAACCTGACCACTGGTCAT
286--FlowBir--	TATCTGCCCTC
287--FlowBir--	CGTTCATTTGCAGCCAGCTGGAAGCCTTCATTTTTTAGAAGGTGATAAGCA
288--FlowBir--	GGAGAAACATACGAAGTTTTTTGCGCATAACGAAACT
289--FlowBir--	ATACCACTGACACGGCAGCAAT
290--FlowBir--	ATAAGCAATGCCTCAGCAATCTTAAACTTCTGGATGAACATA
291--FlowBir--	TGACGTAGACGAATCATTTTCAGAACGGAATCAAA
292--FlowBir--	GCACCTTTTTTAGCGTTAACAGGCCGTTTGTTTAATGT

293--FlowBir--	TAGAAATTTCCGCCAGCAATATCGGTATAAGAACATCCTTCA
294--FlowBir--	AACAGGGTACGCGGCGGCAAGTTGTCAATAGT
295--FlowBir--	CACACAGTTTCCTTGACGGTAACGTTATTGCTTTGTACGGGGAAGGACGCCATACAA
296--FlowBir--	TCTTTAGTCGTCAGCGCCTTCCATGATGAGAGGTACTGAATC
297--FlowBir--	CAACAGTTTGAGCAGGAAACGTCAGAAAATTTTTTTCGAAATCATCGAGGG
298--FlowBir--	CCACAAAGTCAGCAATCCAAACTTTGTTACTGCGAGGGTATC
299--FlowBir--	GAGCGGTCAGTCAGCGTACCAT
300--FlowBir--	GCAGCGACGAGCACGAAAACGCAAGCCTCAAC
301--FlowBir--	CCAAAACGGCACTCATCGCGAATATCCTTAATTCGGTTAAAT
302--FlowBir--	GGGGTAATTATAGAAGCCTGAA
303--FlowBir--	TAATACCTTTCTTTTTTGAGCTTAATAGAGGC
304--FlowBir--	TCAAGTTGGGGTTAGCATTGTGC
305--FlowBir--	ATTATCGAACTTTATACGAAAAG
306--FlowBir--	GGAGTACCAGCGATACGCTCATTTTAAGTCAAGAGCCAAT
307--FlowBir--	CTGGTACTCC
308--FlowBir--	ACCATCAGCTTTACCGTTTCCTAGACAAATTAAATAATCAGCGTGACATTCA
309--FlowBir--	GAAGGGTAATAAGAACGAACCATAAAAAAGCCTCCAAAGCGAAACCAATCCGCGG
310--FlowBir--	AAATAATCTCTTTAATTTTAACCTGATTCGATTT
311--FlowBir--	GGAGGCTTTATGAAAACATATACCATGAAATTTAATATCAACCACACC
312--FlowBir--	GGCAACAGCTTTATCAACAATTGGGAGGGTGTCAATCATTGTTCCAAGTATC
313--FlowBir--	CTGACGGTTATCTTTCCAGAA
314--FlowBir--	CATTTAGTAGCGGTTTTTAAAGTTAGAACCATTCAAAGTTTTTTGATAAACATCATAGG
315--FlowBir--	AACCAACATAATAATAACCACCATCATGGCGCCAAACCATGA
316--FlowBir--	AAAGCTTCAGCGGCTTTTTTTAACCGGACGCTCGA

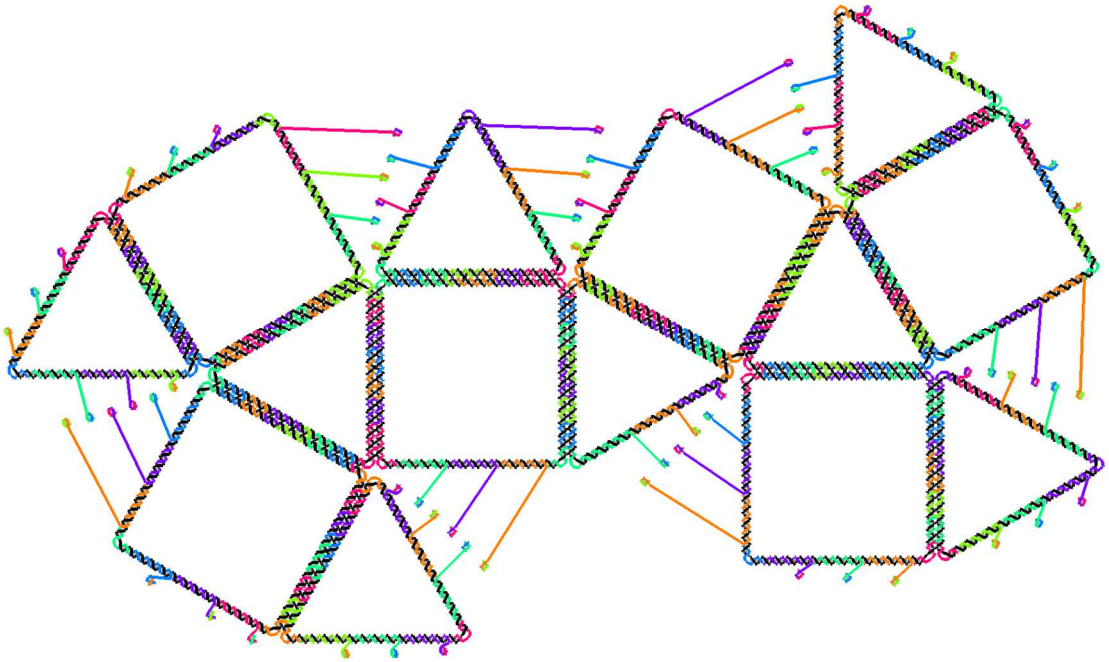


Fig. S78. Design pattern for the 2D open net of the cuboctahedron

Table S14. Sequences of the cuboctahedron frame

Number	Sequence
Frame-1	CGGTATTCTAGAGAACAAGCAAGCCGTTTTTGAAGGCTTATC
Frame-2	ATCATTCCAAGTTTTAACGGGTATTAACCTCCCGACTTTTTTGCGGGAGGAACAG
Frame-3	CCGCACTCATCAGAACGCGAGGCGTTTTAGCGAAACCAAGTA
Frame-4	AAAACAGAGCCCAATAGCAAGCAAATCAGATATAATTTTCATCGTAG
Frame-5	AACAATTCGACTTTTAACTCGTATTCCTGATTGTTTTTTTGGATTATACTCCTTT
Frame-6	TACATCTTTTGGGAGAAACAATTAATTACATTTTTTTAACAATTT
Frame-7	GAATCATTACCGCAATAAAGAAATTGCGTTTGCACGT
Frame-8	GTTTAACGTCAAACCTACCATATCAAAATTATAGATTTTCAG
Frame-9	GAAGGGTTAGGATGAATATACAGTAACAGTATCTGAATAATG
Frame-10	TTATCAGATGTTAATTTTAAAAGTTTGAGTACATATTCCTGA
Frame-11	CCCGAACGTTAATGGCAATTCATCAATATAATAAATCCTTTG
Frame-12	AGTGCCCGACCACCAGAAGGAGCGGAATTATCATACATTATCATTTT
Frame-13	CTCAGAGCCGCTTTTCACCAGAACCTCATACATGGCTTTTTTTTGATGATATTAA
Frame-14	GAGGCTTTTTGAGACTCCTCTACCGCCACCCTTTTTCAGAACCGC
Frame-15	GCGGAACAAAGAATATAAACAGTTAATGCTGAGTAAC
Frame-16	TTTCGGAACCTGTTTTAACGGGGTCAGTGCCTCCCCTGCCTA
Frame-17	CTGGTAATAAATTATTCTGAAACATGAAAGTACAGGAGTGTA
Frame-18	GCAGTCTCTGATTGACAGGAGGTTGAGGCAGGAATGGAAAGC
Frame-19	CCGCCGCCAGCAATTTACCGTTCCAGTAAGCGACCACCAGAG

Frame-20	TTTATCCTAAACAAATAAATCCTCATTAAAGCCAGTCAGACGATTGG
Frame-21	CCATATTTTTTATTTATCCCTGAACAAAGTCTTTTAGAGGGTAAT
Frame-22	CCTTGATATTCACGAATCTTACCAACGCTGCTACAAT
Frame-23	TTTCCAGAGCCTTAGTTGCTATTTTGCACCCAAACGAGCGTC
Frame-24	TAAATCAAGATAATTTGCCAGTTACAAAATATTTTGAAGCCT
Frame-25	ATCACCTTGCTAAAACAGAGGTGAGGCGGTCAAATCTAAAGC
Frame-26	AACATCGCCATTTTTTAAAAATACCTCAATCAATATTTTTCTGGTCAGTT
Frame-27	CCAGCAGAAGATGAACCTCAAATATCAAACCCGAACGAACCA
Frame-28	GCGAAAGACACGCTGAGAGCCAGCAGCAAATGAAAGTATTAACACCG
Frame-29	CGAATAATAATTTTTTTTTTCACGTAACCGATATATTTTTTCGGTCGCTGATGAG
Frame-30	GAAGTTTTTTTCCATTAAACGCCTGATAAATTTTTTGTGTCGAAA
Frame-31	CCTGCAACAGTGCCAGCATCGGAACGAGGCCTCAGCA
Frame-32	CTACAGAGGCTCCGCTTTTGCGGGATCGTCACGTAGCAACGG
Frame-33	GAGTTAAAGGTTGAGGACTAAAGACTTTTTCAGGCTTGCAGG
Frame-34	CCGACAATGACTCCAAAAGGAGCCTTTAATTCGATAGTTGCG
Frame-35	CAAAAAAAAGGCAACAACCATCGCCCACGCATTGAAAATCTC
Frame-36	TGCCGTCGGGTGAATTTCTTAAACAGCTTGATACGTATCGGTTTATC
Frame-37	AGCTTGCTTTCGAAGAGGGTTGATATAAGCGGATAAG
Frame-38	AATAGGTGTATGGGGTTTTGCTCAGTACCAGGTATAGCCCGG
Frame-39	TAGGATTAGCCACCGTACTCAGGAGGTTTAGAAGAGAAGGAT
Frame-40	TCATTAAAGGATTCATATGGTTTACCAGCGCACGGAAATTAT
Frame-41	AGACACCACGGTTTTAATAAGTTTATTGAGCCATTTTTTTGGGAATTAGA
Frame-42	ATCAATAGAAATGAATTATCACCGTCACCGACTTTTGTCACA
Frame-43	GCCTTTACGATTGAGGGAGGGAAGGTAAATATTGCAAAGACAAAAGG
Frame-44	GCGACATTCAACCAGAGAGAATAACATAAAAATAGCA
Frame-45	GCGCATTAGACTTTGTTTAACGTCAAAAATGAAAACAGGGAA
Frame-46	GAAACGATTTGGGAGAATTAACTGAACACCCAATCCAAATAA
Frame-47	AAGCCTGTTTCTAAATTTAATGGTTTGAAATATTACTAGAAA
Frame-48	AAAACTTTTTCTTTTAAATATATTTATTCTTACCAGTTTTTATAAAGCCA
Frame-49	CATCTTCTGACAGTATCATATGCGTTATACAATAGTTAATTT
Frame-50	AGGCGAATAAATAAGAATAAACACCGGAATCATAACCGACCGTGTGA
Frame-51	TAAATAAGGCGTTTATTCATTTCAATTACTCGCGCAG
Frame-52	GAAGATGATGATTTGAATACCAAGTTACAAAACTGAGCAAAA
Frame-53	GCCTGATTGCAACAAACATCAAGAAAACAAAATAACGGATTC
Frame-54	TTTAATCATTGTTTTTGAATTACCTCAACTAATGCATTTTGATACATAACGTAAT
Frame-55	AGTAAATTTTATGTTTAGACACTATTATAGTTTTTCAGAAGCAAA
Frame-56	TTAAATGTGAGCGACGACGATAAAAACCAAACCCTCG
Frame-57	GCTCATTTTTTTTTTAACCAATAGGGTGTAGATGGGTTTTCGCATCGTAAATTCA
Frame-58	GGCTGCTTTTGCAACTGTTGCGACGGCCAGTTTTTGCCAAGCTTG
Frame-59	TTTACCAGAGTAACAACCCGTCGGATTCTCCGTGGCTTTCATCAACA

Frame-60	CGGATTGACCCGTCTGGCCTTCCTGTAGCCAGGAACAAACGG
Frame-61	AAAATAATTCGGTAATGGGATAGGTCACGTTGAACGCCATCA
Frame-62	CCAGTTTGAGCCAGGCAAAGCGCCATTCGCCCCGTGCATCTG
Frame-63	GGTGCCGGAAAGGGACGACGACAGTATCGGCCCACCGCTTCT
Frame-64	TTGCTTTGACGAGCCAGCCAGCTTTCCGGTCAGGAAG
Frame-65	CAAGTGTAGCGTTTTGTCACGCTGCTAGACAGGAACTTTTGGTACGCCAGGTAGA
Frame-66	AGAACTTTTTCAAACTATCGAGATTCACCAGTTTTTCACACGACC
Frame-67	ATCGCACTCACGTATAACGTGCTTTCCTCGTTAGGCGCGTACTATGG
Frame-68	GAGCTAAACACGCTTAATGCGCCGCTACAGGAATCAGAGCGG
Frame-69	CACACCCGCCGGGAGGCCGATTAAAGGGATTTGCGTAACCAC
Frame-70	GTGTTTTTATGTAATAACATCACTTGCCTGAAATCCTGAGAA
Frame-71	TTCTTTGATTAAATCAGTGAGGCCACCGAGTAGTAGCAATAC
Frame-72	CTACGTTAATAAAACGCAAATTAACCGTTAAAGAGTC
Frame-73	TGTCCATCACGAACTAACGGAACAACATTATTACTGGGAAGAAAAAT
Frame-74	AATATCCAGAAAATGGATTATTTACATTGGCGCCTTGCTGGT
Frame-75	TCAATCGTCTGACAATATTACCGCCAGCCATTATTTTGACGC
Frame-76	AGAATACACTAAATCATGGAAATACCTACGCAACAGG
Frame-77	AAAAACGCACACTCATCTTTGACCCCCAGCGATTCGAAAGAGGCAAA
Frame-78	GAAACAAAGTTACGAAGGCACCAACCTAAAAATACCAAGCGC
Frame-79	CGTAATGCCACACAACGGAGATTTGTATCATCGGGTAAAATA
Frame-80	CGGTGCGGGCCCCAGTCACGACGTTGTAAAAGGAAGGGCGAT
Frame-81	GCCAGGGTTTTCTCTTCGCTATTACGCCAGCTGTTGGGTAAC
Frame-82	AGTTGCAGCAAGCGCTGCAAGGCGATTAAGGCGAAAG
Frame-83	GGGCGCCAGGGTTTTTGGTTTTTCTCTTATAAATCATTTTAAAGAATAGC
Frame-84	GGGGATGTGGTCCACGCTGGTTTGCCCCAGCAGGCTGGCCCTGAGAG
Frame-85	TTTGATGGTGCAGCTGATTGCCCTTCACCGCCGAAAATCCTG
Frame-86	GAGACGGGCAAGTTCCGAAATCGGCAAAATCCTTTCACCAGT
Frame-87	AGGCTTTTGCATAGTAAGAGCAACACTATCATAAATAGCGAG
Frame-88	TTACGAGGCAAAAGAAGTTTTGCCAGAGGGGGCCAAAAGGAA
Frame-89	CAATACTGCGAAAAATCAGGTCTTTACCCTGTGGATAGCGTC
Frame-90	GACCATAAATCGAATCGTCATAAATATTCATTAAACGAGAAT
Frame-91	GATAAAAATTTTTCTTTAAACAGTTCAGAGAATCCCC
Frame-92	CGAACGAGTAGTTTTATTTAGTTTGGGCCGGAGACATTTTGTCAAATCAC
Frame-93	CTCAAATGAGAACCCTCATATATTTTAAATGCAAATTTCAACGCAAG
Frame-94	TGTGTAGGTAATGGTCAATAGAGAAGCCTTTTGCCTGAGTAA
Frame-95	ACATTTCGCAAAAGATTCAAAAGGGTGAGAAAACCATTAGAT
Frame-96	TAAGAACTGGCGAGATTTAGGAATACCACATTTATGCGATTT
Frame-97	TTCATCAGTTTCATTATACCAGTCAGGACGTAGGTAGAAAGA
Frame-98	GTCCAGACGACTTTTGACAATAAAC
Frame-99	AACAAAGTTACTTTTCAGAAGGAAA

Frame-100	ATAAAGTGTAATTTTAGCCTGGGGT
Frame-101	ACGGTAATCGTTTTTAAAACTAGCA
Frame-102	ACCTTCATCAATTTTGAGTAATCTT
Frame-103	ACAGACAGCCCTTTTTCATAGTTAG
Frame-104	CGCGTTTTCATTTTTCGGCATTTTC
Frame-105	TTAGATTAAGATTTTCGCTGAGAAG
Frame-106	AATCAAGTTTTTTTTTTGGGGTCGA
Frame-107	TTGCTCCTTTTTTTTGATAAGAGGT

Table S15. Sequences of the set strands for the 2D open net of the cuboctahedron

Number	Sequence
Form-2D-Set-1	TCTGACATAATAAGAGAATATAAAGTACCGACAAAAGGTAAAGTAATTCT
Form-2D-Set-2	GCTAGTCGAACGCCAACATGTAATTTAGGCAGAGGCATTTTCGAGCCAGT
Form-2D-Set-3	AATTCTAGACGCTCAACAGTAGGGCTTAATTGAGAATCGCCATATTTAAC
Form-2D-Set-4	GAGTCGTTTAAATGCTGATGCAAATCCAATCGCAAGACAAAGAACGCGAG
Form-2D-Set-5	TCATGACTCCTTTTTAACCTCCGGCTTAGGTTGGGTTATATAACTATATG
Form-2D-Set-6	CTATCAACAGTCAATAGTGAATTTATCAAAATCATAGGTCTGAGAGACTA
Form-2D-Set-7	CTCATTTTGGCAAATCAACAGTTGAAAGGAATTGAGGAAGGTTATCTAAA
Form-2D-Set-8	AAGAGTATTAATTAATTTTCCCTTAGAATCCTTGAAAACATAGCGATAGC
Form-2D-Set-9	GCTTTGCCATATCTTTAGGAGCACTAACAACTAATAGATTAGAGCCGTCA
Form-2D-Set-10	GCCCTATATATATGTGAGTGAATAACCTTGCTTCTGTAAATCGTCGCTAT
Form-2D-Set-11	CATGTTTAATAGATAATACATTTGAGGATTTAGAAGTATTAGACTTTACA
Form-2D-Set-12	ATCATACGCATTTGAATTACCTTTTTTAATGGAAACAGTACATAAATCAA
Form-2D-Set-13	CCCACCGTGGTGCCGTAAAGCACTAAATCGGAACCCTAAAGGGAGCCCCC
Form-2D-Set-14	CTATTTTATGAATGGCTATTAGTCTTTAATGCGCGAACTGATAGCCCTAA
Form-2D-Set-15	GTCAGTGAGATTTAGAGCTTGACGGGGAAAGCCGGCGAACGTGGCGAGAA
Form-2D-Set-16	TACAGCACCTGACCTGAAAGCGTAAGAATACGTGGCACAGACAATATTTT
Form-2D-Set-17	GTTCTGGGAGGAAGGGAAGAAAGCGAAAGGAGCGGGCGCTAGGGCGCTGG
Form-2D-Set-18	CCTATCACAGTAATAAAAGGGACATTCTGGCCAACAGAGATAGAACCCTT
Form-2D-Set-19	ACTTGTGGCCGAGATAGGGTTGAGTGTTGTTCCAGTTTGGAACAAGAGTC
Form-2D-Set-20	AAGTGCTACACTATTAAAGAACGTGGACTCCAACGTCAAAGGGCGAAAAA
Form-2D-Set-21	TCCCTAAACCGTCTATCAGGGCGATGGCCCACTACGTGAACCATCACCCA
Form-2D-Set-22	GCGACCCCTAATGAATCGGCCAACGCGCGGGGAGAGGCGGTTTGCGTATT
Form-2D-Set-23	AACTACATCTGCCCGCTTTCCAGTCGGGAAACCTGTCGTGCCAGCTGCAT
Form-2D-Set-24	TCGCCGAAGCCTAATGAGTGAGCTAACTCACATTAATTGCGTTGCGCTCA
Form-2D-Set-25	TGTAGTACTGTCAATCATATGTACCCCGGTTGATAATCAGAAAAGCCCCA
Form-2D-Set-26	TTATTCAGTGTTATCCGCTCACAATTCCACACAACATACGAGCCGGAAGC
Form-2D-Set-27	GGTTTACTAAAACAGGAAGATTGTATAAGCAAATATTTAAATTGTAAACG
Form-2D-Set-28	CACACGTTTCGAATTCGTAATCATGGTCATAGCTGTTTCCTGTGTGAAAT
Form-2D-Set-29	CAAAATACTTAATATTTTGTTAAAATTCGCATTAAATTTTTGTTAAATCA

Form-2D-Set-30	CCCCAATCCATGCCTGCAGGTCGACTCTAGAGGATCCCCGGGTACCGAGC
Form-2D-Set-31	CCTTACCAGGTCATTGCCTGAGAGTCTGGAGCAAACAAGAGAATCGATGA
Form-2D-Set-32	AAGTGCTTCGGAGAGGGTAGCTATTTTTGAGAGATCTACAAAGGCTATCA
Form-2D-Set-33	CAAGGTTCCATCAATATGATATTCAACCGTTCTAGCTGATAAATTAATGC
Form-2D-Set-34	ACTTCTGGTCTGGAAGTTTCATTCCATATAACAGTTGATTCCCAATTCTG
Form-2D-Set-35	ACTTGGTTCTGTAGCTCAACATGTTTTAAATATGCAACTAAAGTACGGTG
Form-2D-Set-36	CGGGAATCCATTTTTGCGGATGGCTTAGAGCTTAATTGCTGAATATAATG
Form-2D-Set-37	AACGAAGAGCGGATTGCATCAAAAAGATTAAGAGGAAGCCCGAAAGACTT
Form-2D-Set-38	ATCTTTGTAGAACGAGTAGTAAATTGGGCTTGAGATGGTTTAATTTCAAC
Form-2D-Set-39	CATATACGCAAATATCGCGTTTTAATTCGAGCTTCAAAGCGAACCAGACC
Form-2D-Set-40	TCAGGTTACTGCTCATTCAGTGAATAAGGCTTGCCCTGACGAGAAACACC
Form-2D-Set-41	TCTTTGTTGGAAGCAAACTCCAACAGGTCAGGATTAGAGAGTACCTTTAA
Form-2D-Set-42	CATTTATCGACAAGAACCGGATATTCATTACCCAAATCAACGTAACAAAG
Form-2D-Set-43	TAACGAACTCCGCGACCTGCTCCATGTTACTTAGCCGGAACGAGGCGCAG
Form-2D-Set-44	CTCGTCGCTTTCAGCGGAGTGAGAATAGAAAGGAACAACTAAAGGAATTG
Form-2D-Set-45	CATAAAAGACGGTCAATCATAAGGGAACCGAACTGACCAACTTTGAAAGA
Form-2D-Set-46	CGGAACTGATGAATTTTCTGTATGGGATTTTGCTAAACAACTTTCAACAG
Form-2D-Set-47	TACTTAATGGACAGATGAACGGTGTACAGACCAGGCGCATAGGCTGGCTG
Form-2D-Set-48	TTCTACTCCGTAACGATCTAAAGTTTTGTCGTCTTTCCAGACGTTAGTAA
Form-2D-Set-49	GTGAGTTCCACCCTCAGAACCGCCACCCTCAGAGCCACCACCCTCATTTT
Form-2D-Set-50	ACCAAGCAAGAGCCGCCACCCTCAGAACCGCCACCCTCAGAGCCACCACC
Form-2D-Set-51	GTGACCGGCAGGGATAGCAAGCCCAATAGGAACCCATGTACCGTAACACT
Form-2D-Set-52	ATCATACTAAAATCACCGGAACCAGAGCCACCACCGGAACCGCCTCCCTC
Form-2D-Set-53	TACAAATTGAGTTTCGTCACCAGTACAAACTACAACGCCTGTAGCATTCC
Form-2D-Set-54	GAGTGTATGGTCATAGCCCCCTTATTAGCGTTTGCCATCTTTTCATAATC
Form-2D-Set-55	CGTAAAGTGCCAGCAAAATCACCAGTAGCACCATTACCATTAGCAAGGCC
Form-2D-Set-56	AACACCTGGGAAACGTCACCAATGAAACCATCGATAGCAGCACCGTAATC
Form-2D-Set-57	CTATGTTCAGTAGCGACAGAATCAAGTTTGCCTTTAGCGTCAGACTGTAG
Form-2D-Set-58	TCACGCGTCCGAGGAAACGCAATAATAACGGAATACCCAAAAGAACTGGC
Form-2D-Set-59	TTTAGGGGATGATTAAGACTCCTTATTACGCAGTATGTTAGCAAACGTAG
Form-2D-Set-60	ATAGTAGCAAAATACATACATAAAGGTGGCAACATATAAAAGAAACGCAA
Form-2D-Set-61	GTCACTAAAACATGTTCAGCTAATGCAGAACGCGCCTGTTTATCAACAAT
Form-2D-Set-62	CATTCTGTATCTTACCGAAGCCCTTTTTAAGAAAAGTAAGCAGATAGCCG
Form-2D-Set-63	CTCTTGTTAGATAAGTCCTGAACAAGAAAAATAATATCCCATCCTAATTT
Form-2D-Set-64	AGTGTCTGTGAGCGCTAATATCAGAGAGATAACCCACAAGAATTGAGTTA
Form-2D-Set-65	TGTGGTTAACGAGCATGTAGAAACCAATCAATAATCGGCTGTCTTTCCTT
Form-2D-Set-66	GCTGACAGAGCCCAATAATAAGAGCAAGAAACAATGAAATAGCAATAGCT

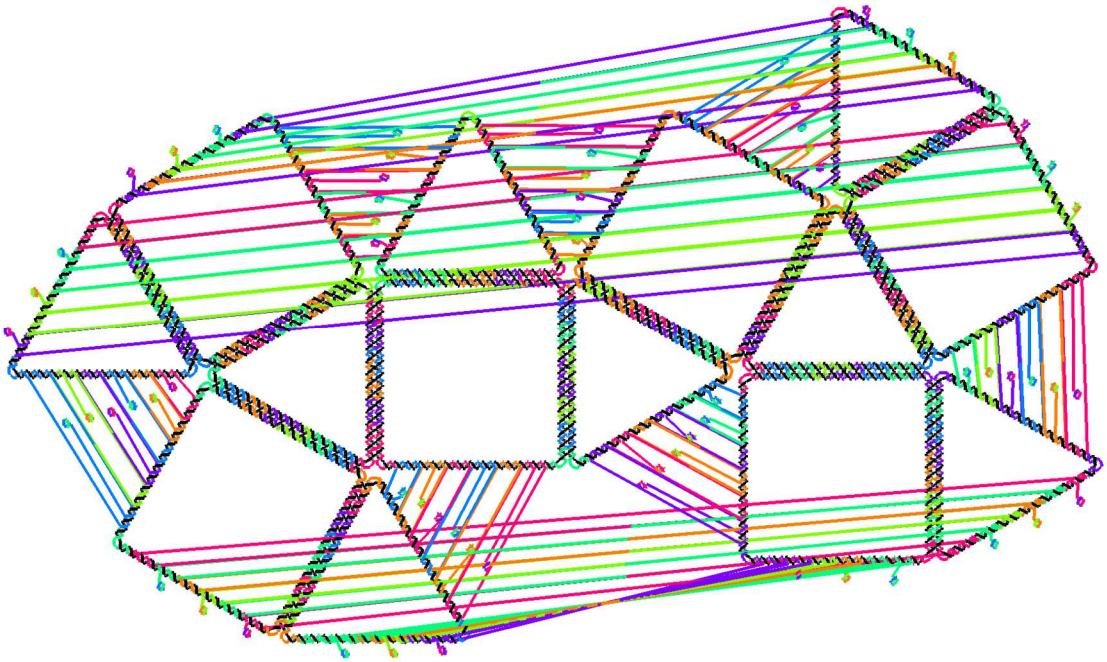


Fig. S79. Design pattern for the 3D closed structure of the cuboctahedron

Table S16. Sequences of the set strands for the 3D closed structure of the cuboctahedron

Number	Sequence
Form-3D-Set-1	GAACGTGGACTGTTGGGTTATATAACTATATGCACTATTAAAGCTAAACT
Form-3D-Set-2	GTTGAGTGTTGTCGCAAGACAAAGAACGCGAGCCGAGATAGGTCTGAATC
Form-3D-Set-3	AGCACTAAATCTGCGCGAACTGATAGCCCTAAGGTGCCGTAATGTAGAGG
Form-3D-Set-4	TAGTCTTTAAGGAACCCTAAAGGGAGCCCCCTGAATGGCTATTTCAGGGC
Form-3D-Set-5	TTGACGGGGAAACGTGGCACAGACAATATTTTGATTTAGAGCGTCCGCAC
Form-3D-Set-6	GAAAGCGAAAGGCCAACAGAGATAGAACCCTTAGGAAGGGAACAAAGAAA
Form-3D-Set-7	GCGTAAGAATAGCCGGCGAACGTGGCGAGAACTGACCTGAAAACAGCCTC
Form-3D-Set-8	GGACATTCTGGAGCGGGCGCTAGGGCGCTGGAGTAATAAAAGTCGTCAGT
Form-3D-Set-9	TGAGCTAACTCGACAAAAGGTAAAGTAATTCTGCCTAATGAGTGACATTA
Form-3D-Set-10	TCCAGTCGGGACAGAGGCATTTTCGAGCCAGTCTGCCCGCTTCCAAAGGT
Form-3D-Set-11	GCCAACGCGCGTTGAGAATCGCCATATTTAACTAATGAATCGCCCGCAGA
Form-3D-Set-12	ATGTACCCCGGCACAACATACGAGCCGGAAGCTGTCAATCATCCTGCATG
Form-3D-Set-13	CACAATTCCATTGATAATCAGAAAAGCCCCATGTTATCCGCTGACATTTA
Form-3D-Set-14	GATTGTATAAGTAGCTGTTTCCTGTGTGAAATAAAACAGGAACACGAAGT
Form-3D-Set-15	ATCATGGTCACAAATATTTAAATTGTAAACGTCGAATTCGTAAATATGCA
Form-3D-Set-16	GTTAAAATTCGGAGGATCCCCGGGTACCGAGCTTAATATTTTTTGCTGTT
Form-3D-Set-17	GTCGACTCTACATTAAATTTTTGTTAAATCACATGCCTGCAGCCTAAAAT

Form-3D-Set-18	TATTCAACCGCAACATATAAAAGAAACGCAACATCAATATGATCTGTACA
Form-3D-Set-19	GAGAGTCTGGGGAATACCCAAAAGAACTGGCGGTCATTGCCTTCGTTTGT
Form-3D-Set-20	GCTATTTTTGGCAGTATGTTAGCAAACGTAGCGGAGAGGGTACAGATGAG
Form-3D-Set-21	ATGGCTTAGAGCCTTTAGCGTCAGACTGTAGCATTTTTGCGGTTTCGGAC
Form-3D-Set-22	CATTCCATATACCATTACCATTAGCAAGGCCTCTGGAAGTTTCATCCACA
Form-3D-Set-23	CATGTTTTAAATCGATAGCAGCACCGTAATCCTGTAGCTCAATTAGTACA
Form-3D-Set-24	CAAAAAGATTTTGAGATGGTTTAATTTCAACGCGGATTGCATTGCGCTTA
Form-3D-Set-25	GTAAATTGGGCAAGAGGAAGCCCGAAAGACTTAGAACGAGTATTACCCCA
Form-3D-Set-26	TTTTAATTCGCTTGCCCTGACGAGAAACACCCAAATATCGCGCAAAGGCG
Form-3D-Set-27	AGTGAATAAGGAGCTTCAAAGCGAACCAGACCCTGCTCATTCTGCTCAGT
Form-3D-Set-28	CCAACAGGTCACCCAAATCAACGTAACAAAGGGAAGCAAACTTTGTACTT
Form-3D-Set-29	GGATATTCATTAGGATTAGAGAGTACCTTTAAGACAAGAACCACGAATAC
Form-3D-Set-30	TAAGGGAACCTTGCTAAACAACTTTCAACAGACGGTCAATCACCAGGGGC
Form-3D-Set-31	TGTATGGGATTGAACTGACCAACTTTGAAAGAATGAATTTTCCCGTAGAG
Form-3D-Set-32	CGGTGTACAGCGTCTTTCCAGACGTTAGTAAGGACAGATGAAGCTCCTGT
Form-3D-Set-33	TAAAGTTTTGTACCAGGCGCATAGGCTGGCTGCGTAACGATCACGACGAG
Form-3D-Set-34	CTCCATGTTAAAGGAACAACTAAAGGAATTGTCCGCGACCTGGGTGAGCA
Form-3D-Set-35	GTGAGAATAGACTTAGCCGGAACGAGGCGCAGTTTCAGCGGAGGTACTCA
Form-3D-Set-36	CCGCCACCCTGCCACCCTCAGAGCCACCACCCACCCTCAGAATCTACCGC
Form-3D-Set-37	CCCTCAGAACCCAGAGCCACCACCCTCATTTTAGAGCCGCCAAGATCCCC
Form-3D-Set-38	AGCCCAATAGACCACCGGAACCGCCTCCCTCCAGGGATAGCAAGAATCCC
Form-3D-Set-39	GAACCAGAGCCGAACCCATGTACCGTAACACTAAAATCACCGCGAAAAAC
Form-3D-Set-40	CCAGTACAAAGTTTGCCATCTTTTCATAATCGAGTTTCGTCATCGCAAAA
Form-3D-Set-41	CCCTTATTAGCCTACAACGCCTGTAGCATTCCGGTCATAGCCCGAAAGAA
Form-3D-Set-42	TCACCAGTAGCAACAGTTGATTCCCAATTCTGGCCAGCAAAAGGCAAAAT
Form-3D-Set-43	CCAATGAAACCATATGCAACTAAAGTACGGTGGGAAACGTCACGCCACGT
Form-3D-Set-44	GAATCAAGTTTGCTTAATTGCTGAATATAATGAGTAGCGACAATTAGCAC
Form-3D-Set-45	GCAATAATAACAGCAAACAAGAGAATCGATGACCGAGGAAACCCCTCTAT
Form-3D-Set-46	CTCCTTATTACAGAGATCTACAAAGGCTATCAATGATTAAGAGCCCCTTG
Form-3D-Set-47	CATAAAGGTGGTTCTAGCTGATAAATTAATGCAAAATACATACGAATGTA
Form-3D-Set-48	ATCAGAGAGAAATAATCGGCTGTCTTTCCTTTGAGCGCTAATTTGTATTC
Form-3D-Set-49	AGAAACCAATCTAACCCACAAGAATTGAGTTAACGAGCATGTAGCTCTTG
Form-3D-Set-50	AAGAGCAAGAAATAATATCCCATCCTAATTTAGCCCAATAATTGACTCGA
Form-3D-Set-51	TGAACAAGAAAAACAATGAAATAGCAATAGCTAGATAAGTCCAATAGAGA
Form-3D-Set-52	GCCCTTTTTAACGCGCCTGTTTATCAACAATATCTTACCGAACCAAGTAC
Form-3D-Set-53	GCTAATGCAGAAGAAAAGTAAGCAGATAGCCGAACATGTTCACCGTGACT
Form-3D-Set-54	TAGGGCTTAAGGGAGAGGCGGTTTGCGTATTACGCTCAACAGACGTGTTT
Form-3D-Set-55	GTAATTTAGGAACCTGTCGTGCCAGCTGCATAACGCCAACATACGCAACT
Form-3D-Set-56	ATAAAGTACCACATTAATTGCGTTGCGCTCAAATAAGAGAATGTGATTGC
Form-3D-Set-57	AATTTATCAACACTACGTGAACCATCACCCAAGTCAATAGTGTAGGCGGC

Form-3D-Set-58	TCCGGCTTAGCCAACGTCAAAGGGCGAAAAACCTTTTTAACCTAAGTCGT
Form-3D-Set-59	GCAAATCCAATTCCAGTTTGGAACAAGAGTCTAAATGCTGATTTCAAGCC
Form-3D-Set-60	CAGTTGAAAGGCCTTGAAAACATAGCGATAGCGGCAAATCAAGGCAAAGC
Form-3D-Set-61	CCCTTAGAATAATTGAGGAAGGTTATCTAAATAATTAATTTTCGAGTGGA
Form-3D-Set-62	GAGCACTAACAGCTTCTGTAAATCGTCGCTATATATCTTTAGGAGCAGGG
Form-3D-Set-63	GAATAACCTTACTAATAGATTAGAGCCGTCATATATGTGAGTTACATACG
Form-3D-Set-64	CATTTGAGGATTGGAAACAGTACATAAATCAAATAGATAATATAATATCT
Form-3D-Set-65	CCTTTTTTAATTAGAAGTATTAGACTTTACACATTTGAATTAGTTCGAGG
Form-3D-Set-66	GGGCGATGGCCAATCATAGGTCTGAGAGACTACCGTCTATCAGGCAGGTG

Table S17. Sequences of the fuel strands for reconfiguration from 2D to 3D

Number	Sequence
2D-to-3D-Fuel-1	AGAATTACTTTACCTTTTGTCGGTACTTTATATTCTCTTATTATGTCAGA
2D-to-3D-Fuel-2	ACTGGCTCGAAAATGCCTCTGCCTAAATTACATGTTGGCGTTCGACTAGC
2D-to-3D-Fuel-3	GTTAAATATGGCGATTCTCAATTAAGCCCTACTGTTGAGCGTCTAGAATT
2D-to-3D-Fuel-4	CTCGCGTTCTTTGTCTTGCGATTGGATTTGCATCAGCATTTAAACGACTC
2D-to-3D-Fuel-5	CATATAGTTATATAACCCAACCTAAGCCGGAGGTTAAAAAGGAGTCATGA
2D-to-3D-Fuel-6	TAGTCTCTCAGACCTATGATTTTGATAAATTCACTATTGACTGTTGATAG
2D-to-3D-Fuel-7	TTTAGATAACCTTCCTCAATTCCTTTCAACTGTTGATTTGCCAAAATGAG
2D-to-3D-Fuel-8	GCTATCGCTATGTTTTCAAGGATTCTAAGGGAAAATTAATTAATACTCTT
2D-to-3D-Fuel-9	TGACGGCTCTAATCTATTAGTTGTTAGTGCTCCTAAAGATATGGCAAAGC
2D-to-3D-Fuel-10	ATAGCGACGATTTACAGAAGCAAGGTTATTCACTCACATATATATAGGGC
2D-to-3D-Fuel-11	TGTAAAGTCTAATACTTCTAAATCCTCAAATGTATTATCTATTAAACATG
2D-to-3D-Fuel-12	TTGATTTATGTACTGTTTCCATTAAAAAAGGTAATTCAAATGCGTATGAT
2D-to-3D-Fuel-13	GGGGGCTCCCTTTAGGGTTCCGATTTAGTGCTTTACGGCACCACGGTGGG
2D-to-3D-Fuel-14	TTAGGGCTATCAGTTCGCGCATTAAAGACTAATAGCCATTCATAAAATAG
2D-to-3D-Fuel-15	TTCTCGCCACGTTCGCCGGCTTTCCCCGTCAAGCTCTAAATCTCACTGAC
2D-to-3D-Fuel-16	AAAATATTGTCTGTGCCACGTATTCTTACGCTTTCAGGTCAGGTGCTGTA
2D-to-3D-Fuel-17	CCAGCGCCCTAGCGCCCGCTCCTTTCGCTTTCTTCCCTTCCTCCCAGAAC
2D-to-3D-Fuel-18	AAGGGTTCTATCTCTGTTGGCCAGAATGTCCCTTTTATTACTGTGATAGG
2D-to-3D-Fuel-19	GACTCTTGTTCCAAACTGGAACAACACTCAACCCTATCTCGGCCACAAGT
2D-to-3D-Fuel-20	TTTTTCGCCCTTTGACGTTGGAGTCCACGTTCTTTAATAGTGTAGCACTT
2D-to-3D-Fuel-21	TGGGTGATGGTTCACGTAGTGGGCCATCGCCCTGATAGACGGTTTAGGGA
2D-to-3D-Fuel-22	AATACGCAAACCGCCTCTCCCCGCGCGTTGGCCGATTCATTAGGGGTCGC
2D-to-3D-Fuel-23	ATGCAGCTGGCACGACAGGTTTCCCGACTGGAAAGCGGGCAGATGTAGTT
2D-to-3D-Fuel-24	TGAGCGCAACGCAATTAATGTGAGTTAGCTCACTCATTAGGCTTCGGCGA
2D-to-3D-Fuel-25	TGGGGCTTTTCTGATTATCAACCGGGGTACATATGATTGACAGTACTACA
2D-to-3D-Fuel-26	GCTTCCGGCTCGTATGTTGTGTGGAATTGTGAGCGGATAACACTGAATAA
2D-to-3D-Fuel-27	CGTTTACAATTTAAATATTTGCTTATACAATCTTCCTGTTTTAGTAAACC
2D-to-3D-Fuel-28	ATTTCACACAGGAAACAGCTATGACCATGATTACGAATTCGAAACGTGTG

2D-to-3D-Fuel-29	TGATTTAACAAAAATTTAATGCGAATTTTAACAAAATATTAAGTATTTTG
2D-to-3D-Fuel-30	GCTCGGTACCCGGGGATCCTCTAGAGTCGACCTGCAGGCATGGATTGGGG
2D-to-3D-Fuel-31	TCATCGATTCTCTTGTTTGCTCCAGACTCTCAGGCAATGACCTGGTAAGG
2D-to-3D-Fuel-32	TGATAGCCTTTGTAGATCTCTCAAAAATAGCTACCCTCTCCGAAGCACTT
2D-to-3D-Fuel-33	GCATTAATTTATCAGCTAGAACGGTTGAATATCATATTGATGGAACCTTG
2D-to-3D-Fuel-34	CAGAATTGGGAATCAACTGTTATATGGAATGAAACTTCCAGACCAGAAGT
2D-to-3D-Fuel-35	CACCGTACTTTAGTTGCATATTTAAAACATGTTGAGCTACAGAACCAAGT
2D-to-3D-Fuel-36	CATTATATTCAGCAATTAAGCTCTAAGCCATCCGCAAAAATGGATTCCCG
2D-to-3D-Fuel-37	AAGTCTTTCGGGCTTCCTCTTAATCTTTTTGATGCAATCCGCTCTTCGTT
2D-to-3D-Fuel-38	GTTGAAATTAAACCATCTCAAGCCCAATTTACTACTCGTTCTACAAAGAT
2D-to-3D-Fuel-39	GGTCTGGTTCGCTTTGAAGCTCGAATTAAAACGCGATATTTGCGTATATG
2D-to-3D-Fuel-40	GGTGTTTCTCGTCAGGGCAAGCCTTATTCACTGAATGAGCAGTAACCTGA
2D-to-3D-Fuel-41	TTAAAGGTACTCTCTAATCCTGACCTGTTGGAGTTTGCTTCCAACAAAGA
2D-to-3D-Fuel-42	CTTTGTTACGTTGATTTGGGTAATGAATATCCGGTTCTTGTCGATAAATG
2D-to-3D-Fuel-43	CTGCGCCTCGTTCCGGCTAAGTAACATGGAGCAGGTCGCGGAGTTCGTTA
2D-to-3D-Fuel-44	CAATTCCTTTAGTTGTTCCTTTCTATTCTCACTCCGCTGAAAGCGACGAG
2D-to-3D-Fuel-45	TCTTTCAAAGTTGGTCAGTTCGGTTCCCTTATGATTGACCGTCTTTTATG
2D-to-3D-Fuel-46	CTGTTGAAAGTTGTTTAGCAAAATCCCATACAGAAAATTCATCAGTTCCG
2D-to-3D-Fuel-47	CAGCCAGCCTATGCGCCTGGTCTGTACACCGTTCATCTGTCCATTAAGTA
2D-to-3D-Fuel-48	TTACTAACGTCTGGAAAGACGACAAAACTTTAGATCGTTACGGAGTAGAA
2D-to-3D-Fuel-49	AAAATGAGGGTGGTGGCTCTGAGGGTGGCGGTTCTGAGGGTGGAACTCAC
2D-to-3D-Fuel-50	GGTGGTGGCTCTGAGGGTGGCGGTTCTGAGGGTGGCGGCTCTTGCTTGGT
2D-to-3D-Fuel-51	AGTGTTACGGTACATGGGTTCCTATTGGGCTTGCTATCCCTGCCGGTCAC
2D-to-3D-Fuel-52	GAGGGAGGCGGTTCCGGTGGTGGCTCTGGTTCCGGTGATTTTAGTATGAT
2D-to-3D-Fuel-53	GGAATGCTACAGGCGTTGTAGTTTGTACTGGTGACGAAACTCAATTTGTA
2D-to-3D-Fuel-54	GATTATGAAAAGATGGCAAACGCTAATAAGGGGGCTATGACCATACACTC
2D-to-3D-Fuel-55	GGCCTTGCTAATGGTAATGGTGCTACTGGTGATTTTGCTGGCACTTTACG
2D-to-3D-Fuel-56	GATTACGGTGCTGCTATCGATGGTTTCATTGGTGACGTTTCCCAGGTGTT
2D-to-3D-Fuel-57	CTACAGTCTGACGCTAAAGGCAAACTTGATTCTGTCGCTACTGAACATAG
2D-to-3D-Fuel-58	GCCAGTTCTTTTGGGTATTCCGTTATTATTGCGTTTCCTCGGACGCGTGA
2D-to-3D-Fuel-59	CTACGTTTGCTAACATACTGCGTAATAAGGAGTCTTAATCATCCCCTAAA
2D-to-3D-Fuel-60	TTGCGTTTCTTTTATATGTTGCCACCTTTATGTATGTATTTTGCTACTAT
2D-to-3D-Fuel-61	ATTGTTGATAAACAGGCGCGTTCTGCATTAGCTGAACATGTTTTAGTGAC
2D-to-3D-Fuel-62	CGGCTATCTGCTTACTTTTCTTAAAAAGGGCTTCGGTAAGATACAGAATG
2D-to-3D-Fuel-63	AAATTAGGATGGGATATTATTTTTCTTGTTCAGGACTTATCTAACAAGAG
2D-to-3D-Fuel-64	TAACTCAATTCTTGTGGGTTATCTCTCTGATATTAGCGCTCACAGACACT
2D-to-3D-Fuel-65	AAGGAAAGACAGCCGATTATTGATTGGTTTCTACATGCTCGTTAACCACA
2D-to-3D-Fuel-66	AGCTATTGCTATTTCATTGTTTCTTGCTCTTATTATTGGGCTCTGTCAGC

Table S18. Sequences of the fuel strands for reconfiguration from 3D to 2D

Number	Sequence
3D-to-2D-Fuel-1	AGTTTAGCTTTAATAGTGCATATAGTTATATAACCCAACAGTCCACGTTC
3D-to-2D-Fuel-2	GATTCAGACCTATCTCGGCTCGCGTTCTTTGTCTTGCGACAACACTCAAC
3D-to-2D-Fuel-3	CCTCTACATTACGGCACCTTAGGGCTATCAGTTCGCGCAGATTTAGTGCT
3D-to-2D-Fuel-4	GCCCTGAAATAGCCATTCAGGGGGCTCCCTTTAGGGTTCCTTAAAGACTA
3D-to-2D-Fuel-5	GTGCGGACGCTCTAAATCAAAATATTGTCTGTGCCACGTTTCCCCGTCAA
3D-to-2D-Fuel-6	TTTCTTTGTTCCCTTCCTAAGGGTTCTATCTCTGTTGGCCTTTCGCTTTC
3D-to-2D-Fuel-7	GAGGCTGTTTTCAGGTCAGTTCTCGCCACGTTCGCCGGCTATTCTTACGC
3D-to-2D-Fuel-8	ACTGACGACTTTTATTACTCCAGCGCCCTAGCGCCCGCTCCAGAATGTCC
3D-to-2D-Fuel-9	TAATGTCACTCATTAGGCAGAATTACTTTACCTTTTGTCGAGTTAGCTCA
3D-to-2D-Fuel-10	ACCTTTGGAAGCGGGCAGACTGGCTCGAAAATGCCTCTGTCCCGACTGGA
3D-to-2D-Fuel-11	TCTGCGGGCGATTCATTAGTTAAATATGGCGATTCTCAACGCGCGTTGGC
3D-to-2D-Fuel-12	CATGCAGGATGATTGACAGCTTCCGGCTCGTATGTTGTGCCGGGGTACAT
3D-to-2D-Fuel-13	TAAATGTCAGCGGATAACATGGGGCTTTTCTGATTATCAATGGAATTGTG
3D-to-2D-Fuel-14	ACTTCGTGTTCCTGTTTTATTTCACACAGGAAACAGCTACTTATACAATC
3D-to-2D-Fuel-15	TGCATATTTACGAATTCGACGTTTACAATTTAAATATTTGTGACCATGAT
3D-to-2D-Fuel-16	AACAGCAAAAAATATTAAGCTCGGTACCCGGGGATCCTCCGAATTTTAAC
3D-to-2D-Fuel-17	ATTTTAGGCTGCAGGCATGTGATTTAACAAAAATTTAATGTAGAGTCGAC
3D-to-2D-Fuel-18	TGTACAGATCATATTGATGTTGCGTTTCTTTTATATGTTGCGGTTGAATA
3D-to-2D-Fuel-19	ACAAACGAAGGCAATGACCGCCAGTTCTTTTGGGTATTCCCCAGACTCTC
3D-to-2D-Fuel-20	CTCATCTGTACCCTCTCCGCTACGTTTGCTAACATACTGCCAAAAATAGC
3D-to-2D-Fuel-21	GTCCGAAACCGCAAAAATGCTACAGTCTGACGCTAAAGGCTCTAAGCCAT
3D-to-2D-Fuel-22	TGTGGATGAAACTTCCAGAGGCCTTGCTAATGGTAATGGTATATGGAATG
3D-to-2D-Fuel-23	TGTACTAATTGAGCTACAGGATTACGGTGCTGCTATCGATTTAAAACATG
3D-to-2D-Fuel-24	TAAGCGCAATGCAATCCGCGTTGAAATTAAACCATCTCAAAATCTTTTTG
3D-to-2D-Fuel-25	TGGGGTAATACTCGTTCTAAGTCTTTCGGGCTTCCTCTTGCCCAATTTAC
3D-to-2D-Fuel-26	CGCCTTTGCGCGATATTTGGGTGTTTCTCGTCAGGGCAAGCGAATTAAAA
3D-to-2D-Fuel-27	ACTGAGCAGAATGAGCAGGGTCTGGTTCGCTTTGAAGCTCCTTATTCACT
3D-to-2D-Fuel-28	AAGTACAAAGTTTGCTTCCCTTTGTTACGTTGATTTGGGTGACCTGTTGG
3D-to-2D-Fuel-29	GTATTCGTGGTTCTTGTCTTAAAGGTACTCTCTAATCCTAATGAATATCC
3D-to-2D-Fuel-30	GCCCCTGGTGATTGACCGTCTGTTGAAAGTTGTTTAGCAAGGTTCCCTTA
3D-to-2D-Fuel-31	CTCTACGGGAAAATTCATTCTTTCAAAGTTGGTCAGTTCAATCCCATACA
3D-to-2D-Fuel-32	ACAGGAGCTTCATCTGTCCTTACTAACGTCTGGAAAGACGCTGTACACCG
3D-to-2D-Fuel-33	CTCGTCGTGATCGTTACGCAGCCAGCCTATGCGCCTGGTACAAAACTTTA
3D-to-2D-Fuel-34	TGCTCACCCAGGTCGCGGACAATTCCTTTAGTTGTTCCTTTAACATGGAG
3D-to-2D-Fuel-35	TGAGTACCTCCGCTGAAACTGCGCCTCGTTCCGGCTAAGTCTATTCTCAC
3D-to-2D-Fuel-36	GCGGTAGATTCTGAGGGTGGGTGGTGGCTCTGAGGGTGGCAGGGTGGCGG
3D-to-2D-Fuel-37	GGGGATCTTGGCGGCTCTAAAATGAGGGTGGTGGCTCTGGGTTCTGAGGG
3D-to-2D-Fuel-38	GGGATTCTTGCTATCCCTGGAGGGAGGCGGTTCCGGTGGTCTATTGGGCT
3D-to-2D-Fuel-39	GTTTTTCGCGGTGATTTTAGTGTTACGGTACATGGGTTCGGCTCTGGTTC

3D-to-2D-Fuel-40	TTTTGCGATGACGAAACTCGATTATGAAAAGATGGCAAACTTTGTACTGG
3D-to-2D-Fuel-41	TTCTTTCGGGCTATGACCGGAATGCTACAGGCGTTGTAGGCTAATAAGGG
3D-to-2D-Fuel-42	ATTTTGCCTTTTGCTGGCCAGAATTGGGAATCAACTGTTGCTACTGGTGA
3D-to-2D-Fuel-43	ACGTGGCGTGACGTTTCCCACCGTACTTTAGTTGCATATGGTTTCATTGG
3D-to-2D-Fuel-44	GTGCTAATTGTCGCTACTCATTATATTCAGCAATTAAGCAAACTTGATTC
3D-to-2D-Fuel-45	ATAGAGGGGTTTCCTCGGTCATCGATTCTCTTGTTTGCTGTTATTATTGC
3D-to-2D-Fuel-46	CAAGGGGCTCTTAATCATTGATAGCCTTTGTAGATCTCTGTAATAAGGAG
3D-to-2D-Fuel-47	TACATTCGTATGTATTTTGCATTAATTTATCAGCTAGAACCACCTTTATG
3D-to-2D-Fuel-48	GAATACAAATTAGCGCTCAAAGGAAAGACAGCCGATTATTTCTCTCTGAT
3D-to-2D-Fuel-49	CAAGAGCTACATGCTCGTTAACTCAATTCTTGTGGGTTAGATTGGTTTCT
3D-to-2D-Fuel-50	TCGAGTCAATTATTGGGCTAAATTAGGATGGGATATTATTTCTTGCTCTT
3D-to-2D-Fuel-51	TCTCTATTGGACTTATCTAGCTATTGCTATTTCATTGTTTTTCTTGTTCA
3D-to-2D-Fuel-52	GTACTTGGTTCGGTAAGATATTGTTGATAAACAGGCGCGTTAAAAAGGGC
3D-to-2D-Fuel-53	AGTCACGGTGAACATGTTCGGCTATCTGCTTACTTTTCTTCTGCATTAGC
3D-to-2D-Fuel-54	AAACACGTCTGTTGAGCGTAATACGCAAACCGCCTCTCCCTTAAGCCCTA
3D-to-2D-Fuel-55	AGTTGCGTATGTTGGCGTTATGCAGCTGGCACGACAGGTTCCTAAATTAC
3D-to-2D-Fuel-56	GCAATCACATTCTCTTATTTGAGCGCAACGCAATTAATGTGGTACTTTAT
3D-to-2D-Fuel-57	GCCGCCTACACTATTGACTTGGGTGATGGTTCACGTAGTGTTGATAAATT
3D-to-2D-Fuel-58	ACGACTTAGGTTAAAAAGGTTTTTCGCCCTTTGACGTTGGCTAAGCCGGA
3D-to-2D-Fuel-59	GGCTTGAAATCAGCATTTAGACTCTTGTTCCAAACTGGAATTGGATTTGC
3D-to-2D-Fuel-60	GCTTTGCCTTGATTTGCCGCTATCGCTATGTTTTCAAGGCCTTTCAACTG
3D-to-2D-Fuel-61	TCCACTCGAAAATTAATTATTTAGATAACCTTCCTCAATTATTCTAAGGG
3D-to-2D-Fuel-62	CCCTGCTCCTAAAGATATATAGCGACGATTTACAGAAGCTGTTAGTGCTC
3D-to-2D-Fuel-63	CGTATGTAACTCACATATATGACGGCTCTAATCTATTAGTAAGGTTATTC
3D-to-2D-Fuel-64	AGATATTATATTATCTATTTGATTTATGTACTGTTTCCAATCCTCAAATG
3D-to-2D-Fuel-65	CCTCGAACTAATTCAAATGTGTAAAGTCTAATACTTCTAATTAAAAAAGG
3D-to-2D-Fuel-66	CACCTGCCTGATAGACGGTAGTCTCTCAGACCTATGATTGGCCATCGCCC

Table S19. Sequences of the snub cube.

Number	Sequence
1--snub--	AGACTTTTTTTCAAATATCGCGGAAGCAAACTTTTTTCCAAC
2--snub--	AGGTCGAACCATCACCTTTTCAAATCAAGTTCGGA
3--snub--	ACGTCATTTTAAGGGCGAAATGATAAGAGGTTTTTTCATTTTTGCGCTGTA
4--snub--	CCCACTACGTAGGATTAGAGAGTACCTTTAACAGGGCGATGG
5--snub--	TTGCTCCTTTAACCGTCTAT
6--snub--	TGATTCCATTAGATACTTTTTATTTCGCAAACTCCA
7--snub--	AGTGTTTTTTGTTCCAGTTTAGTGCCAAGCTTTTTTTGCAT
8--snub--	AGAACGTGGATGGTCAATAACGACGTTGTAATCCACTATTAA
9--snub--	AACGACGGCCGGAACAAGAG
10--snub--	AATTCCTTTTTACACAACATATGCCTAATGAGTTTTTTGAGCTAACTGGTTG

11--snub--	CGAAATTTTTCGGCAAAATCTTCCAGTCGGGTTTTTAAACC
12--snub--	CCCGAGATAGCACATTAATTGCGTTGCGCTCCAAAAGAATAG
13--snub--	ACTGCCCGCTCCTTATAAAT
14--snub--	ACCCTATTTTTAAGGGAGCCCGACTATTATAGTTTTTTCAGAAGCAACCGAA
15--snub--	GGCGAAAGCCGTAAAGCACTAAATTTTTGGG
16--snub--	GGGCAATTTTTCAGCTGATTGCAAGCGGTCCATTTTTCGCTGGTTTGGGTTC
17--snub--	GTCGAGGTATCCTGTTTGATGGTCCCCAGCA
18--snub--	TGGATTTTTTTATTTACATTGAAAGGGACATTTTTTTCTGGCCAACAAGAAT
19--snub--	ACGTGGTTTTTCACAGACAATTTACCTTTTTTTTTTTAATGG
20--snub--	TGCGAACGTGACCTGAAAGCGTAGAGATAGA
21--snub--	GCTCAATTTTTCATGTTTTAACATTCCATATATTTTTACAGT
22--snub--	ACCCTTCAGTAGATTTAGTTTGACCCAATTC
23--snub--	AAAATACCGGTGTCTGGAAGTTTATATGCAA
24--snub--	AAACATTTTTTCAAGAAAACAGAACTGATAGCTTTTTCCTAAAACATCAGAA
25--snub--	GATAAATTTTTACAGAGGTGAATCGGGAGAAATTTTTCAATA
26--snub--	CTAAAGTACGAACGAACCACCAGCGCCATTA
27--snub--	AATCTAAATTGCTGAATATAATGGATGGCTT
28--snub--	AGAGCTTAAGCATCACCTTGCTGAAATGAAA
29--snub--	GCTGAGTTTTTAGCCAGCAGCAACCTCAAATATTTTTTCAAA
30--snub--	GCCTGGCTCGAATTCGTTTTTTAATCATGGTCTCAC
31--snub--	ACCAGTCGATCCCCGGGTACCGACAGGTCGA
32--snub--	CTCTAGAGACACGACCAGTAATAGCAGATTC
33--snub--	GAACAATGTGAAATTGTTATCCGCATAGCTG
34--snub--	CTATATTTTTTGTAAATGCTGCAAACTATCGGTTTTTCCTTGCTGGTTGCAA
35--snub--	CAGGAATTTTTAAACGCTCATACATAGCGATATTTTTGCTTA
36--snub--	TTTCCTGTATTACCGCCAGCCATAATATCCA
37--snub--	CCATCACAGTGTAAAGCCTGGGGCGAGCCGG
38--snub--	AGGCCATTTTTCCGAGTAAAATAGCAATACTTTTTTTCTTTG
39--snub--	AAGCATAAGCAAATTAACCGTTGGAGTCTGT
40--snub--	GTCTTTACCCTCCGATTTAGAG
41--snub--	AAAGCCGGCGGACCATAAATCAAAAATCAGCTTGACGGGG
42--snub--	TTCAGATTTTTAAACGAGAATAACGTGGCGAGTTTTTAAAGG
43--snub--	TACATTTAAGATTAAGAGGAAGCAGCGGATT
44--snub--	ATATAATTTTTTCCTGATTGTACTAATAGATTTTTTTAGAGCCGTCATTAGA
45--snub--	CTTTACTTTTTAAACAATTCGTTATTAATTTTTTTTTAAAAG
46--snub--	GCATCAAAGAGGATTTAGAAGTAATAGATAA
47--snub--	TCTGGTCCAAAGCGAACCAGACCGTTTTAAT
48--snub--	CCCTCAGTTGAAAGGATTTTTATTGAGGAAGAGAAC
49--snub--	TCGAGCTTAGTTGGCAAATCAACAATCAATA
50--snub--	TGTCGGGGGAGAGGCGTTTTTGTTTGCGTATGAGAC

51--snub--	TCCTGAGGAATCGGCCAACGCGCTGCCAGCT
52--snub--	TTGAAATTTTTTACCGACCGTGGATTTTAGACTTTTTAGGAACGGTACAGTG
53--snub--	GCATTAATAAGTGTTTTTATAATCGCCAGAA
54--snub--	TGCTTTGTTTTCTTTTCACCAGTTGGGCGCC
55--snub--	TTTAACTTTTTAACGCCAACAATGCGCCGCTATTTTTCAGGGCGCGTGTGCT
56--snub--	TTCCTCTTTTTGTTAGAATCAACCGGAATCATTTTTTAATTA
57--snub--	AGGGTGGTACGAGCACGTATAACACTATGGT
58--snub--	GAAAGGACCTGAGAGAGTTGCAGCCCTTCAC
59--snub--	AAGGGCTGGCAAGTGTTTTTTAGCGGTCACGTAATA
60--snub--	CGCCTGGCGCGGGCGCTAGGGCGAAGAAAGC
61--snub--	TTCATTTGAAATTTTTGAAT
62--snub--	TTTAATGCGCAAATTAATTACATTTAACAATGGCTATTAGTC
63--snub--	AAACATTGCTTCTGTATTTTTAATCGTCGCTTGAAA
64--snub--	ATCCTTGAAAGGAAATACCT
65--snub--	CTCAATCGTCATTAATTAATTTTCCCTTAGAACATTTTGACG
66--snub--	AAAGAACGCGTAATAACATC
67--snub--	ATTAGAGAAAACTTTTTTTTTTCAAATATATATGGT
68--snub--	TAGAAGAACTATGCAAATCCAATCGCAAGACACTTGCCTGAG
69--snub--	CCGATTAAAGGTGATAAATAAGGCGTTAAATTAAACAGGAGG
70--snub--	AAGAATAAACGAGCGGGAGC
71--snub--	TTCGAGCCAGCTGCGCGTAA
72--snub--	AGAGAATTTTTTATAAAGTACCCAATACTGCGTTTTTGAATCGTCATAACAG
73--snub--	GCCGCGCTTATGTAATTTAGGCAGAGGCATTCCACCACACCC
74--snub--	ATGCTTTAAAATATTCATTGAATCTCCTTTGC
75--snub--	CCGAACGACAACTCGTATTAAACCCCTCAA
76--snub--	TGGAAGGGTTGTTATCTAAA
77--snub--	CTACCATTTTTTATCAAAATTGTAGATTTTCATTTTTGGTTTAACGTGCCAC
78--snub--	AGCACTAACATTGGATTATACTTCTGAATAAATATCTTTAGG
79--snub--	CTGCAACAGTCAGATGAATATACAGTAACAGATTAACACCGC
80--snub--	TACCTTTTACGGCGGTCAGT
81--snub--	AATTGATTTTTGTTAAGCCCAATAGCTATCTTTTTTTACCGAAGCCCAGTTA
82--snub--	CCAGAATTTTTGGAAACCGAGAATAGGAACCCTTTTTATGTA
83--snub--	CCGTAAACGCCTGTAGTTTTTCATTCCACAGACAAG
84--snub--	TAACATTTTTTAAAAACAGGGACCCTGAACAATTTTTAGTCA
85--snub--	GAGGGGTTTTGTCGTCTTTTTTTTCCAGACGGCTAA
86--snub--	ACAACTTTTTTTTCAACAGTTAGAAACGATTTTTTTTTTTGTTTAACGAGAA
87--snub--	TTTTATTTTTTCCTGAATCTTTAATTTGCCAGTTTTTTTACAAAATAAAGGA
88--snub--	ATTGCGTTTTTAATAATAATTCTCCAAAAGGATTTTTGCCTT
89--snub--	TAATTAATTTCTTAAATTTTTCAGCTTGATATACAA
90--snub--	AGGTTTTTTTTTGAAGCCTTATCGCCCACGCATTTTTTAACCGATATAGGCC

91--snub--	GCTTTTTTTTTGCGGGATCGTCATCGTAGGAATTTTTTCATT
92--snub--	AAACCATTTTTATCAATAATCGTATTAAACCATTTTTAGTACCGCACGAGGG
93--snub--	TAGCAATTTTTCGGCTACAGAGAAGTTTCCATTTTTTTAAAC
94--snub--	GGGTACCTAAAACGAATTTTTAGAGGCAAAATGTAG
95--snub--	AATAAATTTTTCAACATGTTCAATAGATAAGTTTTTTCCTGA
96--snub--	ACAAGAGCGATTATACTTTTTCAAGCGCGAACTGAT
97--snub--	AAATTGTTTTTTGTCGAAATCAAAGAAGTTTTTTTTTGCCAGAGGGGACGAC
98--snub--	CGTTTATTTTTCCAGACGACGGAGGCGCAGACTTTTTGGTCA
99--snub--	ATCATGGACAGATGAATTTTTCGGTGTACAGAAGAG
100--snub--	TAATCTTTTTTTGACAAGAACTACATAACGCCTTTTTAAAAGGAATTACCCT
101--snub--	AATAATAGAAAGATTCTTTTTATCAGTTGAGAGCTG
102--snub--	CTCATTTTTTTCAGTGAATAAGTAGTAAATTGTTTTTGGCTTGAGATTACCT
103--snub--	TATGCGTTTTTATTTTAAGAAAAGAAAAATCTTTTTTACGTT
104--snub--	TGCTCAGTACGCAGCACCGTAATCAGTAGCGTAGCGGGGTTT
105--snub--	AATGAATTTTTACCATCGATACAGGCGGATAATTTTGTGCC
106--snub--	GTCGAACAAAAGGGCGTTTTTACATTCAACCATTAT
107--snub--	TCATTATTTTTAAGGTGAATTTAGAGCCAGCATTTTTAAATCACCAGTCACC
108--snub--	AGCGCCAAAGGAGGGTTGAT
109--snub--	CCGGAATAGGAGAAAATTCATATGGTTTACCATAAGTATAGC
110--snub--	CATAAATTTTTGGTGGCAACAGTTTATTTTGTTTTTTCACAATCAATTGTAT
111--snub--	CACCGTTTTTACTCAGGAGGCAGAACCGCCATTTTTCCCTC
112--snub--	AGAGCACTGGCATGATTTTTTTAAGACTCCTACATA
113--snub--	CAGTGCCTCAGAACCGCCACCCTTTTAGTAC
114--snub--	CGCCACCCCGTATAAACAGTTAATTGAGTAA
115--snub--	TACCCAAAAGACACCACCCTCA
116--snub--	TAGCAAGCCCGAAACGCAATAATAACGGAATTTTCAGGGA
117--snub--	TGATACAGACCAGTACAAACTACACACTGAG
118--snub--	TTTCGTCGAGTGTACTGGTAATAGCTTTTGA
119--snub--	GAGATAACCCACAGCCCTCA
120--snub--	ACGATCTAAATAATTGAGCGCTAATATCAGATAGTTAGCGTA
121--snub--	AAAGCGCATCTGTATGGGATTTTTTAGTAAA
122--snub--	TGAATTTGTCTCTGAATTTACCGCAGAATGG
123--snub--	AATCCAAATATCAGCGGAGT
124--snub--	GGAACAACTAAACAGCCATATTATTTATCCCGAGAATAGAAA
125--snub--	ACGATTGGTCTCCAAAAAAAAGGTTTTCACG
126--snub--	TTGAAAACCTTGATATTCACAAACAGGTCAG
127--snub--	CGCCACCCCTTGCTTTCGAGGTGGTATCGGT
128--snub--	TTATCAGTCAGAACCGCCACCCTCTCAGAGC
129--snub--	TGCACCCAGCCCGATAGTTG
130--snub--	ACAACAACCAAATCAAGATTAGTTGCTATTTCGCCGACAATG

131--snub--	ATAATCAACTTGCAGGGAGTTAAATTCGGTC
132--snub--	GCTGAGGAATCACCGGAACCAGAATCTTTTC
133--snub--	TTTTTATTTTCACCCTCAGC
134--snub--	GCATCGGAACTCATCGAGAACAAGCAAGCCGAGCGAAAGACA
135--snub--	TTTTCATCAGACTTTTTCATGAGGGCTTTGA
136--snub--	GGACTAAGGCATTTTCGGTCATAGTAGCGCG
137--snub--	TTACCATTCTACGAAGGCACCAAAAATACGT
138--snub--	AATGCCAAGCAAGGCCGGAAACGTAGCACCA
139--snub--	TTTACGAGCAGAATACACTA
140--snub--	TTTGACCCCCAAAAATAATATCCCATCCTAAAAACACTCATC
141--snub--	CACCGACTGATTTGTATCATCGCACAAAGTA
142--snub--	CAACGGATGAGCCATTTGGGAATATCACCGT
143--snub--	GGCTTTTGCACGCGACCTGC
144--snub--	TAGCCGGAACATAAAAACCAAAATAGCGAGATCCATGTTACT
145--snub--	GAGGGAAGACCAACTTTGAAAGAAAGGGAAC
146--snub--	CGAACTGGTAAATATTGACGGAAGATTGAGG
147--snub--	AAACGCAATGGCTGACCTTCATCACCAGGCG
148--snub--	CATAGGCAGACACCACGGAATAATATAAAAG
149--snub--	CTAATGCAGACGGATATTCA
150--snub--	AACGTAACAAATTTAGGAATACCACATTCAATTACCCAAATC
151--snub--	AGTATGTTGAAACACCAGAACGAGGCTTGCC
152--snub--	CTGACGAAGCAAACGTAGAAAATTATTACGC
153--snub--	TGTGAATGGTTTAATTTCAACTGCAGATAG
154--snub--	CCGAACAATTTTTAAGAAAAGTAATTAATCAT
155--snub--	ACAGAATCAAGGATTAGGAT
156--snub--	GAGAAGTTTGCCTTTATTTTTGCGTCAGACTGCCCC
157--snub--	CTTATTTTTTTAGCGTTTGCCGCCACCACCGGTTTTTAACCGCCTCCCAGAG
158--snub--	CCACCATTTTTCCCTCAGAGCGGCTGAGACTCTTTTCTCAA
159--snub--	AGTATTAAGACGCCACCAGAACCACCACCAGTGAAACATGAA
160--snub--	AGCCGCCGCCCCTATTATTC
161--snub--	GTGCCTGCCCCCTGCCTTTTTATTTCGGAAAGCAT
162--snub--	TGACAGTTTTTGAGGTTGAGGCAAATAAATCCTTTTTTCATTAAAGCTTCCA
163--snub--	GTAAGCTTTTTGTCATACATGAGTTTTAACGGTTTTTGGTCA
164--snub--	TTATACCGAAATAAAGAAATTGCATTTGCAC
165--snub--	GTAAAACAAGTCAGGACGTTGGGCTGGCTCA
166--snub--	GATTAAAAATCATAGGTTTTTTCTGAGAGACTATAA
167--snub--	GTTTTAGGGCTTAGGTTGGGTTATACCTTTT
168--snub--	ACCGCGCTTATCCGGTTTTTTATTCTAAGAAGCGGG
169--snub--	TAACCTCCCGAACCTCCCGACTTCGCGAGGC
170--snub--	CTAGAAATTCTTACCATTTTTGTATAAAGCCCCATA

171--snub--	CAGAACGGCTTAATTGAGAATCGAACGCTCA
172--snub--	ACAGTAGGCGCCTGTTTATCAACAGCTAATG
173--snub--	TTTGAACCAGAAGGAGTTTTTCGGAATTATCCATCA
174--snub--	AACGGAATCAGATGATGGCAATTATCATATT
175--snub--	CCTGATTACAACATTATTACAGGAACGAACT
176--snub--	ACGGAAATCGCGCAGATTTTTGGCGAATTATGAAAC
177--snub--	TAGACGGAGCAAAAGAAGATGATTCATTTCA
178--snub--	ATTACCTGGAGAATTAACTGAACAAGCGCAT
179--snub--	ATCAATATAGCAGCCTTTACAGAGTCAAAAA
180--snub--	TGAAAATATGTGAGTGAATAACCGTACATAA
181--snub--	GCAAGCATTCTGACCTAAATTTATTTAGTTA
182--snub--	ATTTCATCAATCAGATATAGAAGGCCCAATA
183--snub--	TGGATAGCGTCGACAAAAGG
184--snub--	CTGTCCAGACGGTAATAGTAAAATGTTTAGACTAAAGTAATT
185--snub--	ATCATTTTAGCAACACTATCATAACGAGGCA
186--snub--	TAGTAAGGCGGAACAAAGAAACCGTAACATT
187--snub--	ATTGCTTTAACAATGAAATAGCAATAATAAG
188--snub--	AGCAAGAGAATACCAAGTTACAATTCGCCTG
189--snub--	TAACGAGCAATAGTGAATTTATCAGACGCTG
190--snub--	AGAAGAGTCGTCTTTCCAGAGCCACCAACGC
191--snub--	TTCCTTATCATATGCGTTATACAAAAAGCCT
192--snub--	GTTTAGTATCATTCCAAGAACGGGGCTGTCT

APPENDIX C

SUPPORTING INFORMATION FOR CHAPTER 4

RECONFIGURABLE DNA ORIGAMI TO GENERATE QUASI-FRACTAL PATTERNS

Experimental Methods

Materials: M13mp18 ssDNA was purchased from New England Biolabs, Inc. (NEB, Catalog number: #N4040S). All other strands were purchased from Integrated DNA Technologies Inc. (www.IDTDNA.com) in 96-well plates at a 25 nmole synthesis scale, and were normalized to 200 □M x 100 □L and used without further purification.

Folding Frame 1: A one-step annealing reaction was used to form Frame 1; 10 nM M13mp18 ssDNA was mixed with 10x excess of all of the staples corresponding to the design for Frame 1 (100 nM each) in 1xTAE-Mg^{2+} buffer (20 mM Tris, pH 7.6, 2 mM EDTA, 12.5 mM $MgCl_2$). The oligonucleotide mixture was annealed in a thermocycler that was programmed to cool from 95°C to 4°C: 94°C to 86°C at 4°C per 5 minutes; 85°C to 70°C at 1 °C per 5 minutes; 70°C to 40°C at 1°C per 15 minutes; 40°C to 25°C at 1°C per 10 minutes; then hold at 4°C.

Standard Transformation: For the transformation from Frame 1 to Frame 2, first, 10 equivalents of Release Set 1 (42 strands, 100 nM each), fully complementary to Primer Set 1 (staples with an exposed toehold domain), are introduced to pre-formed Frame 1. This mixture is held at 45°C for 6 hours, generating a partially relaxed structure (particularly in the dynamic portions of the structure). Next, Closure Set 1 (38 strands, 100 nM each) is introduced to reorganize and fold the relaxed structure into Frame 2. The mixture is subsequently cooled from 45°C to 4°C (1°C per 10 minutes; then hold at 4°C). The transformation from Frame 2 to Frame 3 proceeds in a similar way as described above, except for the temperature used to prime the structures - the mixture was held at 37°C instead of 45°C for 6 hours.

All the samples are then subjected to AFM imaging without further purification.

AFM imaging: 1 □L sample was deposited onto a slide of freshly cleaved mica (Ted Pella, Inc.) and left to absorb for 1 min. 100 □L 1xTAE-Mg^{2+} Buffer was added to the liquid cell and the sample was scanned in a tapping mode on a Pico-Plus AFM (Molecular Imaging, Agilent Technologies) with SNL tips (Veeco, Inc.).

Transformation yield: The % yields reported in the main text were determined by tallying the number of target structures (structures expected after a given transformation) present in an AFM image and dividing by the total number of DNA origami structures (transformed and not-transformed) in the same AFM image. For all cases, no less than 100 total structures were analyzed.

Figure S1. Detailed design: toehold mediated strand displacement to transform preformed DNA origami Frame 1 into Frame 2. The upper panels are zoom out schematics and the lower panels are zoom in schematics of selected areas. (a) In Frame 1, staple strands (blue) in dynamic portions of the structure that crossover between helices 4 and 5 have toehold extension incorporated. Additional staple strands (rose) also contain toehold extensions (strategically located between helices 2 and 3) and will facilitate the transformation from Frame 2 to Frame 3. (b) In Frame 2, the 4th and 5th helices in the dynamic portions of the structure are separated and the inner helices in the corresponding corners are refolded towards the center of the frame. Removing the blue staples by toehold mediated strand displacement will allow the 4th and 5th helices to separate at the designed locations. The addition of new staple strands (red and yellow) refolds the structure into Frame 2. The yellow staples between helices 2 and 3 contain toehold extensions to facilitate the transformation from Frame 2 to Frame 3.

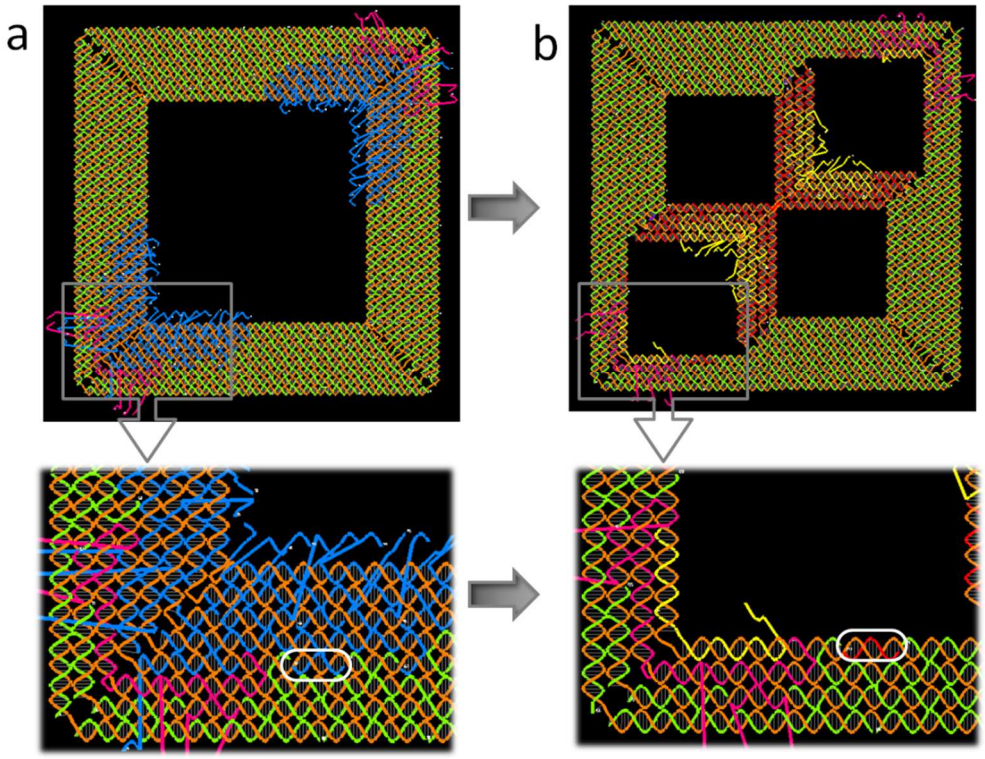


Figure S2. Intermediate structures. Left panels show the schematics, middle panels show zoom in AFM images, and right panels show zoom out AFM images. (a) Partially relaxed Frame 1 structure. (b) Partially relaxed Frame 2 structure. They are the product of priming the reconfigurable portions of the structures through the release of primer staples. The relaxed structures permit a certain degree of twist tolerance, facilitating the next step (refolding) of the transformation.

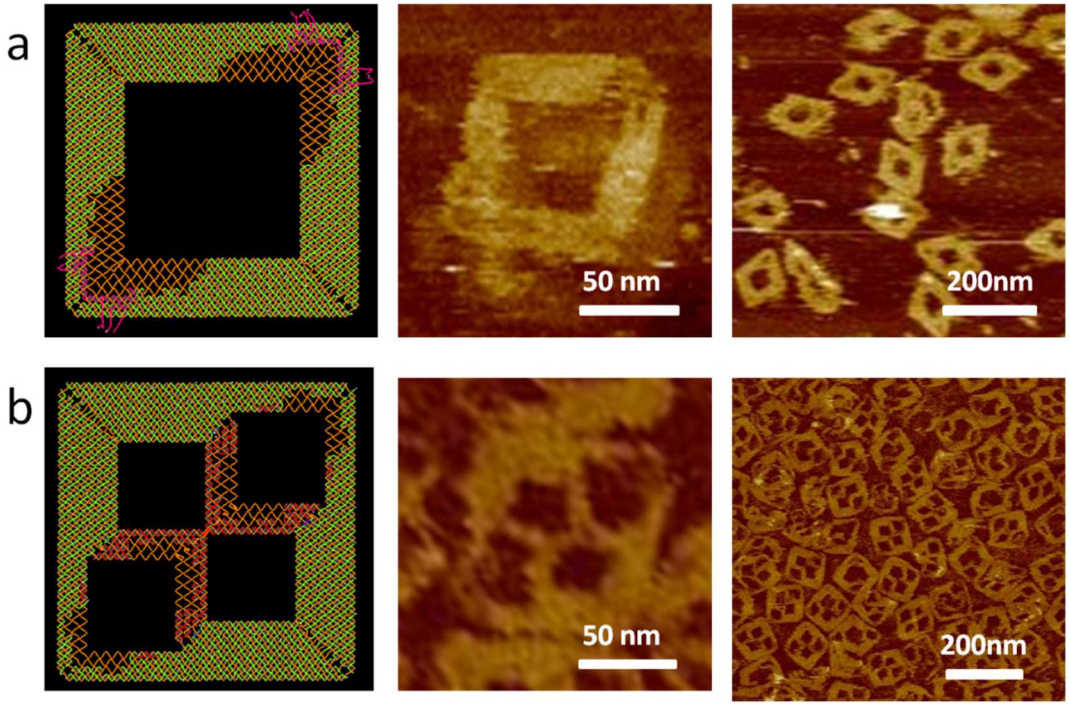
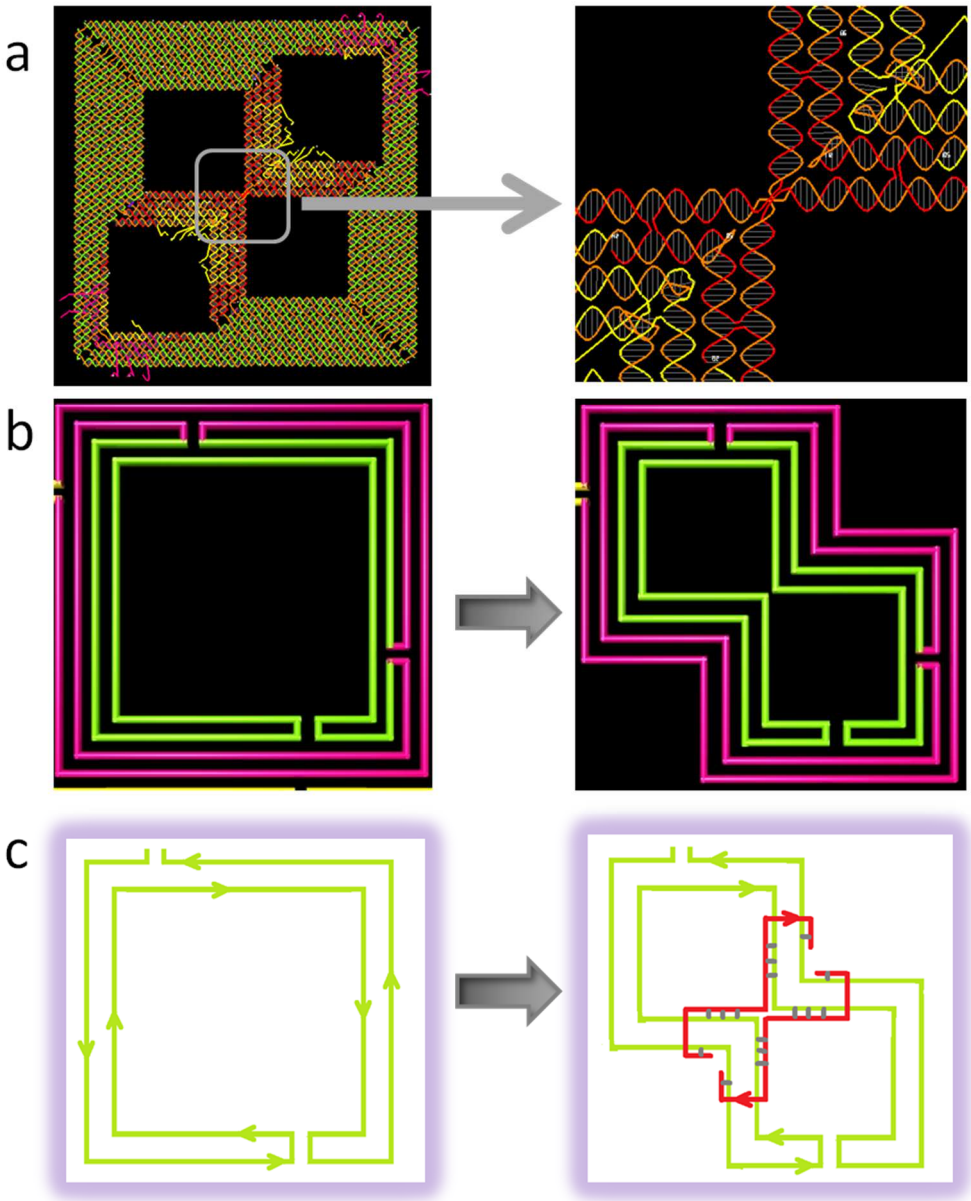


Figure S3. Detailed design: securing the connection between helices from opposite corners of the initial structures. (a) Schematic design of the connection point in Frame 2. The left panel is a zoom out schematic and the right panel is a zoom in schematic of the selected area. (b) Cylindrical model of the DNA origami frame structure. The panel on the left illustrates the inner four helical layers in Frame 1 and the panel on the right illustrates how these inner helices are refolded to form Frame 2. (c) Design of the two clamp strands (red) used to secure the two opposite corners. Each clamp staple binds to four exposed sections of the M13 scaffold: two from the lower left corner and two from the upper right corner. The center of the clamp staples contains five nucleotides (A_5) that are intentionally left unpaired to relax the tension that may exist in the center.



Temperature of transformation: The effect of temperature on structural reconfiguration was evaluated to identify suitable experimental conditions. In all cases, higher transformation temperatures destabilized the pre-formed Frame structures and led to re-annealing of DNA origami structures. Lower transformation temperatures resulted in low (or no) transformation yields. Figures S4-S7 describes the pertinent experiments and results.

Figure S4. Higher temperature for the transformation from Frame 1 to Frame 2. (a) The AFM image shows the product of adding Closure Set 1 directly to pre-formed Frame 1 at 65°C without adding Displacement Set 1 first. Unlike the same experiment carried out at 45°C (Figure 4c in the main text) in which no transformation was observed, a mixture of Frames 1 and 2 are evident here. This indicates that the higher temperature destabilizes the structure and facilitates incorporation of the closure staples by thermal annealing. (b) The AFM image shows the result of directly annealing a mixture of M13, all the staples required to fold the scaffold into Frame 1, and Closure Set 1 from 65°C. As in Figure S4a, a mixture of Frames 1 and 2 is evident. This further indicates that the higher temperature destroys the frame structure and initiates a new thermal annealing process.

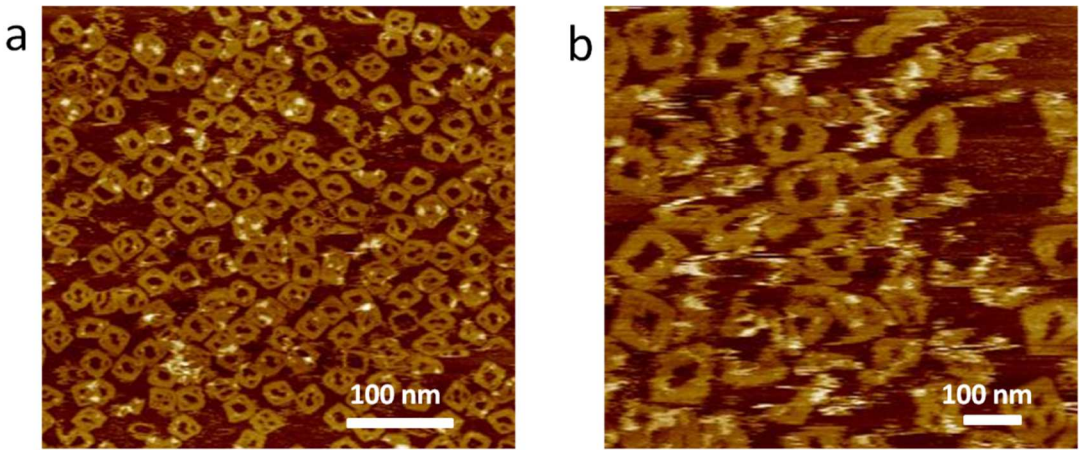


Figure S5. Direct thermal annealing of Frame 2. (a) The AFM image shows the result of directly annealing a mixture of M13 and the staples required to fold the scaffold into Frame 2 using a standard protocol (95-4°C). As expected, the yield of assembly is very high. (b) The AFM image shows the result of annealing the same mixture using a modified protocol (65-4°C). Similar to the standard protocol, the yield of assembly is high. These results support the conclusion that 65°C is higher than the melting temperature of the structure and will facilitate full thermal annealing.

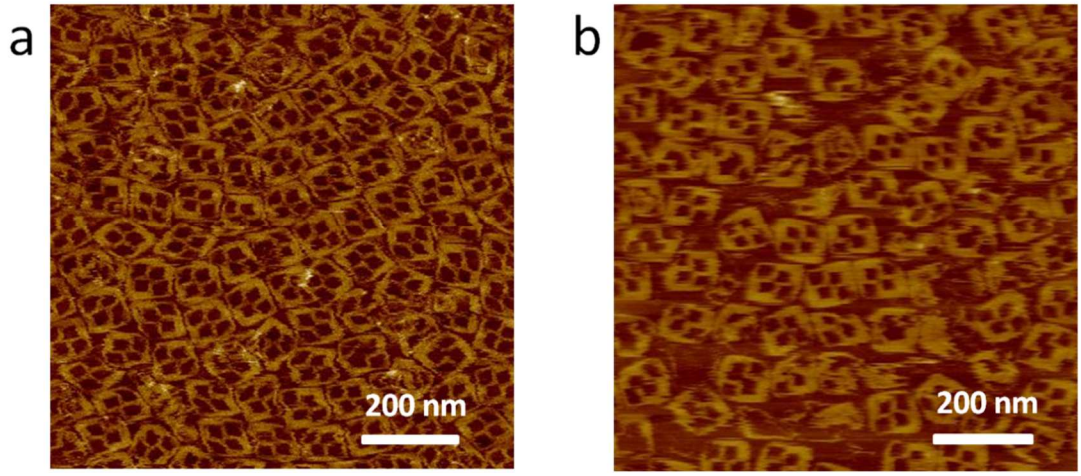


Figure S6. The effect of an elevated temperature on partially relaxed intermediate structures. (a) The AFM image shows the result of priming Frame 2 for the transformation to Frame 3 (by displacing the primer strands) at 65°C, rather than 37°C. As shown in the AFM image, the fractal-like features in the center of the structure are destroyed by the higher temperature, preventing transformation into Frame 3. (b) The AFM image shows the result of adding Closure Set 2 to the sample in (a). As evidenced by the presence of Frame 3 in the AFM image, the high temperature initiated a new thermal annealing process.

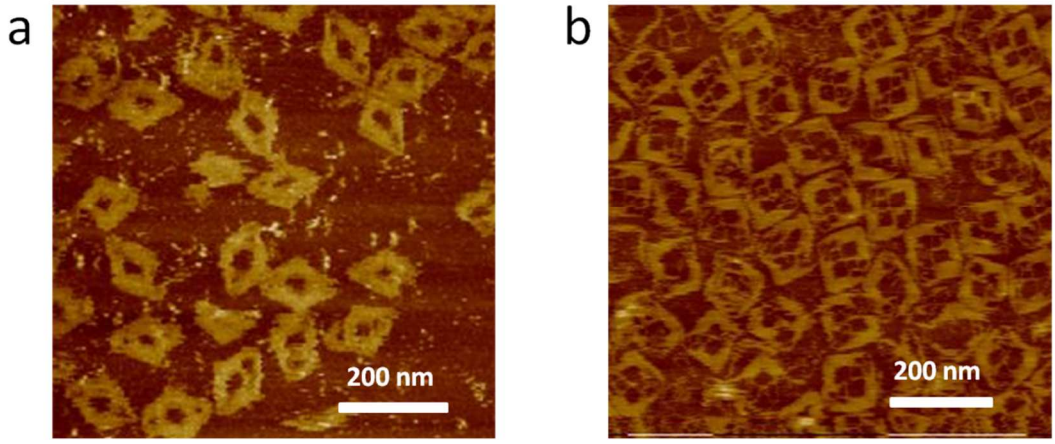
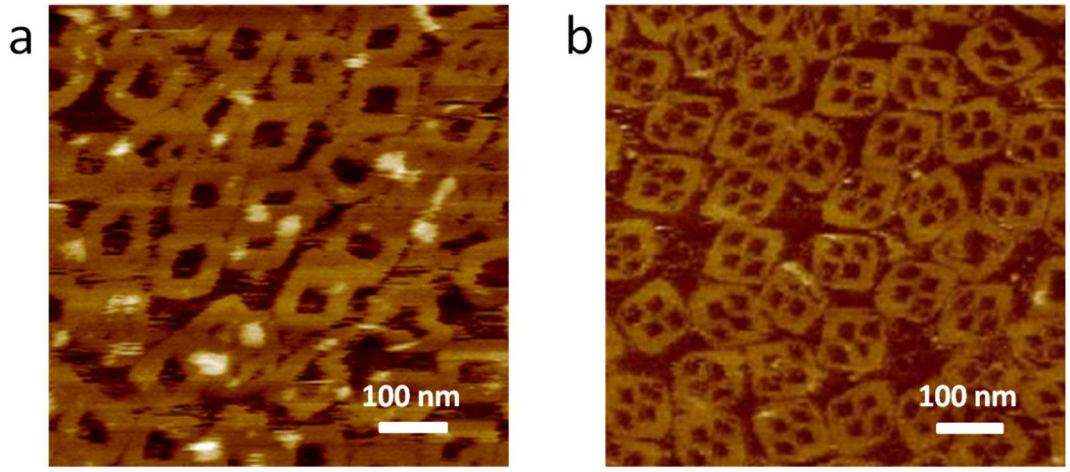


Figure S7. Lower temperature for the transformation from Frame 1 to Frame 2. (a) The AFM image shows the result of conducting both steps of the transformation from Frame 1 to Frame 2 at 37ºC. There is little evidence of Frame 2 in the AFM image. Considering the same experiment carried out at 45°C (Figure 2b in the main text), the result indicates that the lower temperature does provide enough energy to facilitate the transformation. (b) The AFM image shows the result of conducting both steps of the transformation from Frame 2 to Frame 3 at 30ºC. Again, there is little evidence of Frame 3 in the AFM image. The same experiment conducted at 37°C is shown in Figure 2c in the main text. This result further indicates that the lower temperature does not make the transformation possible.



The following table lists the sequences of the staples strands for Frame 1. All the sequences are from 5’ to 3’.

Number	Sequence
2	CAATTCCACACAACATACATCACTTGCCTGAGCCGGAACGAGGCGCAGTCCATGTT
3	ACTTAGCAAAACAACTTTCAACAGCACCGGAA
4	TCATAATTACTGGCATGATTAAGAAGAAAAGTAAGCAGAT
5	AGCCGAACAAAGTTAGGAATACCCAAAAGAACTAGAA
6	AAAGCCTGGTATGGGATTTTGCTGAGGTGAGGCGGTCACAGCAGA
7	AGATAAAATAGAAGAACTCAAACATTGTTATCCGCTCA
8	GGTCATAGCTGTTTCCTGTGTGAATATCGGCCTTGCTGGTATACCGAA
9	CGAACCACGTATTAACACCGCCTGAGTAAATGAATTTTCTTTTAGTATCATATGCG
10	GAAACGCAATAATAACCCAGAAGGAAACCGAGAGAAAAATAATATCCCAGATAAGT
11	CCTGAACATTATACAAATTCTTACCGTCTTTCCAGACGTTCAACAGTG
12	CCACGCTGAACATCGCCATTAAAAAATATCCAGAACAATAGTAATCAT
13	GCTTTGAGGTTTAATTTCAACTTTCACCCGCCGCGCTTAAAGCAAGCGGTCCACG
14	CTGGTTTGCCCCAGCATGCGCGTAACCACCAAATCATTGTGAATTACGAGGGTA
15	GCAACGGTCGGTCGCTGAGGCTTATCAAAA
16	TCATAGGTATTGACGGAAATTATTTACAGAGAGAATA
17	ACATAAAAACAGGGAAAGGGAGGGAAGGTAAATCTGAGAG
18	ACTACCTTCGCATAACCGATATATCTACAGAG
19	AGTGAATTTGCAGGGAGTTAAAGGAAAGACAG
20	CATCGGAACCTTATGCGATTTTAAAGTGTAGCGGTCACGCGGCGAAAATCCTGTTT
21	GATGGTGGTTCCGAAACGCTAGGGCGCTGGCAGAACTGGCTCATTATAGTCACCCT
22	CAGCAGCGCCGCTTTTGCGGGATCATTAAGAC
23	GCTGAGAATATCACCGTCACCGACTTTGTTTAACGTCAAA
24	AATGAAAATAGCAGCCTTCATTAAAGGTGAATGAGTCAAT
25	AGCCCTAAAGAGCCAGCAGCAAATCGATCTAAAGTTTTGTCAGTATA
26	AAGCCAACGCCTGTTTATCAACAATATCCTAATTTACGAGCATGTAGAA
27	ATCAATAATCGGCTGTCTTTCCTTTCCAGACGACGACAATCAACATGT
28	AATTTAGGCACTGAGTTTCGTCACCAAATCAACAGTTGAACATTCTG
29	GCCAACAGACCAGTAATAAAAGGGAGGGATGTGCTGCAAG
30	GCGATTAAGTTGGGTATCACCAGTCACACGAGATAGAACCCTTCTGTATCTGGTCAGTTGGCAGTACAAAC
31	TACAACGCCTGTAGCAACAGACAGCCCTCATA
32	TACCGAGCTCGAATTCTTACCGCCAGCCATTGGAACTGAT

33	GTTAGCGTAAGAAAAATCTAAAGCATAGTCTTTAATGCGCCAACAGGAAAAACG CTAGAGGATCCCCGGG
34	AATGCAGAACGCGCTCAACAGTAGGGCTTAATTATCGCCATATTTAACAACGCA AACAACATGTTC
35	ATTTACATTGGCAGATACGCCAGGGTTTTCCCAGTCACGACGTTGTAAAACG
36	CGTCTGAAATGGATTATACGTGG
37	ATTTTTGAATGGCTATTCACCTTGCTGAACCTTTCAAACCCTCAATCAAACCTGAA AGCGTAAGA
38	GGCCAGTGCCAAGCTTGCATGCCTGCAGGTCGACTCTCATGGAAATACCTAC
39	CAGACAATATTTTGACGCTCAAT
40	CTCTCCCGAGAAAACTTTTATATATTTTAGT
41	CTCTCCTAATTTCATCTTCTTGAAAATCTCCAAAAAGATTATAC
42	GTACAACGGAATAATTTTTTCACGTGACCTAAACTCTCC
43	TTTAATGGAGGTGGCAACATATAAAGCCCAATAATAAGACTCTCC
44	CTCTCCGCAAGAAACAATGAAAAAAATACATACATAATTTGAAA
45	CTCTCCTACCGACCAAAGGAATTGCGAATAGATTTG
46	ATTGTGTCATAGAAAGGAACAACTGTGTGATACTCTCC
47	AATAAGGCATGTTAGCAAACGTAGTAGCAATAGCTATCTTCTCTCC
48	CTCTCCACCGAAGCCCTTTTTACTCCTTATTACGCAGTGTTAAATA
49	AGAATAAATTTCAGCGGAGTGAGAGAAATCCGCGACCTGCACGGTCAA
50	CTCTCCCAAGAATTGAGTTAAAGAAACGCAAAGACACCA
51	AGAGTAATCTTGACTTTGAACCGGATATTCTGTGTG
52	ATTACCCAAATCAACGGCTAAACATGTGTG
53	GAGCGGGATAACAAAGCTGCTCATAGGCACCATGTGTG
54	ACTACGATCAGTGAATAAGGCTACGTGCTTTCCTCGTCTTTTCACCAGTGAG
55	ACGGGCAACAGCTGATTTGACGAGCACGTATATGCCCTGACGAGAAACTAAACG GG
56	GTTTCCATACCAGAACGAGTAGTAGCGTACTATGGTTGCTTGCCCTTCACCGCCT G
57	GCCCTGAGAGAGTTGCTGCGCCGCTACAGGGCAATTGGGCTTGAGATGGACTAA AG
58	TCATAAGGTCTTTGATTAGTAATAACGAGCCGGAAGCATA
59	AAGTGTAAAGCCTGGGCCGTTGTAGCAATACTGAACCGAACTGACCAACCTGAT AA
60	TATCATCGCTTTGAAAGAGGACACATCACGCAAATTAAGTGCCTAATGAGTGA
61	GCTAACTCACATTAATAGTAAAAGAGTCTGTCGATGAACGGTGTACAGAAACAA A
62	CAAGCGCGACCAGGCGCATAGGCTATCAGTGATGTGTG
63	TTTTTATAGGCTGACCTTCATCATGTGTG
64	ACTTTTTCCAACAACCATCGCCCATTTAACCT
65	CTCTCCTATATAACGTTGCGCCGACAATGAATGAGGAA
66	TAAAATACCAGCTTGATACCGATATATATGTACTCTCC
67	CTCTCCCAATCGCGGTGAATTTCTTAAAGTAATGCC
68	CTCTCCACCTAAAAATCAGCTTGCTTTCGAAAGACAAAGAACG
69	CTCTCCTTAATTGTATCGGTTTCGAAAGAGGCAAAAGAATACAC

70	CTCTCCACACTCATCTTTGACCCCCAGCAAAGGCTCCAAAAGGA
71	CCGGCTTAGACATTCAACCGATTGGCGCATTAGACGGGAGCTCTCC
72	CTCTCCAATTAACTGAACACCCCCAAAGACAAAAGGGCGGTTGGGT
73	AATGCTGAATATGGTTTACCAGCGTGAACAAAGTCAGAGCTCTCC
74	CTCTCCGGTAATTGAGCGCTAAATCAATAGAAAATTCTGCAAATC
75	ATAAGTTTATTTTGTCACATATCAGAGAGATAACTCTCC
76	GGCCACCGTGCGTTGCGCTCACTGCCCGCTTTCCAGTCGGGAGAAGTG
77	GAACGGTACGCCAGAATCCTGAAACCTGTCGTGCCAGCTGCATTA
78	GGAGGCCGCGGTTTGCGTATTGGGCGCCAGGGTGGTTTTTTAGAATCA
79	GAATCGGCCAACGCGCGGGGAGAGGATTAAAGGGATTTTAGACAG
80	CATAACCCATTGTTTGGATTATACCTATTTCGGAACCTATAGTTAATG
81	CCCCCTGCGCGACAGAATCAAGTTCTACAATTTTATCCTG
82	AATCTTACCAACGCTGCACCGTAATCAGTAATTACCT
83	TTTTTAATAACAATTTCATTTGATTCTGAATAATGGAAAGGCATA
84	GTAAGAGCAAAGGTGGCATCAATAAAAATAATTCGCGT
85	CTGGCCTTCCTGTAGCTTGGGGCGCGAGCTGAAACACTAT
86	GTGAACCAGCATTGGAACGCCATCTCTACTAATAGTAGTAGCCAAAAG
87	GAATTACGGGGTTAGAACCTACCAAAAATTAATTACATTTGGAAACAGTACATAAA
88	GAAACCATCGATAGCAAACGAGCGTCTTTCCAGAGCCTAATTTGCCAGCGGAAACG
89	TCACCAATTCAATATATGTGAGTGCAAACATCAAGAAAACTATCAAAA
90	TTATTTGCAATGCAGATACATAACTCACCCAAATCAAGTTCCCACTAC
91	TAGAGCCGAGAAGCAAAGCGGATTGGTCAGGATTAGAGAGGCGCAACTGTTGGGA
92	AGGGCGATCGGTGCGGAAGCAAACTCCAACAGCATCAAAAAGATTAAGGAGCAC
93	TAACAACCCTCATTTTCAGGGAGCCGCCGC
94	CAGCATTAGAACCACCACCAGACGCACTCATCGAGAA
95	CAAGCAAGCCGTTTTTCCTCAGAGCCGCCACCGACAGGAG
96	GTTGAGGCCCCTCAGAGCCACCACTAATAGAT
97	AAGAGAATTAGCAAGCCCAATAGGTATCTAAA
98	ATATCTTTAGAGGAAGCCCGAAAGAAGCGAACCAGACCGGGCCTCTTCGCTATTAC
99	GCCAGCTGGCGAAAGGTTTAATTCGAGCTTCAACTTCAAATATCGCGTAGGAATTG
100	AGGAAGGTAACCCATGTACCGTAACAGAGGCA
101	TTTTCGAGAGGTAAAGTAATTCTGATCATTCCAAGAACGG
102	GTATTAAACCAAGTACATAAAGTACCGACAAACCAGTAAT
103	ATTCAACTACGTAAAACAGAAATAAAAGAAGATGATGAAAAATAACC
104	TTGCTTCTGTTACCATTAGCAAGGCTTACAAAATAAACAGCCATATTAT
105	CCCAATCCAAATAAGAAACGATTTTTGAGCCATTTGGGAACATAGCGA
106	TAGCTTAGTCGCCTGATTGCTTTGAGAAACAATAACGGATCCAGTCA
107	GGACGTTGGAAGCGAAAGGAGCGGGTCGGCAAAATCCCTT
108	ATAAATCAAAAGAATAGAAAGGAAGGGAAGAGAAGAAAAATCTACGACCTTTT

	ACATCGGGAATACCAAGT
109	TACAAAATCGCGCAGAAATTATTCATTTCAAT
110	TCTATCAGGGCGATGGTTTTGGGGTCGAGGTGAATACCAC
111	TACCTGAGCAAAGAAATTGCGTAGACAGTTGAGATTTAGGCCGTAAAGCACTAA AAAGGGCGAAAAACCG
112	ACCAGTAGCACCATAAATCGTCGCTATTAATTATCCCTTAGAATCCTTGAAAATT AGAGCCAGCAA
113	CCGGCGAACGTGGCGAGCCCGAGATAGGGTTGAGTGTTGTTCCAGTTTGGAA
114	GCTTGACGGGGAAAGACGGAACA
115	AGGTAGAAAGATTCATTTTTCAGGTTTAACGTTAATATACAGTAACAGTTTAATA AAACGAACTA
116	AGAGTCCACTATTAAAGAACGTGGACTCCAACGTCATCGGAACCCTAAAGGG
117	ATTATTACAGCCCCCGATTTAGA
118	ATTTACCGTTCCAGCGTCATACATGCTCTCC
119	CTCTCCGCTTTTGATGATACGGTTTTGCTCAGTACCCAAAGAAA
120	GAATTATCAGGATTAGGATTAGCGGAGGAGTGTCTCTCC
121	ACTGGTAAATAGCCCCCTTATTAGACCTCCCGACTTGCGCTCTCC
122	CTCTCCGGAGGTTTTGAAGCCTTCGGCATTTTCGGTCTAAGTTT
123	CTCTCCTAACGGGGGACTCCTCAAGAGAATCATATT
124	AGTATTAAGAGGCTGATCAGTGCCCTCTCC
125	TTGAGTAACTGTAGCGCGTTTTCATAAATCAAGATTAGTTCTCTCC
126	CTCTCCGCTATTTTGCACCCAGTGCCTTTAGCGTCAGACAGTGCCC
127	GTATAAACTATTCTGAAACATGAACATCAATATAATCCTGTCGTTTAC
128	CTCTCCAGGCGTTTTAGCGACGTTTGCCATCTTTTCATA
129	TAGCGTCCAATACTTTTGGAATCGTCATAATGTGTG
130	ATATTCATTGAATCCCTCAACATGTGTGTG
131	GCTGTAGCCCTCAAATGCTTTAAAATTAAATCTGTGTG
132	AACTCGTCAGTTCAGAAAACGAAGAGCTTAATTGCTGCCAGCTTTCCGGCAC
133	CGCTTCTGGTGCCGGATTTTTGCGGATGGCTTGAATGACCATAAATCAATTAGAC T
134	TTGAGGATTTAGAAGTAAAATCAGGTCTTTACTTTTGATAAGAGGTCAAACCAGG CAAAGCGCC
135	ATTCGCCATTCAGGCTTACCTTTAATTGCTCCCCTGACTATTATAGTCTCAATAGA
136	CAGACGACTTAGCTATATTTTCATCAGCTTTCATCAACAT
137	TAAATGTGAGCGAGTAATGGTCAATAACCTGTGATAAAAACCAAAATATCAGAT GATGGCAATT
138	CCTGATTAGCGAGAGGCTTTTGCAGATACATTTCGCAAACAACCCGTCGGATT
139	CTCCGTGGGAACAAACATTTAGTTTGACCATTAAAAGAAGTTTTGCCAAGGAGCG
140	CCACCAGAGAGGGGGTAATAGTAATTCTGCGATGTGTG
141	ATTCCCAAAATGTTTAGACTGGATGTGTG
142	TAATACATACCCTCAGAACCGCCAAGGTCAGA
143	TTACAAACGGAGGTTTAGTACCGCAAATAAATCTCTCC
144	CTCTCCATGGAAAGTATCACCGTACTCAAATTCGAC
145	CTTTGCCCATAGCCCGGAATAGGTGCGCAGTCTCTGACTCTCC
146	CTCTCCGAGGGTTGATATAAGTGAACGTTATTAATTTTAAAAGT

147	CTCTCCGTAACATTATCATTTTGCGGAAAGGCGGATAAGTGCCG
148	CTCTCCTCACAAACCACCCTCAGAACCGCC
149	CGATTGGCCCCTCAGAGCCACCACATTTTCATCGTAGGAACTCTCC
150	CTCTCCTCATTACCGCGCCCAAACCCTCAGAACCGCCACTTGATAT
151	CCTCATTACTCCCTCAGAGCCGCCTAGCAAGCAAATCAGCTCTCC
152	CTCTCCATATAGAAGGCTTATCCACCACCGGAACCGCAAGCCAGA
153	CTCTCCAAATCACCGGAACCAGAGCCGGTATTCTAAGAA
154	ACGAGTAGGGCGGATTGACCGTAATGGGATAGGTCACGTTAACAGTTG
155	CTGGAAGTTTCATTCCATATGGTGTAGATGGGCGCATCGTAACC
156	TTTTAAATACAGTATCGGCCTCAGGAAGATCGCACTCCAGAATATAAT
157	ATCTGCCAGTTTGAGGGGACGACGATGCAACTAAAGTACGGTGT

The following table lists the sequences of Release Set 1 and Closure Set 1 for the transformation from Frame 1 to Frame 2.

Release Set 1	Sequence
s40	ACTAAAATATATAAAAGTTTTCTCGATCTAC
s41	GTATAATCTTTTTGGAGATTTTCAAGAAGATGAAATTAACCTTG
s42	TCTATCTTTAGGTCACGTGAAAAAATTATTCCGTTGTAC
s43	GATTGTTCTTATTATTGGGCTTTATATGTTGCCACCTCCATTAAA
s44	TTTCAAATTATGTATGTATTTTTTTCATTGTTTCTTGCTACCAC
s45	CAAATCTATTCGCAATTCCTTTGGTCGGTAGGGGTC
s46	CTTACATATCACACAGTTGTTCCTTTCTATGACACAAT
s47	AAACCTAAGATAGCTATTGCTACTACGTTTGCTAACATGCCTTATT
s48	TATTTAACACTGCGTAATAAGGAGTAAAAAGGGCTTCGGTTAGCAG
s50	TGGTGTCTTTGCGTTTCTTTAACTCAATTCTTGTCTTCC
s65	TTCCTCATTCATTGTCGGCGCAACGTTATATAGTGGTC
s66	GAACCCTACATATATATCGGTATCAAGCTGGTATTTTA
s67	GGCATTACTTTAAGAAATTCACCGCGATTGGTGAAA
s68	CGTTCTTTGTCTTTCGAAAGCAAGCTGATTTTTAGGTACATTC
s69	GTGTATTCTTTTGCCTCTTTCGAAAGGCATACAATTAAGGGTAC
s70	TCCTTTTGGAGCCTTTGCTGGGGGTCAAAGATGAGTGTTGGCTA
s71	CCTAAACTCCCGTCTAATGCGCCAATCGGTTGAATGTCTAAGCCGG
s72	ACCCAACCGCCCTTTTGTCTTTGGGGGTGTTCAGTTAATTGATGAC
s73	TAACGGCTCTGACTTTGTTCACGCTGGTAAACCATATTCAGCATT
s74	GATTTGCAGAATTTTCTATTGATTTAGCGCTCAATTACCTCTGCC
s75	CCTCGGTTATCTCTCTGATATGTGACAAAATAAACTTAT
s118	TCCAAACATGTATGACGCTGGAACGGTAAAT
s119	TTTCTTTGGGTACTGAGCAAAACCGTATCATCAAAAGCAGTTTA
s120	GTACAAACACTCCTCCGCTAATCCTAATCCTGATAATTC
s121	CAGCGCCGCAAGTCGGGAGGTCTAATAAGGGGGCTATTTACCAGT
s122	AAACTTAGACCGAAAATGCCGAAGGCTTCAAAACCTCCACTCTT
s123	AATATGATTCTCTTGAGGAGTCCCCCGTTATTTTCG
s124	CTAGAAGGCACTGATCAGCCTCTTAATACT
s125	TTCCCCAACTAATCTTGATTTATGAAAACGCGCTACAGTTACTCAA

s126	GGGCACTGTCTGACGCTAAAGGCACTGGGTGCAAAATAGCGGGAGG
s128	TATGAAAAGATGGCAAACGTCGCTAAAACGCCTCTTGCA
s143	TATTGAATTTATTTGCGGTACTAAACCTCCGTTTGTAA
s144	GTCGAATTTGAGTACGGTGATACTTTCCATAAATCG
s145	AACCTTTCAGAGACTGCGCACCTATTCCGGGCTATGGGCAAAG
s146	ACTTTTAAAATTAATAACGTTCACTTATATCAACCCTCCACTAT
s147	CGGCACTTATCCGCCTTTCCGCAAAATGATAATGTTACCAAGGA
s148	GGCGGTTCTGAGGGTGGTTTGTGACCATTC
s149	TCACGTTTCCTACGATGAAAATGTGGTGGCTCTGAGGGGCCAATCG
s150	ATATCAAGTGGCGGTTCTGAGGGTTTGGGCGCGGTAATGAATTCCT
s151	GGGGACCTGATTTGCTTGCTAGGCGGCTCTGAGGGAGTAATGAGG
s152	TCTGGCTTGCGGTTCCGGTGGTGGATAAGCCTTCTATATCTGTGC
s153	TTCTTAGAATACCGGCTCTGGTTCCGGTGATTTCAACAT
Closure Set 1	**Sequence**
78	TTAGGGTTCCAGAAAGCCTC
79	TCGGGGAGGCGGATAAGTGCCGATACATGGCTTTTGACCCTTATT
80	CATAGCCTGATACAGGAGTGTACGGTTTTGCTCAGTACCGTTGCT
81	TAAACGGGATTAGGATTAGCGGTGGTAATAAGTTTTAAGCGCGTTT
82	TAGCGTCAGACTGTACGGGGTCAGTGCCTTGAGACTCCTCAAGAGAA
83	TATTAAGAGGCTGAGTAACAGTG
85	ATAAGTTTATTTTGCTAATATCAGAGAGATAAAAAAAGGCGTTTTAGCGAACCTCGCCATCTTTTCATA
86	AGCGTTTCCGACTTGCGGGAGGTTTTGAAGCCTTAAATATTTTCGGT
87	TCATCGGCCAAGATTAGTTGCTATTTTGCACCCA
102	TTACAAACAATTCGAC
103	CTTTGCCCGAACGTTATTAATTTTAAAAGATAACATTATCATTTTGCGGAACAAAGAAAATGGCC
104	GAATTATCATCATATT
105	CCACCGGATCTCTGAATTTACCGGAGGGTTGATATAAGTGGGCCA
106	TGGGGCATAGCCCGGAATAGGTCAGAATGGAAAGCGCAGACCGCCTC
107	CTCAGAACAATCCTCATTAAAGCGTATCACCGTACTCAGTAGCCT
108	GAGGTTTAGTACCGCTATTCACAAACAAATACGCCACCCTCAGAGC
109	TTGGCCTTGACACCCTCAGAACCGCC
110	TTTTCATCGTAGGAATCATTACCGCCCGCCACC
111	CCTCAGAGGCCCAATAGCAAGCAAATCAGATATAGAAGGAGAGCCA
112	AAATCACCGGAACCCTTATCCGGTATTCTAAGAAAAAACAAGAATTGAGTTAAGCCCACGCAAAGACACCA
118	CTTTTACTTTTTAAAATAT
119	CAGTACAAAGGCTCCAAAAGGAATTTTAGTTAATTTCACAACATAT
120	AAAGGTGGTCTTCTGACCTAAATTTGAAAATCTCCAAAAATGGGCG
121	TTCGACAATAATTTTTTCACGTTAATGGTTTGAAATACAGCAAACG
122	ATTACGCAGTATGTTCGACCGTGTGATAAATTAAAGGAATTGCGAAT
123	AGAAAGGAACAACAAGGCGTTAA
139	TAAAATACGTAATGCC
140	ACCTAAAACGAAAGAGGCAAAAGAATACAACACTCATCTTTGACCCCCAGCGATTATACGTCTCA
141	GTACAACGGAGATTTG

142	CAATAGAAAAGAACGCGAGAAAATTAATTGTATCGGTTTCTACAC
143	TAAACAATCAGCTTGCTTTCGAAATCCAATCGCAAGACAAATTCATA
144	AAGACAAAGTAAATGCTGATGCAGGTGAATTTCTTAAAATGGGG
145	CAGCTTGATACCGATAGGTTATATAACTATATAGGGCGACATTCAAC
146	GCTTAGGTTGGTTGCGCCGACAATGA
147	TGGTTTACACACCCTGAACAAAGTCAGAGGGTAATTGAGCGTCACAAT
148	CGCATTAGACGGGAGAATTAACTGACAGCGCCA
151	TAGAAAATATAGCTATCTTACCGAAGCCCTTTTT
152	AAAAGAAAATAATAAGAGCAAGAAACAATGAAATAGCAACATACAT

The following table lists the sequences of Release Set 2 and Closure Set 2 for the transformation from Frame 1 to Frame 2.

Release Set 2	Sequence
78	GAGGCTTTCTGGAACCCTAA
79	AATAAGGGTCAAAAGCCATGTATCGGCACTTATCCGCCTCCCCGA
80	AGCAACGGTACTGAGCAAAACCGTACACTCCTGTATCAGGCTATG
81	AAACGCGCTTAAAACTTATTACCACCGCTAATCCTAATCCCGTTTA
103	GGCCATTTTCTTTGTTCCGCAAAATGATAATGTTAACTTTTAAAATTAATAACGTTCGGG CAAAG
105	TGGCCCACTTATATCAACCCTCCGGTAAATTCAGAGATCCGGTGG
106	GAGGCGGTCTGCGCTTTCCATTCTGACCTATTCCGGGCTATGCCCCA
107	AGGCTACTGAGTACGGTGATACGCTTTAATGAGGATTGTTCTGAG
118	ATATTTTAAAAAGTAAAAG
119	ATATGTTGTGAAATTAACTAAAATTCCTTTTGGAGCCTTTGTACTG
120	CGCCCATTTTTGGAGATTTTCAAATTTAGGTCAGAAGACCACCTTT
121	CGTTTGCTGTATTTCAAACCATTAACGTGAAAAAATTATTGTCGAA
140	TGAGACGTATAATCGCTGGGGGTCAAAGATGAGTGATGTATTCTTTTGCCTCTTTCGTTT TAGGT
142	GTGTAGAAACCGATACAATTAATTTTCTCGCGTTCTTTTCTATTG
143	TATGAATTTGTCTTGCGATTGGATTTCGAAAGCAAGCTGATTGTTTA
144	CCCCATTTTAAGAAATTCACCTGCATCAGCATTTACTTTGTCTT
88	TTATGACGATTCCAAAAGTATTGGACGCTAAGGAGA
89	GAAAGCCATGTTGAGGGATTCAATGAATAT
90	GATTTAATTTTAAAGCATTTGAGGGCTACAGCTCTTTC
99	GTATTTTCGCAGAATTACTATTACCCCCTCTCTGGTGG
100	TCCAGTCTAAACATTTTGGGAATGGAAAC
125	GAATATCCGGTTCAAAGTCAAGATTACTcTAGATCA
126	AAACACTGTTTAGCCGTTGATTTGGGTAAT
127	TCAAGTTGGTGCCTATGAGCAGCTTTGTTATCCCGCTC
136	CTAGTTTCACTGATAGCCTATGCGCCTGGTCGCGCTTG
137	TGATGAAGGTCAGCCTATAAAAACAATAC
Closure Set 2	**Sequence**
78	GCGTCGATAAGTGCCAAAATAACATTATCACCAATAC
79	TTTGATGATACGGTTTTGCTCAGTACCAGGCGATACATGGCT
80	ATTAGCGGAGGAGTGTACTG
81	GTAATAAGTTTTAAGCGCGTTT

88	GAATCGTAATTTTAAAAGTAAAAAGGGTTGATATTCCA
89	GAACGTTATTCATAAATATTCATTGAATCCCATCCTTTGCCC
90	ATTAACCTCAAATGC
101	GTAATAGTAACCAGA
102	CTGGATAGCGTTTTTGCGGAACAAAGAAACCAAATGTTTAGA
106	CCGGAATAGGTGCGCAGTCTCTGAATTTACCGTAAGTATAGC
107	AGAATGGAAAGTATCACC
112	CTCAGAACAATCCTCATTAAAGC
116	CCACCGGAACCGCCTC
118	TCATAGCCCCCTTATT
121	ATTCCCAATTCTGCGA
123	GCTGTAGCTCAACATG
126	ATATACTCCAAAAGGAAAACACTCATCTTTATCTTGA
127	CAAAAAAAAGGTTTTAGTTAATTTCATCTTCTTGAAAATCTC
128	CACGTGACCTAAATT
129	TAATGGTTTGAAATACAGCAAACG
133	AACCGGAAAAAGAATACACAAAAATTAATTGTATCCTTTT
134	CAAATCAACGCCAACCTAAAACGAAAGAGGCTATTCATTACC
135	AGGCATAACAAAGCT
146	CGCATAGGCTGCGCG
147	CATCAAGAGTAGACCCCCAGCGATTATACCAAGGCTGACCTT
151	CGCGAGAAAAGGTTTATCAGCTTGCTTTCGAAAGACAAAGAA
152	TCCAATCGCGGTGAA
157	AAGACAAAGTAAATGCTGATGCA
159	CAATAGAAAATTCATA
165	AAAGGTGGCAACATAT
168	TTTTTATAATCAGTGA
170	GAGCGGGAGCTAAACA

The following table lists the sequences of Release Set 3 and Closure Set 3 for the transformation from Frame 3 to Frame 2.

Release Set 3	Sequence
78	CTGGGAGTATTGGTGATAATGTTATTTTGGCACTTATCGACGC
79	ACACGTAGCCATGTATCGCCTGGTACTGAGCAAAACCGTATCATCAAA
80	CCAAATCAGTACACTCCTCCGCTAAT
81	ACGGAGAAACGCGCTTAAAACTTATTAC
88	TCCATGTGGAATATCAACCCTTTTTACTTTTAAAATTACGATTC
89	GGGCAAAGGATGGGATTCAATGAATATTTATGAATAACGTTCTACGTC
90	CTCCTTGCATTTGAGGTTAAT
101	TCTGGTTACTATTACTACGAA
102	TATATGTCTAAACATTTGGTTTCTTTGTTCCGCAAAAACGCTATCCAG
106	GCTATACTTACGGTAAATTCAGAGACTGCGCACCTATTCCGGAGTTAT
107	TCGGTCGGTGATACTTTCCATTCT
112	GGAATCGCTTTAATGAGGATTGTTCTGAG

116	GAGGCGGTTCCGGTGGTCTAGG
118	AATAAGGGGGCTATGACGGAGG
121	TCGCAGAATTGGGAATATACTT
123	TAACGACATGTTGAGCTACAGC
126	TCAAGATAAAGATGAGTGTTTTCCTTTTGGAGTATATCCTAAT
127	GAGATTTTCAAGAAGATGAAATTAACTAAAACCTTTTTTTTGGCCTTG
128	ACCTAGAATTTAGGTCACGTG
129	TCTACACGTTTGCTGTATTTCAAACCATTA
133	AAAAGGATACAATTAATTTTTGTGTATTCTTTTTCCGGTTGCGACC
134	GGTAATGAATAGCCTCTTTCGTTTTAGGTTGGCGTTGATTTGTCCATT
135	GGCGAAAGCTTTGTTATGCCT
146	CGCGCAGCCTATGCGATTGCG
147	AAGGTCAGCCTTGGTATAATCGCTGGGGGTCTACTCTTGATGCCCGGA
151	ATAATCTTCTTTGTCTTTCGAAAGCAAGCTGATAAACCTTTTCTCGCG
152	CTTTGGTTCACCGCGATTGGA
157	CAACTCTGCATCAGCATTTACTTTGTCTT
159	TATGAATTTTCTATTGTCGTAG
165	TGATCTATATGTTGCCACCTTT
168	TCACTGATTATAAAAATCGTCG
170	GCTCGTTGTTTAGCTCCCGCTC
Closure Set 3	**Sequence**
78	GCGTCGATAAGTGCCAAAATAACATTATCACCAATACTCCCAG
79	TTTGATGATACGGTTTTGCTCAGTACCAGGCGATACATGGCTACGTGT
80	ATTAGCGGAGGAGTGTACTGATTTGG
81	GTAATAAGTTTTAAGCGCGTTTCTCCGT
88	GAATCGTAATTTTAAAAGTAAAAAGGGTTGATATTCCACATGGA
89	GACGTAGAACGTTATTCATAAATATTCATTGAATCCCATCCTTTGCCC
90	ATTAACCTCAAATGCAAGGAG
101	TTCGTAGTAATAGTAACCAGA
102	CTGGATAGCGTTTTTGCGGAACAAAGAAACCAAATGTTTAGACATATA
106	ATAACTCCGGAATAGGTGCGCAGTCTCTGAATTTACCGTAAGTATAGC
107	AGAATGGAAAGTATCACCGACCGA
112	CTCAGAACAATCCTCATTAAAGCGATTCC
116	CCTAGACCACCGGAACCGCCTC
118	CCTCCGTCATAGCCCCCTTATT
121	AAGTATATTCCCAATTCTGCGA
123	GCTGTAGCTCAACATGTCGTTA
126	ATTAGGATATACTCCAAAAGGAAAACACTCATCTTTATCTTGA
127	CAAGGCCAAAAAAAAGGTTTTAGTTAATTTCATCTTCTTGAAAATCTC
128	CACGTGACCTAAATTCTAGGT
129	TAATGGTTTGAAATACAGCAAACGTGTAGA
133	GGTCGCAACCGGAAAAAGAATACACAAAAATTAATTGTATCCTTTT

134	AATGGACAAATCAACGCCAACCTAAAACGAAAGAGGCTATTCATTACC
135	AGGCATAACAAAGCTTTCGCC
146	CGCAATCGCATAGGCTGCGCG
147	TCCGGGCATCAAGAGTAGACCCCCAGCGATTATACCAAGGCTGACCTT
151	CGCGAGAAAAGGTTTATCAGCTTGCTTTCGAAAGACAAAGAAGATTAT
152	TCCAATCGCGGTGAACCAAAG
157	AAGACAAAGTAAATGCTGATGCAGAGTTG
159	CTACGACAATAGAAAATTCATA
165	AAAGGTGGCAACATATAGATCA
168	CGACGATTTTTATAATCAGTGA
170	GAGCGGGAGCTAAACAACGAGC

The following table lists the sequences of Release Set 4 and Closure Set 4 for the transformation from Frame 2 to Frame 1.

Release Set 4	Sequence
R82	CCCCTATTCTCTTGAGGAGTCTCAAGGCACTGACCCCGTACAGTCTGACGCTA
R83	CACTGTTACTCAGCCTCTTAATATCATAA
R86	ATGACTACCGAAAATATTTAAGGCTTCAAAACCTCCCGCAAGTCGGAAACGCT
R87	TGGGTGCAAAATAGCAACTAATCTTGGCCGATGATCGCTT
R102	GTTATTGTCGAATTGTTTGATT
R104	AATATGATGATAATTCAGACAG
R108	GCTCTGAGGGTGGCGTATTTGTTTGTGAATAGCGGTACTAAACCTCCCGCCT
R109	GTCATCGGCGGTTCTGAGGGTGTCAAGGCCAA
R110	ACACGGGGTGGCGGGCGGTAATGATTCCTACGATGAAAA
R111	CTAGCATGGCTCTCCTTCTATATCTGATTTGCTTGCTATTGGGCCTCTGAGG
R122	TCCCGTATTCGCAATTCCTTTAATTTATCACACGGTCGAACATACTGCGTAAT
R123	TTAACGCCTTGTTGTTCCTTTCTCGACTG
R139	GGGGGGGGCATTACGTATTTTA
R141	CAAATCTCCGTTGTACTCGAGC
R145	CTGCCGGTTGAATGTCGCCCTATATAGTTATATAACCTATCGGTATCAAGCTG
R146	GAAGCATCATTGTCGGCGCAACCAACCTAAGC
R147	TAAGTGACGCTCAATTACCCTCTGACTTTGTTCAGGGTGTGTAAACCACGTTGG
R148	ACGTGCTGGCGCTGTCAGTTAATTCTCCCGTCTAATGCG
R151	AAAAAGGGCTTCGGTAAGATAGCTATATTTTCTAGTGCGA
R152	ATGTATGTTGCTATTTCATTGTTTCTTGCTCTTATTATTTTCTTTTCAAGAC
R85	TATGAAAAGATGGCGAGGTTCGCTAAAACGCCTTTTTTTATCTCTCTGATATTAGCAAAA T AAACTTATTCATCT
R112	TGGTGTCTTTGCGTGGGCTTAACTCAATTCTTGTTTTTTCTTAGAATACCGGATAAGGGTT CCGGTGATTTCAGTCG
78	GAGGCTTTCTGGAACCCTAA
79	AATAAGGGTCAAAAGCCATGTATCGGCACTTATCCGCCTCCCCGA
80	AGCAACGGTACTGAGCAAAACCGTACACTCCTGTATCAGGCTATG

81	AAACGCGCTTAAAACTTATTACCACCGCTAATCCTAATCCCGTTTA
103	GGCCATTTTCTTTGTTCCGCAAAATGATAATGTTAACTTTTAAAATTAATAACGTTCGGGCAAAG
105	TGGCCCACTTATATCAACCCTCCGGTAAATTCAGAGATCCGGTGG
106	GAGGCGGTCTGCGCTTTCCATTCTGACCTATTCCGGGCTATGCCCCA
107	AGGCTACTGAGTACGGTGATACGCTTTAATGAGGATTGTTCTGAG
118	ATATTTTAAAAAGTAAAAG
119	ATATGTTGTGAAATTAACTAAAATTCCTTTTGGAGCCTTTGTACTG
120	CGCCCATTTTTGGAGATTTTCAAATTTAGGTCAGAAGACCACCTTT
121	CGTTTGCTGTATTTCAAACCATTAACGTGAAAAAATTATTGTCGAA
140	TGAGACGTATAATCGCTGGGGGTCAAAGATGAGTGATGTATTCTTTTGCCTCTTTCGTTTTAGGT
142	GTGTAGAAACCGATACAATTAATTTTCTCGCGTTCTTTTCTATTG
143	TATGAATTTGTCTTGCGATTGGATTTCGAAAGCAAGCTGATTGTTTA
144	CCCCATTTTAAGAAATTCACCTGCATCAGCATTTACTTTGTCTT
Closure Set 4	**Sequence**
8	TTAGGGTTCCAGAAAGCCTC
79	TCGGGGAGGCGGATAAGTGCCGATACATGGCTTTTGACCCTTATT
80	CATAGCCTGATACAGGAGTGTACGGTTTTGCTCAGTACCGTTGCT
81	TAAACGGGATTAGGATTAGCGGTGGTAATAAGTTTTAAGCGCGTTT
103	CTTTGCCCGAACGTTATTAATTTTAAAAGATAACATTATCATTTTGCGGAACAAAGAAAATGGCC
105	CCACCGGATCTCTGAATTTACCGGAGGGTTGATATAAGTGGGCCA
106	TGGGGCATAGCCCGGAATAGGTCAGAATGGAAAGCGCAGACCGCCTC
107	CTCAGAACAATCCTCATTAAAGCGTATCACCGTACTCAGTAGCCT
118	CTTTTACTTTTTAAAATAT
119	CAGTACAAAGGCTCCAAAAGGAATTTTAGTTAATTTCACAACATAT
120	AAAGGTGGTCTTCTGACCTAAATTTGAAAATCTCCAAAAATGGGCG
121	TTCGACAATAATTTTTTCACGTTAATGGTTTGAAATACAGCAAACG
140	ACCTAAAACGAAAGAGGCAAAAGAATACAACACTCATCTTTGACCCCCAGCGATTATACGTCTCA
142	CAATAGAAAAGAACGCGAGAAAATTAATTGTATCGGTTTCTACAC
143	TAAACAATCAGCTTGCTTTCGAAATCCAATCGCAAGACAAATTCATA
144	AAGACAAAGTAAATGCTGATGCAGGTGAATTTCTTAAAATGGGG
82	TAGCGTCAGACTGTACGGGGTCAGTGCCTTGAGACTCCTCAAGAGAATAGGGG
83	TTATGATATTAAGAGGCTGAGTAACAGTG
86	AGCGTTTCCGACTTGCGGGAGGTTTTGAAGCCTTAAATATTTTCGGTAGTCAT
87	AAGCGATCATCGGCCAAGATTAGTTGCTATTTTGCACCCA
102	TTACAAACAATTCGACAATAAC
104	CTGTCTGAATTATCATCATATT
108	AGGCGGGAGGTTTAGTACCGCTATTCACAAACAAATACGCCACCCTCAGAGC
109	TTGGCCTTGACACCCTCAGAACCGCCGATGAC
110	TTTTCATCGTAGGAATCATTACCGCCCGCCACCCCGTGT
111	CCTCAGAGGCCCAATAGCAAGCAAATCAGATATAGAAGGAGAGCCATGCTAG
122	ATTACGCAGTATGTTCGACCGTGTGATAAATTAAAGGAATTGCGAATACGGGA
123	CAGTCGAGAAAGGAACAACAAGGCGTTAA
139	TAAAATACGTAATGCCCCCCCC

141	GCTCGAGTACAACGGAGATTTG
145	CAGCTTGATACCGATAGGTTATATAACTATATAGGGCGACATTCAACCGGCAG
146	GCTTAGGTTGGTTGCGCCGACAATGATGCTTC
147	CCAACGTGGTTTACACACCCTGAACAAAGTCAGAGGGTAATTGAGCGTCACAAT
148	CGCATTAGACGGGAGAATTAACTGACAGCGCCAGCACGT
151	TCGCACTAGAAAATATAGCTATCTTACCGAAGCCCTTTTT
152	GTCTTGAAAAGAAAATAATAAGAGCAAGAAACAATGAAATAGCAACATACAT
85	AGATGAATAAGTTTATTTTGCTAATATCAGAGAGATAAAAAAAGGCGTTTTAGCGAACC TCGCCATCTTTTCATA
112	CGACTGAAATCACCGGAACCCTTATCCGGTATTCTAAGAAAAAACAAGAATTGAGTTAA GCCCACGCAAAGACACCA