DNA ORIGAMI SCAFFOLD DEVELOPMENT FOR DIGITAL NUCLEIC ACID MEMORY

by

Sarah Elizabeth Kobernat

A dissertation

submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy in Biomolecular Sciences

Boise State University

December 2022

© 2022

Sarah Elizabeth Kobernat

ALL RIGHTS RESERVED

BOISE STATE UNIVERSITY GRADUATE COLLEGE

DEFENSE COMMITTEE AND FINAL READING APPROVALS

of the dissertation submitted by

Sarah Elizabeth Kobernat

Dissertation Title: DNA Origami Scaffold Development for Digital Nucleic Acid Memory

Date of Final Oral Examination: 20 October 2022

The following individuals read and discussed the dissertation submitted by student Sarah Elizabeth Kobernat, and they evaluated the student's presentation and response to questions during the final oral examination. They found that the student passed the final oral examination.

Eric J. Hayden, Ph.D. Chair, Supervisory Committee

Daniel Fologea, Ph.D. Member, Supervisory Committee

Lisa Warner, Ph.D. Member, Supervisory Committee

The final reading approval of the dissertation was granted by Eric J. Hayden Ph.D., Chair of the Supervisory Committee. The dissertation was approved by the Graduate College.

DEDICATION

For my mother, Barbara Ann Kobernat. Without her love, support, guidance, and high expectations I would have never made it this far.

ACKNOWLEDGEMENTS

First, I would like to thank my family starting with my mother. She has been there every step of the way and always encouraged and pushed me. My grandparents, Jerry and Sandy Monson, who have been the best grandparents anyone could ever ask for. My sisters, Rebecca and Rachel, who were my first best friends and will be forever. My brother and his beautiful growing family. I cannot express how much my friendships with Jessica, Alice, and Tawni have meant throughout the years.

Next, I would like to thank my advisor Eric Hayden for giving me a chance and never doubting me. He has been completely supportive and encouraging throughout my graduate career. He guided me into a new, cutting-edge field and gave me the space to try and sometimes fail and try again until I succeeded. I sincerely appreciate everything he does for his students. In addition, there have been many members of the Hayden lab that have provided support along the way including Steven Burden, Clémentine Bohachek-Gibard, Stephanie Hudon, Jessica Roberts, Gianluca Peri, Jermey Herrara-Gutierrez, Ben Balzer, Amanda Wolf. I want to thank my bonus mentor Luca Piantanida and the whole NAM team for all of their help. I would also like to thank the members of my supervisory committee, Daniel Fologea and Lisa Warner, for their support and guidance along the way.

Finally, I would like to acknowledge all of the women in STEM who have paved the way for me to do what I love. My STEM education started very young with my mother who is a chemist and a life-long learner. In high school (and throughout my life),

Beth Lipton showed me how far I could go with my education and research. I also want to thank the many female professors who inspired and encouraged me along the way including Diana Cermak, Janet Kirkley, and Kristen Mitchell. I hope you all know how much you touched my life. Thank you.

ABSTRACT

Recently, DNA nanotechnology has emerged as a promising and rapidly expanding method to utilize nucleic acids as a nanoscale building material. DNA origami is a major structural application of DNA nanotechnology, using DNA to construct two- and three-dimensional shapes. These structures have been employed for a variety of uses including DNA data storage. DNA is a promising material to address the impending shortage of silicon-based storage as data demands increase. There are many sequence-based methods of data storage, but digit Nucleic Acid Memory (dNAM) uses DNA origami as a breadboard and is read by super-resolution microscopy instead. dNAM uses DNA origami to spatially position DNA probe sequences in a matrix arrangement that can be read by DNA-PAINT. The prototype used 15 different origami structures to successfully encode and read “Data is in our DNA!\n”.

The dNAM prototype showed the feasibility of using DNA origami as a breadboard, however, the origami’s size limits data capacity and reading efficiency. In chapter 2, we engineered a larger DNA origami rectangle for dNAM. First, we designed a larger node, with an 8x10 matrix of potential data points, a 67% increase from the dNAM prototype. To construct this larger structure, we designed, cloned, produced, and tested a large, custom ssDNA scaffold. With this scaffold, we successfully folded larger origami as confirmed by AFM, and showed the correct positioning of DNA data point probes by DNA-PAINT. This larger structure enabled a 67% increase in the number of data points per origami, which allows for an 80% increase in the number of bits/node when encoding

data. This larger node supports the scaling of dNAM, and will allow for more efficient production and reading.

To take advantage of recent advances in array-based oligonucleotide synthesis, in chapter three we explore the use of pooled staples for dNAM. First, we tested the performance of pooled staples compared to individually synthesized staples using the original dNAM node with the M13mp18 scaffold. We showed that both sets of staples performed equally well in terms of folding origami, and arranging the matrix of data points. Next, we tested the formation of multiple origami structures using orthogonal scaffolds in the context of mixed pools of oligos. We compared the ability of two different scaffolds to fold into the appropriate origami with individual and mixed sets of staple strands. We showed that origami could be folded successfully with either one scaffold and both sets of staples ("random access") or both scaffolds and both sets of staples ("one-pot synthesis"). Finally, we designed multiple scaffolds that use orthogonal sets of staple strands and analyzed their orthogonality. Together, these results move dNAM towards taking advantage of pooled oligos, which will enhance scalability and efficiency. All moving dNAM towards real world applications.

TABLE OF CONTENTS

DEDICATION iv

ACKNOWLEDGEMENTS v

ABSTRACT vii

CHAPTER 1: INTRODUCTION 1

DNA Nanotechnology 1

History of DNA Origami 2

DNA Origami Design and Assembly 2

DNA Origami Applications 5

DNA data storage 6

Sequence Based DNA Data Storage 6

Not Sequence Based DNA Data Storage 7

DNA-PAINT primer 8

Introduction to dissertation chapters 9

CHAPTER 2: LARGE CUSTOM-SIZED DNA SCAFFOLD AND ORIGAMI STRUCTURE DESIGNED FOR SCALABLE NUCLEIC ACID MEMORY 12

Abstract 12

Acknowledgements 21

Supplemental Information 22

Materials and Methods 22

Primers for PCR amplification of the insert and insert sequence from pScaf-3 26

PSCAF-11054.1 scaffold sequence (verified by sanger sequencing) 27

Staples used for PSCAF-11054.1 rectangular origami 36

Appendix 53

CHAPTER 3: TAKING ADVANTAGE OF ARRAY-BASED OLIGO SYNTHESIS TO ADVANCE DNAM 60

Abstract 60

Introduction 61

Materials and Methods 64

Origami synthesis 64

Atomic Force Microscopy 64

Super Resolution Microscopy sample preparation 65

DNA PAINT Imaging 65

Data Analysis 66

Results 67

M13 Comparison of Staple Performance 67

One-Pot Multi-Origami Synthesis 69

Orthogonal Scaffolds Designs 70

Discussion 75

Appendix 78

REFERENCES 87

LIST OF FIGURES

Figure 2.1 Larger origami design and production scheme. (A) Physical dimenstions of the of the prototype 6x8 dNAM data matrix and the new larger 8x10 data matrix. White circles represent sites with protruding data strands (1) and black circles represent sites without data strands (0). (B) Cloning and scaffold production plan. Restriction based cloning adds 974 bp (yellow) to pScaf10080.1 to create a dsDNA phagemid termed pScaf-11054.1. The phagemid is used to transform E. coli, and co-infection with helper phage (green) results in single-stranded scaffold DNA. Purified scaffold DNA is combined with ssDNA staple strands to synthesize a large rectangular origami (Larger dNAM node) that spatially orients DNA data strands (red) on the surface to make the pattern of a specific data matrix. 16

Figure 2.2 dsDNA phagemid and ssDNA scaffold validation by 1% agarose gel electrophoresis. (1) pScaf-10080.1 dsDNA. (2) newly cloned pScaf-11054.1 dsDNA. (3) ~10KB ssDNA produced from pScaf-10080.1. (4) ~11KB ssDNA produced from pScaf-11054.1. (5) pScaf-11054.1 dsDNA phagemid treated with S1 nuclease. (6) ~11KB ssDNA treated with S1 nuclease (degraded).. 18

Figure 2.3 AFM and DNA-PAINT imaging of origami structures. (A) An individual 6x8 dNAM origami synthesized with M13mp18. In this structure, all data sites are “1”. (B) An individual 8x10 dNAM origami synthesized with the new pScaf-11054.1 scaffold. In this structure data sites are encoded as “1” or “0” at random by mixing both sets of staple strands equimolar. (C) Averaged DNA-PAINT image of an individual origami synthesized with pScaf-11054.1. All data sites are “1” in this dNAM origami structure. Images were collected separately but are shown at approximately the same scale.. 19

Figure 2S.1. Gel electrophoresis of DNA origami synthesized with the pScaf-11054.1. This origami was extracted from the gel and and analyzed by AFM. 51

Figure 2S.2. Larger field of view image of several large origami rectangles synthesized using pScaf-11054.1... 52

Figure 2S.3. Larger field of view image of several large origami rectangles synthesized using pScaf-11054.1... 52

Figure 2A.1 PSCAF-11054.1 8x10 matrix encoding scheme. 54

Figure 2A.2 The nodes to encode "Data is in our DNA!\n" using the PSCAF-11054.1 node with an 8x10 matrix. 55

Figure 2A.3 PSCAF-11054.1 node 0- DNA-PAINT on the left shows an averaged image of PSCAF-11054.1 node 0. Right- is the pattern that should be encoded in node 0. 57

Figure 3.1 Gel comparing M13 origami synthesized with pooled staples (left) compared to individually synthesized staples (right). 67

Figure 3.2 AFM comparing M13mp18 origami made with pooled staples (left) or individually synthesized staples (right). 68

Figure 3.3 DNA-PAINT comparing M13mp18 origami made with pooled staples (right) or individually synthesized staples (left). 68

Figure 3.4 Random access and one-pot synthesis (A) Gel electrophoresis of origami from m13mp18 and a custom 11,054 nt scaffold. Origami bands were observed with one scaffold and one pool of staples (individual synthesis), two mixed pools of staples and a single scaffold (random access) and both staple pools and both scaffolds (one-pot synthesis). (B) AFM characterization confirmed two different sized DNA origami in the same “one-pot” synthesis. 70

Figure 3.5 Intra-scaffold sequences duplicates based on fragment length for 6 scaffolds generated by Scaffold Smith (Seq1-Seq6) compared to M13mp18. 73

Figure 3.6 Sequence alignments to compare inter-sequence overlap using Benchling alignment program. Highlighted is the longest overlap found..................74

LIST OF ABBREVIATIONS

AFM	Atomic force microscopy
DNA	Deoxyribonucleic acid
dNAM	digital Nucleic Acid Memory
DNA-PAINT	DNA point accumulation for imaging in nanoscale topography
KB	kilobases
nt	nucleotide
ssDNA	single stranded deoxyribonucleic acid
SRM	super-resolution microscopy
TEM	transmission electron microscopy
TIRF	Total internal reflection fluorescence

CHAPTER 1: INTRODUCTION

DNA Nanotechnology

Recently, DNA nanotechnology has emerged as a promising and rapidly expanding method to utilize nucleic acids as a nanoscale building material[1]. Owing to Watson-Crick-Franklin base pairing, DNA can be used as a highly predictable and programmable building material for self-assembling nanostructures[2]. In addition, DNA is an appealing building material because of its longevity and stability[3]. While this dissertation will focus on using DNA origami for data storage, here we will discuss the basics of DNA nanostructures, their history, design, and uses. In addition, I will give an overview of DNA-based data storage and introduce DNA point accumulation for imaging in nanoscale topography (DNA-PAINT) as a method to image DNA origami.

DNA nanotech can be divided into structural and dynamic subcategories[4]. These can overlap greatly, but generally structural DNA nanotechnology results in a static, equilibrium product, while dynamic DNA nanotechnology utilizes changes to DNA structures in response to physical or chemical stimuli. DNA origami is a major structural application of DNA nanotechnology, using DNA to construct two- and three-dimensional shapes. These structures have been employed for a variety of uses ranging from tissue engineering and nanomachines to medical delivery devices, and data storage systems[5,6,7,8].

History of DNA Origami

DNA origami has been conceptualized since the early 1980's, when Nadrian Seeman, who is widely credited as the first to recognize the potential of DNA nanotechnology, theorized that DNA origami can be used for self-assembling nanostructures by harnessing its predictable base-pairing[9]. In 1991, Seeman made a cube out of DNA utilizing this technology. Seeman's 1993 paper, "Symmetric immobile DNA branched junctions" demonstrated ground breaking fixed DNA double crossovers that would go on to enable much of the following work in DNA nanotechnology[10]. A little over a decade later, in 2006, Paul Rothemund's seminal paper, "Folding DNA to create nanoscale shapes and patterns" showed the first complex DNA-based nanostructures that went well beyond the cubes and octahedrons that were previously produced[11]. In addition, he introduced a "one-pot method" to produce DNA origami using one long strand folded by many short strands or "scaffolded origami". In 2009, DNA was used to fold complex 3D shapes and an innovative computer program, caDNAno, was introduced to facilitate the design process[12,13]. These foundational works opened the door for many to explore the potential of DNA origami.

DNA Origami Design and Assembly

There are several different models of DNA origami fabrication, but it generally starts with designing the shape and size of the structure, which can range from sub 20 nm to large multimeric structures weighing billions of daltons[14,1516]. There are several folding methods that can be chosen based on the shape and density of the design and/or user preference. The main methods are scaffolded, wireframe, and single-stranded origami[11,17,18]. Scaffolded origami, the most common, utilizes one long ssDNA scaffold

and many shorter staple strands to form parallel DNA helices in a tightly packed structure. Wireframe or multi-stranded origami, on the other hand, use many short strands and can form up to six vertices designs. However, this method requires exactly equal ratios of all parts or there will be many partially formed structures that will require significant purification. Single-stranded origami are composed of one long single strand and very few or no shorter staples strands. This structure is formed by the single strand interacting and hybridizing with itself. We will be focusing mainly on scaffolded DNA origami because it is the basis for the research contained herein.

As previously mentioned, scaffolded DNA origami relies on the hybridization of one long ssDNA "scaffold" folding with many shorter, complimentary "staple" strands combined in a thermal annealing reaction. The most widely used scaffold is based on the M13 bacteriophage's circular ssDNA genome, however there are many ways to also make custom scaffolds[19–22]. Staples must then be designed based on the sequence and path of the scaffold with multiple binding regions that hybridize with different sections of the scaffold to bring together otherwise separated regions at crossover junctions[23]. This is a tedious and daunting process to do by hand, and while it has been done, now there are programs to facilitate this process.

CaDNAno is a commonly used program with a graphical user interface to facilitate the design of DNA origami structures[13]. This program allows the user to design the shape of the origami based on the number of helices (in either a square lattice or honeycomb lattice), the length of the helices, and the DNA path, which is the routing of the ssDNA scaffold through the structure. In addition, the user can define crossovers, breaks, and loops. Once the shape is created, the program will determine staple sequence

based on the selected scaffold. Recently, scadnano was introduced as an entirely web-based version of the program[24]. In addition to design, there are several programs for DNA origami modeling. These include CanDo, a web based coarse grain mechanical strain, and OxDNA, a more recently developed rigid-body simulation[25,26]. OxDNA has the options to show more detail including hydrogen bond occupancy, RMS fluctuation structure, and angle between each duplex.

While there are several programs to design DNA origamis structures, there is still a lot unknown about the intricacies of DNA origami design in terms of what factors may impact origami performance. The National Institute of Standards and Technology (NIST) put out a thorough guide to DNA origami design[27]. One design feature that is often taken advantage of for DNA origami is its inherent addressability, which allows for further modification and functionalization of the origami in specific locations/arrangements, allowing one to use it as a "breadboard".

Regardless of design, DNA origami is generally assembled the same way. Briefly, DNA origami is folded by combining the scaffold and an excess of staple strands in a solution with a high concentration of Mg^{2+} to facilitate folding (Mg^{2+} screens inter-helical repulsion, and stabilizes crossovers)[23,28,29]. The mixture is heated to a high temperature, often 80-90°C, and then slowly cooled to ensure stringent annealing conditions. It has been shown that this process is usually, depending on the structure and sequence design, very robust. DNA origami can even be folded with a Bunsen burner or hotplate, without Mg^{2+}, or at constant temperature[29,30,31]. Folded origami is usually purified by gel electrophoresis (or another method) and imaged using atomic force microscopy (AFM) or transmission electron microscopy (TEM)[23].

DNA Origami Applications

DNA origami and DNA nanostructures have been applied to a huge number of applications from biomedical, to nano-mechanical, to data storage. Simple tetrahedral shaped DNA nanostructures have been used for biomedical applications including tissue engineering for neuronal, skin, bone, and others [5]. Other origami shapes have been used for drug delivery and tumor therapy, as they can be used to encapsulate drugs, be targeted, and have selective pay load delivery[32]. For example, there are numerous studies using origami structures loaded with doxorubicin to treat cancer[33,34]. In terms of material science and engineering applications, origami structures can be used to make nanomachines and self-assembling nanorobots with linear, rotational, and reciprocating movements that can even form molecular motors[35,36]. In addition, DNA tweezers have been developed that open and close, and can capture, hold and release an object triggered by changes in pH[37,38].

Very recently, DNA origami Has even been used as a DNA-based data storage material[39]. Digital Nucleic Acid Memory (dNAM) uses DNA origami to position single-stranded DNA molecules in a matrix pattern extending from the surface of the origami. The presence or absence of each ssDNA molecule at a matrix position is used to encode binary data. When present, the ssDNA molecules hybridize to fluorescently labeled DNA molecules ("imager strands") which is then optically read using a super-resolution microscopy approach called DNA-PAINT. This super-resolution fluorescence microscopy approach allows resolution of binary data positioned nanometers apart, creating a data storage material with extremely high arial density. This data storage

technology is the primary motivation of this dissertation, and more details will be described in the following chapters.

DNA data storage

The amount of data generated daily is rapidly growing, and is predicted to out-pace the growth in silicon supplies and then quickly exceed the material resources needed for current storage methods[40–42]. One promising alternative for a digital data storage material is DNA. Due to its durability, information density, and low energy requirements for production and storage, DNA has started to be developed as a storage medium over the last decade[43]. The majority of research has used sequence-based methods to read data that has been encoded in the order of nucleotides in DNA sequences.

Sequence Based DNA Data Storage

Sequence based DNA data storage involves encoding the target data into one or more DNA sequences using chemical synthesis of DNA oligonucleotides that are stored. To read the data, the DNA must be sequenced, and then decoded according to the encoding scheme. There are many different encoding/decoding schemes which usually include error correction and/or other redundancy. Two popular methods of error correction worth noting are fountain code and Reed Solomon code, which can also be used in combination.

DNA-based data storage systems have achieved file size up to ~200MB and the highest data density achieved so far is 1.98 bits/nt[44,45]. This is nearing the potential data density limit of two bits per nucleotide because there are four distinct nucleotides that could each represent a different combination of two binary digits (e.g. A=01, T=10, C=00, G=11). However, there are some draw backs to sequence-based DNA data storage

because it depends on accurate synthesis, proper storage and handling, equal sampling (if a data is stored in a mix of strands it is often subject to PCR bias), and accurate sequencing (and/or error code that can handle insertions/deletions, and base substitutions)[46,47].

Not Sequence Based DNA Data Storage

There are several other types of DNA data storage that do not encode data directly into the sequence of the DNA, but use it in a variety of ways[48]. For example, there is a rewritable system that uses DNA hairpins that are read when passed through a nanopore with 3nm resolution[49]. There are also multiple systems that use DNA modifications to encode data on DNA like the DNA punch card that uses nicks in dsDNA to encode data, or an encrypted system that used biotin-streptavidin to make a braille-like pattern[50,51]. There are some advantages and disadvantages to these types of systems that do not encode into DNA sequence. A disadvantage is that they generally do not approach the 2 bit/nt limit possible for sequence-based encoding. An advantage is that these systems can be much less sensitive to mutations, given the same data output even when one or two mismatches are present. Finally, reading with methods like AFM and fluorescence microscopy decouple the technology from the advancements in sequencing technology. While this does not take advantage of the astonishing advancements of DNA sequencing, it also means that advancements in data storage are not dependent upon advancements in sequencing platforms.

Digital Nucleic Acid Memory (dNAM) is a DNA-based data storage method that is not sequencing technology dependent and will be the primary technique motivating the work in this dissertation. dNAM was developed by researchers at the Nucleic Acid

Memory Institute at Boise State University[39]. In the currently demonstrated prototype, binary data is encoded in multiple DNA origami structures each with a different 6x8 matrix pattern of fluorescent signals. The data encoding uses a multilevel error correction scheme similar to 2-dimensional barcodes. The pattern is produced on a 2D rectangular DNA origami by extending the staple strands (1) or not (0) at the addressable locations corresponding to positions within the 6x8 matrix. To Read the data, fluorescently labeled imager strands, which are complimentary to the staple extensions, are added to the origami and imaged using DNA-PAINT on a TIRF microscope. Origamis are averaged to get a super resolution image of the 10nm apart points on each matrix and decoded with a custom algorithm. As proof-of-concept, "Data is in our DNA!\n" was encoded in 15 different origami nodes and read using DNA-PAINT. The message was successfully read and the error correction scheme was shown to be able to overcome both missing data points on individual origami, and completely missing origami.

DNA-PAINT primer

DNA points accumulation for imaging in nanoscale topography (DNA-PAINT) is a method of super-resolution microscopy (SRM) first introduced by Ralf Jungmann in 2010[52]. DNA-PAINT uses the transient binding of imager probes to target oligonucleotides to observe structures below the diffraction limit of light[53]. "Imager" strands are short oligonucleotides, conjugated to a fluorophore, that are complimentary to the target "docking" strand. Because these strands are short and complimentary, they will hybridize and unhybridize at a certain rate to create a blinking effect. Due to the predefined characteristics of the DNA hybridization between the imager and docking strands, the resulting blinking can be decoupled from dye photophysics, allowing for sub

10 nm potential localization precision[54,55]. This precision combined with the exact addressability of DNA origami makes DNA-PAINT an excellent method to probe DNA origami structures.

DNA origami structures are often imaged and analyzed using atomic force microscopy (AFM) or TEM because they have the resolution necessary to resolve a structure on the nanometer scale[56]. However, DNA-PAINT and DNA origami are inherently linked: DNA-PAINT was first developed on a DNA origami platform, and continues to be optimized and advanced using DNA origami. Conversely, DNA-PAINT is extremely useful to analyze DNA origami. DNA-PAINT has been used to assess the absolute addressability of DNA origami and to probe the origami structure at the staple strand level[55,57]. On the other hand, DNA origami has been used to test microscope performance with DNA-PAINT[58]. DNA origami has also been used to develop methods to increase the speed of DNA-PAINT 100-fold and implement multiplexing[59].

Introduction to dissertation chapters

Chapter 2 focuses on the development of a new, larger rectangular origami structure ("node") for dNAM to expand the efficiency of this approach. We designed a larger origami structure to accommodate 67% more data points per origami structure. Next, we cloned a phagemid to produce a novel ssDNA scaffold that was 11,054 nt in length, which was needed to make the larger structure. This scaffold is named pScaf11054.1 and referred to as the "11KB scaffold", for short. We were able to produce this scaffold with an E. coli/helper phage-based system and use it to form a 116x82nm rectangular origami, with the size confirmed by AFM. Finally, we confirmed the functionality of this new dNAM node using DNA-PAINT to demonstrate an 8x10 matrix

of data points. Moving forward with this node, we will be able to use fewer origami to store data. In addition to dNAM, the larger origami and 11KB scaffold may also be useful for other applications as DNA nanotechnology continues to expand.

We have also started using this new node to encode the prototypical message "Data is in our DNA!\n". With the larger node we will be able to encode the same message in 6 origami instead of the 15 needed for the protype with M13mp18. We designed the encoding scheme and the staples necessary to produce the corresponding origami. We synthesized node 0 with pooled staples and achieved 70% recovery of intended data points. The progress and ongoing work are summarized in the Chapter 2 Appendix. We are still working to improve this rate to successfully decode the message.

Chapter 3 shows the work that we have done to advance the dNAM system so that we may take advantage of new array-based oligonucleotide synthesis in the future. It is now possible to order vast pools of DNA oligonucleotides that come premixed in a single tube. The number of oligos possible far exceeds what is needed for a single DNA origami structure. Our goal is to develop an approach that can use these huge pools of oligos as staple strands for multiple origami structures, and to develop scaffolds that can form accurate origami in the presence of vast amounts of non-specific staple strands. However, the concentration of each oligo in the vast pool is difficult to validate, and the synthesis errors may be different than those in more established chemical synthesis of individual oligos. To test the feasibility of using staples strands synthesized as pools, we first showed that pooled staple strands work to fold origami using the standard M13mp18 scaffold by gel and AFM. In addition, we showed that origami formed with pooled staples or individually synthesized staples performed equally well for DNA-PAINT

which is required to read data in the dNAM system. Next, we showed that we can fold orthogonal structures simultaneously with both M13mp18-based origami and pScaf-11054.1-based origami in the same mixture (one-pot synthesis), and separately but in the presence of non-specific staples (random access). Moving forward, we would like to use multiple larger origami structure with orthogonal scaffolds, so we designed 6 different, non-overlapping scaffolds using the software Scaffold Smith and evaluated their orthogonality. Together, this shows the potential and starts to move dNAM towards being able to take advantage of the current and developing technologies in related fields for a more efficient and high-throughput system.

CHAPTER 2: LARGE CUSTOM-SIZED DNA SCAFFOLD AND ORIGAMI STRUCTURE DESIGNED FOR SCALABLE NUCLEIC ACID MEMORY

Sarah E. Kobernat[1], Amanda Wolf[2], Benjamin Balzer[2], Golam Md Mortuza[3], George D. Dickinson[4], William L. Hughes[4], Tim Andersen[3], Luca Piantanida[4], Eric J. Hayden[2,4*]

[1] Biomolecular Sciences Graduate Programs, Boise State University, Boise, ID, USA

[2] Department of Biological Sciences, Boise State University, Boise, ID, USA

[3] Department of Computer Science, Boise State University, Boise, ID, USA

[4] Micron School of Materials Science and Engineering, Boise State University, Boise, ID, USA

[*] Corresponding author

Abstract

We recently demonstrated DNA-based data storage where DNA strands are spatially oriented on the surface of DNA origami and decoded using DNA-PAINT. In this approach, larger origami structures can improve both data density and reading efficiency, but require large, custom single-stranded DNA scaffolds. Here, we engineer a large DNA origami rectangle using a new custom scaffold. We confirmed the correct origami structure and positioning of DNA data strands using AFM and DNA-PAINT. This larger origami

structure enabled an increase in the amount of data encoded per origami and will support efforts to efficiently scale-up origami-based digital Nucleic Acid Memory (dNAM).

Keywords: DNA origami, data storage, nucleic acid memory, ssDNA synthesis, DNA nanotechnology

The growing need for data storage is quickly outpacing our current memory materials and technologies. Due to increased rates of data generation, data mining, and long-term data storage, the amount of global data is projected to reach 180 zettabytes by 2025[41]. From 2020 to 2030 the compound annual growth rate of data is projected at ~50%, a rate that is predicted to exceed the world's supply of silicon by 2040[60,42]. In addition to storage capacity needs, there is also a growing demand to reduce the total cost of ownership, which includes costs associated with storing data, hardware, writing/transfer/retrieval, management, and energy usage[61]. DNA – the information molecule of life – is a promising solution to address these issues for archival data storage because of its inherent information density, longevity, and low energy demands[40,43].

Recently, digital Nucleic Acid Memory (dNAM) was developed as an alternative to sequence-based DNA data storage[39]. In the prototype demonstration of this approach, binary data was first encoded into abstract 6x8 data matrices. Data was then physically written by synthesizing rectangular-shaped DNA origami structures (called "nodes") that positioned single stranded DNA "data strands" at addressable positions on the surface of the origami to create the specific data matrix encodings of the abstract matrix at the nanometer scale. The presence (1) or absence (0) of protruding DNA data strands

represented the binary data in the 6x8 matrix. To read the data, single-stranded DNA "imager strands" were added. These imager strands were fluorescently labeled and complementary to the data strands, which allowed the pattern of the data strands on the surface of the origami to be identified using a super-resolution microscopy approach called *DNA Point Accumulation for Imaging in Nanoscale Topography* (DNA-PAINT), which is thoroughly described elsewhere[53]. The fluorescent pattern recorded by DNA-PAINT was then decoded with a custom algorithm. As proof-of-concept, the message "Data is in our DNA!\n" was encoded using 15 different origami structures each with a unique patterns of data strands positioned 10 nm apart in the 6x8 matrix (Fig. 1A).

One challenge to scaling up dNAM to encode larger messages is the amount of data encoded in each origami structure. The quantity of data that can be stored in one node can be increased if the size of the rectangular origami is increased. The dNAM prototype used the commercially available 7,249 nucleotide (nt) long single-stranded DNA scaffold isolated from the phage vector M13mp18. The size of this scaffold determined the size of the origami, which resulted in a data density of 16 bits/node. It is important to note that in the dNAM approach, only some data positions in the matrix were used to encode data, while numerous other data positions were used to ensure correct orientation, data indexing, and error correction. A larger origami with more data strand positions could actually use the same number of data positions for orientation, and more efficiently use the rest of the data positions for actual data storage[39]. As a consequence, larger origami structures enable the same amount of data to be stored within much fewer nodes. This also has the advantage of requiring less time to read the data by DNA-PAINT.

However, while a larger origami can be used to more efficiently encode, store and read dNAM data, it cannot be easily synthesized because it also requires the design and production of a larger ssDNA scaffold molecule that is not commercially available like the M13 derived scaffolds. Here, we report the design, production and characterization of a custom, larger origami structure that accommodates 80 data strands per node (compared to 48 for the dNAM prototype). To make this origami, we also produced a ssDNA scaffold that was designed to be long enough to make the desired origami structure (~11 kilobase). To do this we first cloned a double stranded DNA phagemid and then used an E. coli/helper phage-based system to produce the ssDNA. The long ssDNA molecule was used as a scaffold to assemble a120x80 nm rectangular origami that can serve as a larger dNAM node. Finally, we tested the functionality of this new dNAM node using DNA-PAINT to demonstrate the correct formation of an 8x10 matrix that provides 28 bits/node, which is 80% more than the original node (Figure 1).

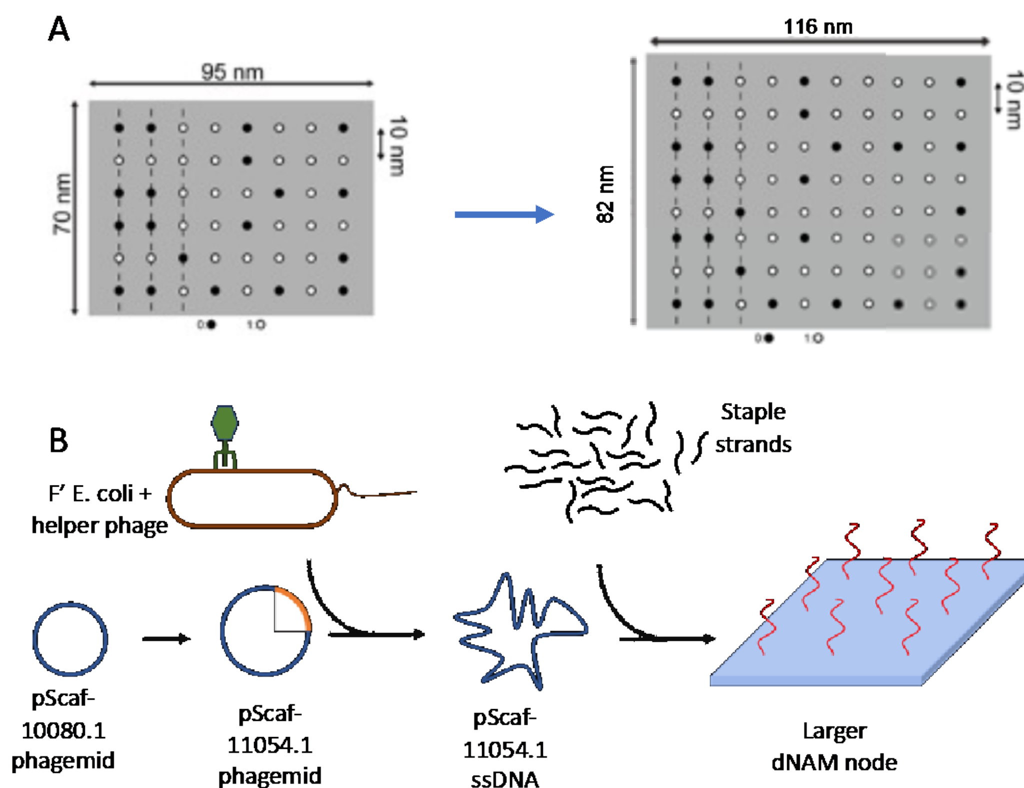


Figure 2.1 Larger origami design and production scheme. (A) Physical dimenstions of the of the prototype 6x8 dNAM data matrix and the new larger 8x10 data matrix. White circles represent sites with protruding data strands (1) and black circles represent sites without data strands (0). (B) Cloning and scaffold production plan. Restriction based cloning adds 974 bp (yellow) to pScaf10080.1 to create a dsDNA phagemid termed pScaf-11054.1. The phagemid is used to transform E. coli, and co-infection with helper phage (green) results in single-stranded scaffold DNA. Purified scaffold DNA is combined with ssDNA staple strands to synthesize a large rectangular origami (Larger dNAM node) that spatially orients DNA data strands (red) on the surface to make the pattern of a specific data matrix.

To make a larger dNAM node, we expanded the prototype rectangular origami design to allow encoding of an 8x10 matrix with 10 nm spacing between data strands[62]. We used the software caDNAno to design a rectangle composed of 32 helices, each 352 bp long[13]. The resulting design was a 116x83 nm rectangle and would require an 11,054 nt long scaffold and 352 DNA oligonucleotide "staples", each 30-32 nt in length. To position data strands on the surface of the origami, specific staple strands were extended with the

sequence 5’-TTGGGAGGA-3’, which has a TT spacer followed by a sequence complementary to the imager probe for DNA-PAINT imaging.

To produce the sequence of the 11,054 nt ssDNA scaffold we first designed and cloned a dsDNA plasmid (phagemid) that can express the scaffold in E. coli. We used restriction enzyme-based cloning to shuffle together DNA from two phagemids, previously reported by others, that produce ssDNA designed for origami scaffolds[63]. These phagemids contain an M13 origin and terminator sequence, and when used to transform E. coli that are also infected with a helper phage, they produce the sequence between the origin and terminator as single-stranded DNA. The other genes of the helper phage also package and export the ssDNA into the media where it can be isolated using common protocols. We started with a phagemid that produces a 10,080 nt scaffold (pScaf-10080.1, Addgene plasmid #111410) and added 982 bp from a second phagemid (pScaf-3024.1, Addgene plasmid #111404) between the M13 origin and terminator sequences. We PCR amplified a 982 nt portion of pScaf-3024.1 with primers that added KpnI and BglII restriction sites to the 5’ and 3’ ends respectively. KpnI and BglII restriction sites are located 8 nts apart on the pScaf-10080.1 plasmid. Both Plasmid and insert amplicon were digested with KpnI and BglII and then ligated together. DH5alpha E. coli were transformed with the ligation product, and plated on ampicillin LB plates to select for the resistance marker that is also in the plasmid, but outside the origin and terminator. Surviving colonies were PCR screened for the 982 bp insert, and colonies with the insert were grown to produce more plasmid. Gel electrophoresis confirmed that the cloned phagemid, which we termed pScaf-11054.1, was larger than the original pScaf-10080.1 phagemid as seen in lanes 1 and 2 of the gel in Figure 2. Sanger sequencing confirmed 11,305 out of 13,809 bp of the phagemid,

with no identified mutations, including the entire 982 bp insert and both ligation junctions (SI).

Next, the pScaf-11054.1 phagemid was used to produce ssDNA. The pScaf-11054.1 phagemid was transformed into 5-alpha F'*Iq* E. coli and grown for 4-6 hours before being infected with helper phage M13KO7 followed by overnight growth[64]. Phage particles were isolated by PEG precipitated and DNA was purified using silica columns. Gel electrophoresis confirmed the presence of ssDNA that was larger than the ssDNA produced by pScaf-10080.1 (Figure 2 lanes 3&4). In addition, single-stranded DNA nuclease S1 was able to degrade the isolated DNA, but not the double-stranded DNA phagemid, confirming that the isolated DNA was single-stranded (Figure 2 lanes 5&6). Sanger sequencing directly from the ssDNA confirmed the presence of the 982 nt insert with no mutations,

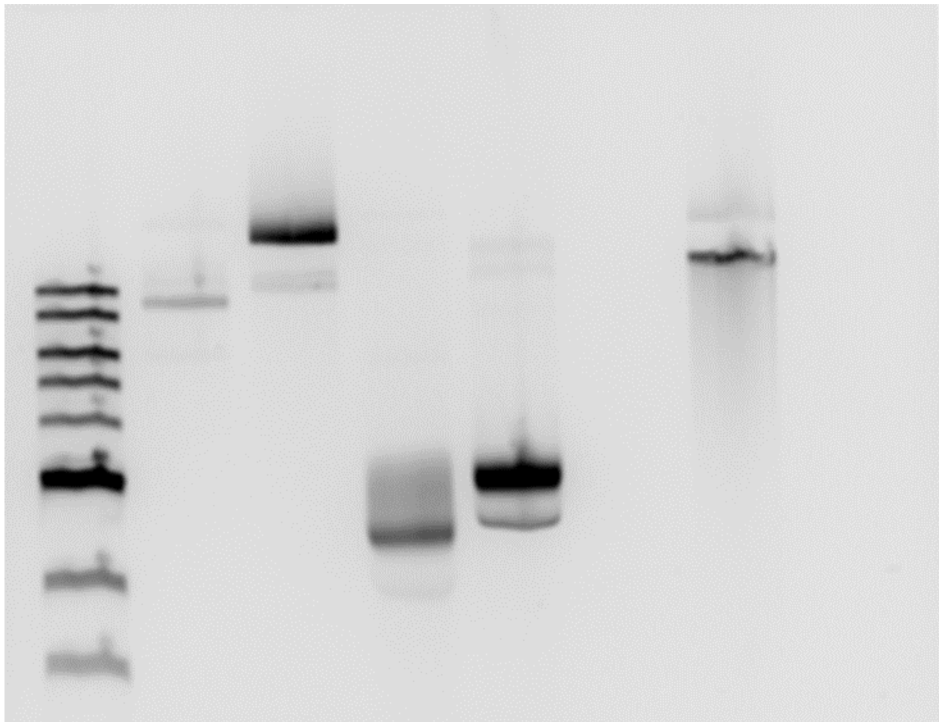


Figure 2.2 dsDNA phagemid and ssDNA scaffold validation by 1% agarose gel electrophoresis. (1) pScaf-10080.1 dsDNA. (2) newly cloned pScaf-11054.1 dsDNA. (3) ~10KB ssDNA produced from pScaf-10080.1. (4) ~11KB ssDNA produced from pScaf-11054.1. (5) pScaf-11054.1 dsDNA phagemid treated with S1 nuclease. (6) ~11KB ssDNA treated with S1 nuclease (degraded).

ruling out contamination of the original "parent" pScaf-10080-1 phagemid. Using this phagemid/helper phage system in E. coli we were able to produce ssDNA with an average yield of 25.4 ng/mL. Together this analysis confirms that the cloned phagemid produces the required ssDNA scaffold needed to synthesize the larger dNAM origami.

To functionally validate the 11,054 nt scaffold, it was mixed with a molar excess of 352 DNA staple strands (SI). The origami was allowed to self-assemble while the mixture cooled slowly from 90°C to 20°C. Following synthesis, a band corresponding to the origami was observed by gel electrophoresis (SI), excised from the gel to extract the origami by squeezing[39], and structurally analyzed by AFM. The results showed numerous well-formed rectangles (SI) with the expected dimensions of 91x113 nm (Figure 3). As expected, the structures contained visible data strands randomly at about half of the sites because the staple strands used for this synthesis contained an equal mixture of molecules with and without data strand extensions.

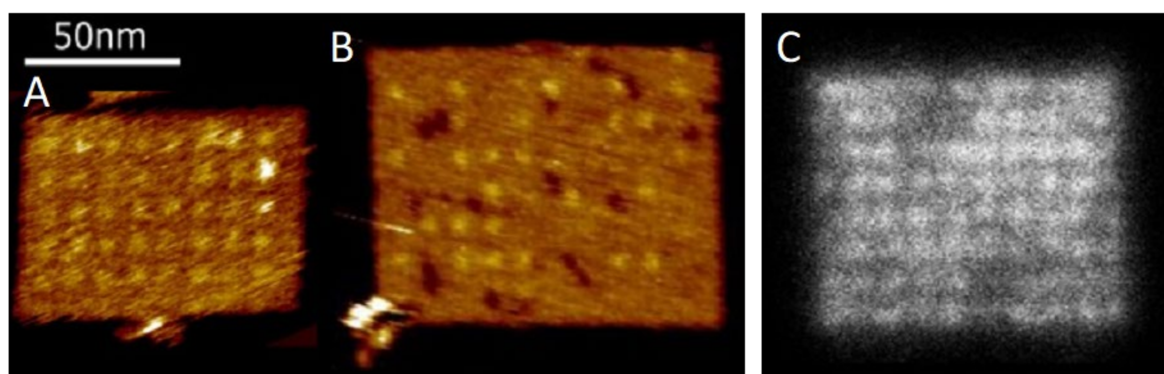


Figure 2.3 AFM and DNA-PAINT imaging of origami structures. (A) An individual 6x8 dNAM origami synthesized with M13mp18. In this structure, all data sites are "1". (B) An individual 8x10 dNAM origami synthesized with the new pScaf-11054.1 scaffold. In this structure data sites are encoded as "1" or "0" at random by mixing both sets of staple strands equimolar. (C) Averaged DNA-PAINT image of an individual origami synthesized with pScaf-11054.1. All data sites are "1" in this dNAM origami structure. Images were collected separately but are shown at approximately the same scale.

Next, the 8x10 matrix of data strands on the origami was validated by DNA-PAINT. Origami was re-synthesized but with DNA staple strands that contained data strand extensions at all 80 points in the matrix. Origami were gel purified, then analyzed by DNA-PAINT as previously described[39]. Images were processed and averaged in a final image, which showed the localization of the 8x10 matrix expected from our design (Figure 3C). This further confirms that the larger origami folds into the correct rectangular shape, but also indicates that the structure positions protruding data strands with the nanometer precision required for data storage in dNAM.

This new larger origami provides more data strands per dNAM node enabling the approach to store more data in fewer origami. For example, using the same encoding scheme as the original prototype, the same message "Data is in our DNA!\n" can be encoded in only 6 origami instead of 15, indicating that it is 2.5 times more efficient at storing data with current data encoding schemes. This improvement makes the writing and storage of data more efficient because fewer origami structures need to be synthesized to store the same amount of data. Also, by reducing the number of structures that need to be imaged, the larger origami also improves the reading of data by reducing the amount of super-resolution data required to decode a given message. In the future, further advancements in DNA synthesis, DNA-PAINT methods, microscopy technology and DNA origami engineering will be needed to continue scaling up dNAM technology[65,59,51].

While the sequence contains some overlap with pScaf-10080.1, it has no similarity with many other scaffolds including M13 derived scaffolds. It may therefore be used to co-synthesize multiple structures with other "orthogonal" scaffolds[19]. Specifically, for dNAM, the simultaneous synthesis of multiple origami structures using orthogonal scaffolds could

provide a significant advancement in the efficiency of synthesis and could also take advantage of staples strands produced in vast oligo pools using cutting-edge DNA synthesis technology[65]. The scaffold structure and its production method reported here open a pathway towards this goal. The custom scaffold used here is 50% larger than the common M13mp18 scaffold. Despite this large size, we were able to produce useful amounts of scaffold with small scale cultures of E. coli. Further advancements to scale-up the production may be needed for many applications, including for efficient data storage using dNAM[66]. The E. coli/helper phage system and DNA origami synthesis only require basic lab equipment making it accessible to many audiences[30]. While the pScaf-11054.1 was designed for a specific rectangular structure, this scaffold might be useful for numerous other DNA origami applications.

Acknowledgements

We acknowledge financial support from the National Science Foundation (Grants no. 1807809 and 2227626) and from the Biomolecular Sciences Graduate Programs at Boise State University.

Supplemental Information

Materials and Methods

Origami Design in Cadnano

A rectangular 80 nm x 120 nm DNA origami structure was designed in cadnano using the square lattice similar to previously described structures (Rafat et al.)[62]. The structure is comprised of 32- 352 nt long helices. This structure requires a 11054 nt scaffold with 352 staple strands ~30 nts long (SI). Staples 10 nm apart in an 8x10 arrangement were elongated to extend from the DNA origami structure with 5'-TTGGGAGGA -3'.

Cloning

Restriction enzyme cloning was used to add an 982nt insert to pScaf-10080.1. The insert was PCR amplified from pScaf-3024.1 (Addgene plasmid #111404) with primers F-5'-ggtaccagtctcacctcaagaagctgtttgctgg-3' and R-5'-agactccttacccaacagctcgcagacggt-3'to added KpnI and BglII primers to the 5' and 3' ends respectively. pScaf-10080.1 and the insert amplicon were digested with KpnI and BglII for 15 min and then ligated overnight at 4C. The newly cloned phagemid was then transformed into DH5alpha E. coli using heat shock. Transformed E. coli were grown in the presence of ampicillin on LB plates overnight. Colonies were PCR screened for the insert using the same primers.

ssDNA Production

A single isolated colony was inoculated in 20mL of LB media with 100ug/mL ampicillin and incubated at 37C for ~4 hrs. Then 20uL of M13KO7 helper phage (NEB) were added to the culture and incubated at 37C for 1hr. Then 50ug/mL of kanamycin was added and culture incubated overnight at 37C. Culture was spun down at 4,000xg for 20

minutes to remove bacteria. Then incubated with 0.2 volumes of 20% PEG/5mM NaCl at 4C for at least 1hr. Samples were spun down at 7,200xg for 1hr. Pellet was resuspended in 800uL TBS, briefly spun again to remove any remaining bacteria, and incubated with 0.2 volumes PEG solution at room temperature for ~5 min. Samples were centrifuged at 25,000xg for 25 min and pellet resuspended in 100uL TE and column purified. ssDNA was run on a 1xTAE 1% agarose gel to confirm size and purity.

Origami Synthesis-

Origami was synthesized using ~10nM scaffold with 30x staples (IDT oPools) in 0.5xTAE with 18mM $MgCl_2$. The origami mixture was fold in the thermocycler [1 min 90 °C, 2 min 80 °C, then from 80 °C to 20 °C over 12 h]. Synthesis mixture was purified on 0.5xTAE 8mM $MgCl_2$ 1% agarose gel at 70V for ~2hrs. Bands were excised, gel crushed, and origami solution collected. Concentration was determined by nanodrop.

Atomic Force Microscopy

AFM analysis was conducted on freshly cut mica. 4 µL of a dNAM origami sample was deposited onto the substrate for 1 min and then 100 µL of AFM buffer added to form a droplet on top of the sample. AFM imaging was performed with a Dimension-FastScan system from Bruker (Billerica MA, USA) set to amplitude modulation mode. Imaging was carried out in liquid with a set-point ratio between the free amplitude and imaging amplitude of ~0.7. The FastScan D cantilever was supplied by Bruker, with a nominal spring constant of 0.25 N/m. Sub-nanometer amplitude was used to image DNA docking strand positions on every origami structure following the method of (10.3791/54924). Tilt correction (line or plane flattening) was performed using WSxM (10.1063/1.2432410)

software package version 5.0 Develop 9.22 (Nanotec Electronica, Madrid, Spain) and a low-pass filter applied to remove noise.

Super Resolution Microscopy Sample Preparation

The formed DNA origami structures were deposited on glass substrates using a microfluidic cell (sticky-Slide, Ibidi GmbH) equipped with inlet and outlet tubes to allow buffer exchange and washing steps between different imager probes solutions. Borosilicate glass slides (25 x 75 and 22 x 22 mm, #1 Gold Seal Coverglass) were sonicated in 0.1% (v/v) Liquinox and nano-pure water (1 min in each) to remove contaminants and dried at 40 ºC for at least 30 min. The coverslips were then, rinsed with methanol and nano-pure water and stored at 40 ºC prior to use. The glow discharge technique previously described by Green[67] was used to deposit DNA origami onto glass coverslips using a PELCO easiGlow™ Glow Discharge Cleaning System (Ted Pella Inc.). Briefly, coverslips that had been cleaned were exposed to glow discharge generated using 20 mAmp at 0.5 mbar for 75 s. For DNA-PAINT analysis, the sticky-Slide flow cell (~50 µL channel volume) was glued to the coverslip and the DNA origami solution deposited by introducing 600 µL of 0.02 nM DNA origami (a mixture of dNAM origami, and sharp triangle origami added as additional fiducial markers, in deposition buffer) into the channel and incubated for 30 min at room temperature. After deposition, the flow chamber was rinsed with 3 mL of deposition buffer (no DNA origami) and mounted on the Fluorescence Microscope.

DNA PAINT Imaging

Immediately before imaging the imager probe solution (imager probe strand in imaging buffer) was supplemented of 5 mM PCA to initiate the oxygen scavenger reaction. DNA origami were imaged below the diffraction-limit of light via DNA-PAINT using an

inverted Nikon Eclipse Ti2 microscope from Nikon Instruments in total internal reflectance fluorescence (TIRF) mode. The images were acquired using an: integrated Perfect Focus System from Nikon Instruments; an oil-immersion CFI Apochromat 100x TIRF objective, with a 1.49 numerical aperture, plus an extra 1.5x magnification from Nikon Instruments; and a 405/488/561/647 nm Laser Quad Band Set TIRF filter cube from Chroma. A 561 nm laser source excited fluorescence from the DNA-PAINT imager strands within an evanescent field extending a few hundred nanometers above the surface of the glass coverslip. The emitted fluorescence was imaged onto the full chip with 512 x 512 pixels (1 pixel = 16μm) using a ProEM EMCCD camera from Princeton Instruments at selected exposure time (50-150-300ms). Images with blinking events were recorded into a stack (typically 40,000 frames per recording) using Nikon NIS-Elements version 5.20.00 from Nikon Instruments prior to processing and analysis.

Data Analysis

After recording a DNA-PAINT stack, the center position of signals (localizations) emitted by imager probes, transiently binding to DNA origami docking strands, were identified using the ImageJ ThunderSTORM plugin[68]. The localizations were rendered and then drift corrected using the Picasso-Render software package, as described by Schnitzbauer et al.[69] When needed, the rendered origami patterns were picked and averaged together using Picasso-Average software package. Data visualization and peak fitting of image data for PSF analysis were performed using OriginPro Version 2019b (OriginLab).

<u>Primers for PCR amplification of the insert and insert sequence from pScaf-3</u>

F- 5’-GGTACCAGTCTCACCTCAAGAAGCTGTTTGCTGG-3’

R- 5’-AGACTCCTTACCCAACAGCTCGCAGACGGT-3’

5’-

CAGTCTCACCTCAAGAAGCTGTTTGCTGGTATCAACTCTGTCTGTTTCGATGA

GAAGTCTAAGCACATTACTGCAATGAAGTCCTTGGAGGGAGAAGTTGTGCCA

TTCAAGAATAAGGTGCCCTTGTCCAATAACGTCGAAACCTGGCTGAACGATC

TGGCCCTGGAGATGAAGAAGACCCTGGAGCAGCTGCTGAAGGAGTGCGTGA

CAACCGGACGCAGCTCTCAGGGAGCTGTGGACCCTTCTCTGTTCCCATCACA

GATCCTGTGCTTGGCCGAACAGATCAAGTTTACCGAAGATGTGGAGAACGCA

ATTAAAGATCACTCCCTGCACCAGATTGAGACACAGCTGGTGAACAAATTGG

AGCAGTATACTAACATCGACACATCTTCCGAGGACCCAGGTAACACAGAGTC

CGGTATTCTGGAGCTGAAACTGAAAGCACTGATTCTCGACATTATCCATAAC

ATCGACGTGGTCAAGCAGCTGAACCAAATCCAAGTGCACACCACCGAAGATT

GGGCCTGGAAGAAGCAGTTGAGGTTCTACATGAAGTCCGACCACACCTGTTG

CGTTCAGATGGTTGACAGCGAGTTCCAGTACACCTATGAGTACCAAGGAAAT

GCCAGCAAGCTCGTTTACACTCCACTCACTGACAAGTGTTACCTCACCTTGAC

ACAGGCTATGAAGATGGGCCTGGGAGGCAACCCATACGGTCCAGCTGGCACT

GGTAAGACAGAGAGCGTTAAGGCACTCGGAGGTCTGCTGGGCAGGCAGGTC

CTCGTGTTCAACTGTGATGAAGGAATCGACGTTAAGTCCATGGGAAGAATCT

TTGTTGGCCTCGTTAAGTGTGGAGCTTGGGGTTGCTTCGACGAGTTCAACAGG

CTGGAGGAATCTGTGCTGAGCGCCGTCTCTATGCAGATCCAGACCATCCAGG

ACGCATTGAAGAACCACAGGACCGTCTGCGAGCTGTTGGGTAAG-3’

PSCAF-11054.1 scaffold sequence (verified by sanger sequencing)

AACAACACTCAACCCTATCTCGGGCTATTCTTTTGATTTATAAGGGATTTTGC
CGATTTCGGGGTACCAGTCTCACCTCAAGAAGCTGTTTGCTGGTATCAACTCT
GTCTGTTTCGATGAGAAGTCTAAGCACATTACTGCAATGAAGTCCTTGGAGG
GAGAAGTTGTGCCATTCAAGAATAAGGTGCCCTTGTCCAATAACGTCGAAAC
CTGGCTGAACGATCTGGCCCTGGAGATGAAGAAGACCCTGGAGCAGCTGCTG
AAGGAGTGCGTGACAACCGGACGCAGCTCTCAGGGAGCTGTGGACCCTTCTC
TGTTCCCATCACAGATCCTGTGCTTGGCCGAACAGATCAAGTTTACCGAAGAT
GTGGAGAACGCAATTAAAGATCACTCCCTGCACCAGATTGAGACACAGCTGG
TGAACAAATTGGAGCAGTATACTAACATCGACACATCTTCCGAGGACCCAGG
TAACACAGAGTCCGGTATTCTGGAGCTGAAACTGAAAGCACTGATTCTCGAC
ATTATCCATAACATCGACGTGGTCAAGCAGCTGAACCAAATCCAAGTGCACA
CCACCGAAGATTGGGCCTGGAAGAAGCAGTTGAGGTTCTACATGAAGTCCGA
CCACACCTGTTGCGTTCAGATGGTTGACAGCGAGTTCCAGTACACCTATGAGT
ACCAAGGAAATGCCAGCAAGCTCGTTTACACTCCACTCACTGACAAGTGTTA
CCTCACCTTGACACAGGCTATGAAGATGGGCCTGGGAGGCAACCCATACGGT
CCAGCTGGCACTGGTAAGACAGAGAGCGTTAAGGCACTCGGAGGTCTGCTGG
GCAGGCAGGTCCTCGTGTTCAACTGTGATGAAGGAATCGACGTTAAGTCCAT
GGGAAGAATCTTTGTTGGCCTCGTTAAGTGTGGAGCTTGGGGTTGCTTCGACG
AGTTCAACAGGCTGGAGGAATCTGTGCTGAGCGCCGTCTCTATGCAGATCCA
GACCATCCAGGACGCATTGAAGAACCACAGGACCGTCTGCGAGCTGTTGGGT
AAGAGATCTAATGCGGCATACCTCAGTGGCGTGGAGTGCAGGTATACAGATT
AATCCGGCAGCGTCCGTCGTTGTTGATATTGCTTATGAAGGCTCCGGCAGTGG

CGACTGGCGTACTGACGGATTCATCGTTGGGGTCGGTTATAAATTCTGATTAG

CCAGGTAACACAGTGTTATGACAGCCCGCCGGAACCGGTGGGCTTTTTTGTG

GGGTGAATATGGCAGTAAAGATTTCAGGAGTCCTGAAAGACGGCACAGGAA

AACCGGTACAGAACTGCACCATTCAGCTGAAAGCCAGACGTAACAGCACCAC

GGTGGTGGTGAACACGGTGGGCTCAGAGAATCCGGATGAAGCCGGGCGTTAC

AGCATGGATGTGGAGTACGGTCAGTACAGTGTCATCCTGCAGGTTGACGGTT

TTCCACCATCGCACGCCGGGACCATCACCGTGTATGAAGATTCACAACCGGG

GACGCTGAATGATTTTCTCTGTGCCATGACGGAGGATGATGCCCGGCCGGAG

GTGCTGCGTCGTCTTGAACTGATGGTGGAAGAGGTGGCGCGTAACGCGTCCG

TGGTGGCACAGAGTACGGCAGACGCGAAGAAATCAGCCGGCGATGCCAGTG

CATCAGCTGCTCAGGTCGCGGCCCTTGTGACTGATGCAACTGACTCAGCACG

CGCCGCCAGCACGTCCGCCGGACAGGCTGCATCGTCAGCTCAGGAAGCGTCC

TCCGGCGCAGAAGCGGCATCAGCAAAGGCCACTGAAGCGGAAAAAAGTGCC

GCAGCCGCAGAGTCCTCAAAAAACGCGGCGGCCACCAGTGCCGGTGCGGCG

AAAACGTCAGAAACGAATGCTGCAGCGTCACAACAATCAGCCGCCACGTCTG

CCTCCACCGCGGCCACGAAAGCGTCAGAGGCCGCCACTTCAGCACGAGATGC

GGTGGCCTCAAAAGAGGCAGCAAAATCATCAGAAACGAACGCATCATCAAG

TGCCGGTCGTGCAGCTTCCTCGGCAACGGCGGCAGAAAATTCTGCCAGGGCG

GCAAAAACGTCCGAGACGAATGCCAGGTCATCTGAAACAGCAGCGGAACGG

AGCGCCTCTGCCGCGGCAGACGCAAAAACAGCGGCGGCGGGGAGTGCGTCA

ACGGCATCCACGAAGGCGACAGAGGCTGCGGGAAGTGCGGTATCAGCATCG

CAGAGCAAAAGTGCGGCAGAAGCGGCGGCAATACGTGCAAAAAATTCGGCA

AAACGTGCAGAAGATATAGCTTCAGCTGTCGCGCTTGAGGATGCGGACACAA

CGAGAAAGGGGATAGTGCAGCTCAGCAGTGCAACCAACAGCACGTCTGAAA
CGCTTGCTGCAACGCCAAAGGCGGTTAAGGTGGTAATGGATGAAACGAACAG
AAAAGCCCACTGGACAGTCCGGCACTGACCGGAACGCCAACAGCACCAACC
GCGCTCAGGGGAACAAACAATACCCAGATTGCGAACACCGCTTTTGTACTGG
CCGCGATTGCAGATGTTATCGACGCGTCACCTGACGCACTGAATACGCTGAA
TGAACTGGCCGCAGCGCTCGGGAATGATCCAGATTTTGCTACCACCATGACT
AACGCGCTTGCGGGTAAACAACCGAAGAATGCGACACTGACGGCGCTGGCA
GGGCTTTCCACGGCGAAAAATAAATTACCGTATTTTGCGGAAAATGATGCCG
CCAGCCTGACTGAACTGACTCAGGTTGGCAGGGATATTCTGGCAAAAAATTC
CGTTGCAGATGTTCTTGAATACCTTGGGGCCGGTGAGAATTCGGCCTTTCCGG
CAGGTGCGCCGATCCCGTGGCCATCAGATATCGTTCCGTCTGGCTACGTCCTG
ATGCAGGGGCAGGCGTTTGACAAATCAGCCTACCCAAAACTTGCTGTCGCGT
ATCCATCGGGTGTGCTTCCTGATATGCGAGGCTGGACAATCAAGGGGAAACC
CGCCAGCGGTCGTGCTGTATTGTCTCAGGAACAGGATGGAATTAAGTCGCAC
ACCCACAGTGCCAGTGCATCCGGTACGGATTTGGGGACGAAAACCACATCGT
CGTTTGATTACGGGACGAAAACAACAGGCAGTTTCGATTACGGCACCAAATC
GACGAATAACACGGGGGCTCATGCTCACAGTCTGAGCGGTTCAACAGGGGCC
GCGGGTGCTCATGCCCACACAAGTGGTTTAAGGATGAACAGTTCTGGCTGGA
GTCAGTATGGAACAGCAACCATTACAGGAAGTTTATCCACAGTTAAAGGAAC
CAGCACACAGGGTATTGCTTATTTATCGAAAACGGACAGTCAGGGCAGCCAC
AGTCACTCATTGTCCGGTACAGCCGTGAGTGCCGGTGCACATGCGCATACAG
TTGGTATTGGTGCGCACCAGCATCCGGTTGTTATCGGTGCTCATGCCCATTCT
TTCAGTATTGGTTCACACGGACACACCATCACCGTTAACGCTGCGGGTAACG

CGGAAAACACCGTCAAAAACATTGCATTTAACTATATTGTGAGGCTTGCATA

ATGGCATTCAGAATGAGGGATCTAGTGATGAAGAATGAACATCCCGCGTTCT

TCCCTCCGAACAGGACGATATTGTAAATTCACTTAATTACGAGGGCATTGCA

GTAATTGAGTTGCAGTTTTACCACTTTCCTGACAGTGACAGACTGCGTGTTGG

CTCTGTCACAGACTAAATAGTTTGAATGATTAGCAGTTATGGTGATCAGTCAA

CCACCAGGGAATAATCCTTCATATTATTATCGTGCTTCACCAACGCTGCCTCA

ATTGCTCTGAATGCTTCCAGAGACACCTTATGTTCTATACATGCAATTACAAC

ATCAGGGTAACTCATAGAAATGGTGCTATTAAGCATATTTTTACACGAATCA

GATCCACGGAGGGATCATCAGCAGATTGTTCTTTATTCATTTTGTCGCTCCAT

GCGCTTGCTCTTCATCTAGCGGTTAAAATATTACTTCAAATCTTTCTGTATGA

AGATTTGAGCACGTTGGCCTTACATACATCTGTCGGTTGTATTTCCCTCCAGA

ATGCCAGCAGGACCGCACTTTGTTACGCAACCAATACTATTAAGTGAAAACA

TTCCTAATATTTGACATAAATCATCAACAAAACACAAGGAGGTCAGACCAGA

TTGAAACGATAAAAACGATAATGCAAACTACGCGCCCTCGTATCACATGGAA

GGTTTTACCAATGGCTCAGGTTGCCATTTTTAAAGAAATATTCGATCAAGTGC

GAAAAGATTTAGACTGTGAATTGTTTTATTCTGAACTAAAACGTCACAACGTC

TCACATTATATTTACTATCTAGCCACAGATAATATTCACATCGTGTTAGAAAA

CGATAACACCGTGTTAATAAAAGGACTTAAAAAGGTTGTAAATGTTAAATTC

TCAAGAAACACGCATCTTATAGAAACGTCCTATGATAGGTTGAAATCAAGAG

AAATCACATTTCAGCAATACAGGGAAAATCTTGCTAAAGCAGGAGTTTTCCG

ATGGGTTACAAATATCCATGAACATAAAAGATATTACTATACCTTTGATAATT

CATTACTATTTACTGAGAGCATTCAGAACACTACACAAATCTTTCCACGCTAA

ATCATAACGTCCGGTTTCTTCCGTGTCAGCACCGGGGCGTTGGCATAATGCAA

TACGTGTACGCGCTAAACCCTGTGTGCATCGTTTTAATTATTCCCGGACACTC
CCGCAGAGAAGTTCCCCGTCAGGGCTGTGGACATAGTTAATCCGGGAATACA
ATGACGATTCATCGCACCTGACATACATTAATAAATATTAACAATATGAAATT
TCAACTCATTGTTTAGGGTTTGTTTAATTTTCTACACATACGATTCTGCGAACT
TCAAAAAGCATCGGGAATAACACCATGAAAAAAATGCTACTCGCTACTGCGC
TGGCCCTGCTTATTACAGGATGTGCTCAACAGACGTTTACTGTTCAAAACAAA
CCGGCAGCAGTAGCACCAAAGGAAACCATCACCCATCATTTCTTCGTTTCTG
GAATTGGGCAGAAGAAAACTGTCGATGCAGCCAAAATTTGTGGCGGCGCAG
AAAATGTTGTTAAAACAGAAACCCAGCAAACATTCGTAAATGGATTGCTCGG
TTTTATTACTTTAGGCATTTATACTCCGCTGGAAGCGCGTGTGTATTGCTCAC
AATAATTGCATGAGTTGCCCATCGATATGGGCAACTCTATCTGCACTGCTCAT
TAATATACTTCTGGGTTCCTTCCAGTTGTTTTTGCATAGTGATCAGCCTCTCTC
TGAGGGTGAAATAATCCCGTTCAGCGGTGTCTGCCAGTCGGGGGGAGGCTGC
ATTATCCACGCCGGAGGCGGTGGTGGCTTCACGCACTGACTGACAGACTGCT
TTGATGTGCAACCGACGACGACCAGCGGCAACATCATCACGCAGAGCATCAT
TTTCAGCTTTAGCATCAGCTAACTCCTTCGTGTATTTTGCATCGAGCGCAGCA
ACATCACGCTGACGCATCTGCATGTCAGTAATTGCCGCGTTCGCCAGCTTCAG
TTCTCTGGCATTTTTGTCGCGCTGGGCTTTGTAGGTAATGGCGTTATCACGGT
AATGATTAACAGCCCATGACAGGCAGACGATGATGCAGATAACCAGAGCGG
AGATAATCGCGGTGACTCTGCTCATACATCAATCTCTCTGACCGTTCCGCCCG
CTTCTTTGAATTTTGCAATCAGGCTGTCAGCCTTATGCTCGAACTGACCATAA
CCAGCGCCCGGCAGTGAAGCCCAGATATTGCTGCAACGGTCGATTGCCTGAC
GGATATCACCACGATCAATCATAGGTAAAGCGCCACGCTCCTTAATCTGCTG

CAATGCCACAGCGTCCTGACTTTTCGGAGAGAAGTCTTTCAGGCCAAGCTGCT

TGCGGTAGGCATCCCACCAACGGGAAAGAAGCTGGTAGCGTCCGGCGCCTGT

TGATTTGAGTTTTGGGTTTAGCGTGACAAGTTTGCGAGGGTGATCGGAGTAAT

CAGTAAATAGCTCTCCGCCTACAATGACGTCATAACCATGATTTCTGGGGATC

TAGCCTGCAGTAATAGAGCGGCCCAGGATGCGGAGTTGGGAAATCACCTGAG

ATGAGATCATGCGCTGTGATGAGCGTGCAATGTTGGCAGGCAGACCGAAGAA

GGATGGCTTATCGTCTTCTGGAATCTTCTCAATGACAGCCCTGTAATCCAGAA

TGCTGCAGCTTTGTGGCAGGCTCACGCTGTAGGGAAAGATGGACTTCTTGTTG

CGCTGATTGAACACGTCAATGACGCTAGAGTTGAAGAATTGCTTCAAATAGC

TCTGCAGGACGCGCAAGTCGAAATAGTTATCGATGCGGCCTCCGTAGATGGC

GTTCTCCAGCAGACCATGGACAAACTCCCATTGCACGTCCTTGGCACCATCA

AAGAGCCTGTCAATGATATTGTAACCGGCCCTCAGGTCGGAGAGAGAGAACT

CGTAGAATTTGGTCCAACCTTGAGGAATGTAGTTGCGCCTCTCTTGGCATGCG

GCGTGAAACCAGGCCAGAGAGAACAGAGCATGTGCGCGGTGAGTGTTATCTT

TCTTAGAGATTTGTTCAGGTGTCCAAGATTCATATGTGCGCATGAGATTCTTC

TTGAGTCCAGGTGGTGATTCATAAGTGATCTTGAGGCTTGACTGGAGGAGGA

TAGGTGTAAAGTTTGGGTGCACTTCGGCTGTCAACCAGAGCCTGAAAGTGTC

TTTGGGTTGCAAGGTGTTCAGTTCCTTCTCCAAGACAGGCAGCCAGGAGACA

ACCAGGTGCAAGTTCTTCAAACACAGCCAGTCACCATTCCTTGCGCATTCCTT

GAGCATTTGGATAGCCAAGTCAGCTTGGCCCTGGCCCATTGCAACCTGGTGG

TAACATTCTCCTGATCTTTCGGCGTTGGCGAGCTCCTGGAGTTCCTGGGAGGG

ATCGGCTCCTGGAGAAATAATGATCAGAATGGGCTCGATTTCGAGTGTCTCCT

TGTACAGCCTCTTGAGATTGAGTGGGAGTGGAGAGACTTCCTTCAAACCCAA

GGTCTTGCAGGCAAAGAGGGCCATAGCTGACTGCAATCTATCTGGCCTGAGC

ACCTGCACGACGAGGATCTGCTGGAACAAGCTGACCTTCTTAGCCAAGATGC

TAGGAAATTCCTGCTCACACATGCTGTTGTTGTAGTAGGTGCGCCACAAAGC

GGCGTCCTCGAAACACAGAGTCTGGTACAGGGAAGGCAGAGCGATCTTCAAG

GTTGCCACAGCCCAGGATCTCTCTTGATCGATCCATGAAGGCAACTGGTCCCT

GATCTTCTGCTGGGAGTCGGCTTTGCGCAACATGTCTCCGACCACCACGCCAG

TGAAGGTATCCCATTCGTTCTCCTGGAACAATTCTGGATGCATACCTCTGACG

AAATGCAGGGCGAACATCAGCTGGTCGGCCTTGAACAGGCAGCGGCAGATGT

ATTCGTACACCATGTGCTGCAGGGAGCTGATCAGGGATTGAATCCTCTGTTCT

GTGTTCTCAGAATCCTGCTTATTCTGGAGTGCCCTTTGAAACAACCTCAGAAA

GGCGGCCAGGGAGAACCTGTACATATTATTGATCTTGGAGAGGTCAGAGATA

ATGAAGTACATCTTTGAAGCAGACTCAGCGAGTGGCAGATATGCGTCGCGCT

CCTGGTCGAGAGAGATTTGCAGCTTATAAGATTCCTTCAGAGACTCTTGGATG

AGGGCGGAAGATGCTTTAGTCTGATTCAGAGACTCGATCAGGTCCTTGTTCTC

CAGGATGTTGCCCTGAGAAGTAGCCAATGTCTCCAGCAGAGATTCTTCGAGT

TTGGCCAGTTGGATCTTCTTATCCTCCTCCTGTTGGAGCAGCTTAGTCTTCTGT

TCTTCCAAATCGGGTTTCTCATGTTGGATGGTGAGGGCCAGGAGCTGGCCTCT

CAGTCCTGAGCGGGTGGTTGTGAAATTAACCTCTGTGACGATGCTGGCAGCG

TCTGGTGGAATGAATGGGTTGGGATTCCTAGTGCTCAGGAACAGGCGAAACT

CCTCGTTATAGTCGATGATCTTATCTCCAATCTGGACCACATAGCGTGGGCCT

TGGGCCACCAAATCTCTGCGCAACAGAGGGTACAGAACTGGCTCCACGCCAT

CCATCTCCTGGATGATCAAGGTCTTGCCGAAGCGGACGGCGAGTTCCAGGGC

AGTGATGAAGTTGGAGTCTTGCTGGTTGATCACTTCCAGCCTGGAATCCTTCA

AGTGAGTCTTCAGCCACTCAGTAGCCTGTGAGCTGGGATCGATGAGGAAAGG
GCAAACGCGGGATTGCAAGATAACCAGTGCGTTCTCGATTGAGAGGTCATCG
CTGGGCAGACCTTCAGACTTCCAGATCAGCTGCTCTGACTCGGTGCAGAGAA
ATCTTCTCAGGTCAAACTTCTCCAGGCCGGCAGACTTGGTCCACTCCTCCAGA
CATGTCTTTCTGAGAGACTCAGGTGCAGCGGACAAGTAGGTGATAAAGGCTG
CGGCCAACTGAGCCCTCTTTGGGAGGGTTGCCAGCTCCTCAGTAATCTCGAC
GACCTGAGCATTCCATCTCTTGTGTTCGCGGTCCGGATCTCGAAGGAAAGACC
TGATGCTTTTCGTGCGCGCATAAAATACCTTGATACTGTGCCGGATGAAAGCG
GTTCGCGACGAGTAGATGCAATTATGGTTTCTCCGCCAAGAATCTCTTTGCAT
TTATCAAGTGTTTCCTTCATTGATATTCCGAGAGCATCAATATGCAATGCTGT
TGGGATGGCAATTTTTACGCCTGTTTTGCTTTGCTCGACATAAAGATATCCAT
CTACGATATCAGACCACTTCATTTCGCATAAATCACCAACTCGTTGCCCGGTA
ACAACAGCCAGTTCCATTGCAAGTCTGAGCCAACATGGTGATGATTCTGCTG
CTTGATAAATTTTCAGGTATTCGTCAGCCGTAAGTCTTGATCTCCTTACCTCTG
ATTTTGCTGCGCGAGTGGCAGCGACATGGTTTGTTGTTATATGGCCTTCAGCT
ATTGCCTCTCGGAATGCATCGCTCAGTGTTGATCTGATTAACTTGGCTGACGC
CGCCTTGCCCTCGTCTATGTATCCATTGAGCATTGCCGCAATTTCTTTTGTGGT
GATGTCTTCAAGTGGAGCATCAGGCAGACCCCTCCTTATTGCTTTAATTTTGC
TCATGTAATTTATGAGTGTCTTCTGCTTGATTCCTCTGCTGGCCAGGATTTTTT
CGTAGCGATCAAGCCATGAATGTAACGTAACGGAATTATCACTGTTGATTCTC
GCTGTCAGAGGCTTGTGTTTGTGTCCTGAAAATAACTCAATGTTGGCCTGTAT
AGCTTCAGTGATTGCGATTCGCCTGTCTCTGCCTAATCCAAACTCTTTACCCG
TCCTTGGGTCCCTGTAGCAGTAATATCCATTGTTTCTTATATAAAGGTTAGGG

GGTAAATCCCGGCGCTCATGACTTCGCCTTCTTCCCATTTCTGATCCTCTTCAA

AAGGCCACCTGTTACTGGTCGATTTAAGTCAACCTTTACCGCTGATTCGTGGA

ACAGATACTCTCTTCCATCCTTAACCGGAGGTGGGAATATCCTGCATTCCCGA

ACCCATCGACGAACTGTTTCAAGGCTTCTTGGACGTCGCTGGCGTGCGTTCCA

CTCCTGAAGTGTCAAGTACATCGCAAAGTCTCCGCAATTACACGCAAGAAAA

AACCGCCATCAGGCGGCTTGGTGTTCTTTCAGTTCTTCAATTCGAATATTGGT

TACGTCTGCATGTGCTATCTGCGCCCATATCATCCAGTGGTCGTAGCAGTCGT

TGATGTTCTCCGCTTCGATAACTCTGTTGAATGGCTCTCCATTCCATTCTCCTG

TGACTCGGAAGTGCATTTATCATCTCCATAAAACAAAACCCGCCGTAGCGAG

TTCAGATAAAATAAATCCCCGCGAGTGCGAGGATTGTTATGTAATATTGGGTT

TAATCATCTATATGTTTTGTACAGAGAGGGCAAGTATCGTTTCCACCGTACTC

GTGATAATAATTTTGCACGGTATCAGTCATTTCTCGCACATTGCAGAATGGGG

ATTTGTCTTCATTAGACTTATAAACCTTCATGGAATATTTGTATGCCGACTCTA

TATCTATACCTTCATCTACATAAACACCTTCGTGATGTCTGCATGGAGACAAG

ACACCGGATCTGCACAACATTGATAACGCCCAATCTTTTTGCTCAGACTCTAA

CTCATTGATACTCATTTATAAACTCCTTGCAATGTATGTCGTTTCAGCTAAAC

GGTATCAGCAATGTTTATGTAAAGAAACAGTAAGATAATACTCAACCCGATG

TTTGAGTACGGTCATCATCTGACACTACAGACTCTGGCATCGCTGTGAAGACG

ACGCGAAATTCAGCATTTTCACAAGCGTTATCTTTTACAAAACCGATCTCACT

CTCCTTTGATGCGAATGCCAGCGTCAGACATCATATGCAGATACTCACCTGCA

TCCTGAACCCATTGACCTCCAACCCCGTAATAGCGATGCGTAATGATGTCGAT

AGTTACTAACGGGTCTTGTTCGATTAACTGCCGCGGATCCACGCGCCCTGTAG

CGGCGCATTAAGCGCGGCGGGTGTGGTGGTTACGCGCAGCGTGACCGCTACA

CTTGCCAGCGCCCTAGCGCCCGCTCCTTTCGCTTTCTTCCCTTCCTTTCTCGCC

ACGTTCGCCGGCTTTCCCCGTCAAGCTCTAAATCGGGGGCTCCCTTTAGGGTT

CCGATTTAGTGCTTTACGGCACCTCGACCCCAAAAAACTTGATTTGGGTGATG

GTTCACGTAGTGGGCCATCGCCCTGATAGACGGTTTTTCGCCCTTTGACGTTG

GAGTCCACGTTCTTTAATAGTGGACTCTTGTTCCAAACTGG

Staples used for PSCAF-11054.1 rectangular origami

Start	End	Sequence
0[47]	1[31]	TGTGAATATTATCTGTGGCTAGATAGTTCAG
0[79]	1[63]	TATTAACACGGTGTTATCGTTTTCTCTTTTCG
0[111]	1[95]	ATTTAACATTTACAACCTTTTTAAAAATGGCA
0[143]	1[127]	GACGTTTCTATAAGATGCGTGTTCATGTGAT
0[175]	0[144]	ATGTGATTTCTCTTGATTTCAACCTATCATAG
0[207]	1[191]	GCTTTAGCAAGATTTTCCCTGTATTATTGTTA
0[239]	1[223]	ATGGATATTTGTAACCCATCGGACGATGAAT
0[271]	1[255]	TTATCAAAGGTATAGTAATATCTTTATGTCCA
0[303]	1[287]	GTTCTGAATGCTCTCAGTAAATAGGCGGGAGT
0[335]	1[319]	TTATGATTTAGCGTGGAAAGATTACACAGGG
1[160]	2[144]	CTAAACAATATGTGTAGAAAATTATCGTTTTT
2[47]	0[48]	AAGTGCGGCAATTCACAGTCTAAATAACACGA
2[79]	0[80]	TGTTTTCACGAATATTTCTTTAAGTCCTTT
2[111]	0[112]	TTTGTTGACATTGGTAAAACCTTCTCTTGAGA
2[143]	1[159]	ATCGTTTCCGCGTAGTTTGCATTAAACAAACC
2[175]	0[176]	CAGAATCGTGAGTTGAAATTTCATGCTGAA

2[207]	0[208]	TGGTGTTATAATGTATGTCAGGTGAAACTCCT
2[239]	0[240]	GCCAGCGCGTATTCCCGGATTAACTTATGTTC
2[271]	0[272]	ACGTCTGTACGGGGAACTTCTCTTAATGAA
2[303]	0[304]	GCTACTGCATAATTAAAACGATGCTGTGTAGT
2[335]	0[336]	GAAGAAATCGTACACGTATTGCATACCGGACG
3[32]	5[31]	GCTCAAACAGCGTTGGTGAAGCGTTGACTG
3[64]	5[63]	AATATTTTGGTGTCTCTGGAAGCAAAACTATT
3[96]	5[95]	CATGGAGCCTGATGTTGTAATTGCCAGTCTGT
3[128]	5[127]	GCTGATGATGCTTAATAGCACCGCAACTCA
3[192]	5[191]	ATCGATGGCCTCAGAGAGAGGCTGGTCATGGG
3[224]	5[223]	AATACACTGGCAGACACCGCTGCCATTACC
3[256]	5[255]	GCCTAAAGCTCCGGCGTGGATAATTGCCAGAG
3[288]	5[287]	ACGAATGTTCTGTCAGTCAGTGCGATTACTGA
3[320]	5[319]	ATTTTCTGCTGGTCGTCGTCGGTGCTGCGC
4[47]	2[48]	AATTGAGGTCTTCATACAGAAAGAGCGTAACA
4[79]	2[80]	GAACATAAAACCGCTAGATGAAGTTAGGAA
4[175]	2[176]	TGCAAAAAGCAGTGCAGATAGAGAAGTTCG
4[207]	2[208]	TATTTCACGCAACTCATGCAATTATTTTTTCA
4[239]	2[240]	CCCCCGACACGCGCTTCCAGCGGATAAGCAGG
4[271]	2[272]	ACCACCGCTAATAAAACCGAGCAACAGTAA
4[303]	2[304]	CAAAGCAGTTGCTGGGTTTCTGTTCCTTTGGT
4[335]	2[336]	ATGTTGCCGCGCCGCCACAAATTTCCAGAAAC
5[160]	6[144]	GTTATCTGGTCACCGCGATTATCTAAGTGAAT
6[47]	4[48]	TATAGTTATAACTGCTAATCATTCTTCAGAGC

6[79]	4[80]	TCATTCTGGACAGAGCCAACACGATGTATA
6[111]	4[112]	GCGGGATGGGAAAGTGGTAAAACTATTTCTAT
6[143]	5[159]	TTACAATACAATGCCCTCGTAATTCCGCTCTG
6[175]	4[176]	TGAGCAGACATCATCGTCTGCCTATCACTA
6[207]	4[208]	AGCGGGCGCATTACCGTGATAACGAACGGGAT
6[239]	4[240]	AGGCTGACCCCAGCGCGACAAAAAGCAGCCTC
6[271]	4[272]	GCCGGGCGCTGGCGAACGCGGCATGAAGCC
6[303]	4[304]	TCGACCGTTGCGTCAGCGTGATGTTTGCACAT
6[335]	4[336]	ATGATTGAAAAATACACGAAGGAGGCGTGATG
7[32]	9[31]	TGTGTCCTACTGACTCCAGCCAATGAGCAC
7[64]	9[63]	GGCATGAGTGGATAAACTTCCTGTAGACTGTG
7[96]	9[95]	TGCGCACCGCAATACCCTGTGTGCTCGATTTG
7[128]	9[127]	ACCGGCAGCTGCCCTGACTGTCGTTTTCGT
7[160]	8[144]	TGACGTCACTAGATCCCCAGAAATACAATGAG
7[192]	9[191]	GATTACTCTCCGCATCCTGGGCCGATGGTGCC
7[224]	9[223]	GCTAAACCGCATGATCTCATCTCATGGTCT
7[256]	9[255]	GGACGCTAGTCTGCCTGCCAACATGCCGCATC
7[288]	9[287]	ATGCCTACATTCCAGAAGACGATACCTGCAGA
7[320]	9[319]	TTCTCTCCATTCTGGATTACAGCTAGCGTC
8[47]	6[48]	CTGTTCCAGTGTGAACCAATACTGCCTCACAA
8[79]	6[80]	TTTAACTGCACCGATAACAACCGAGATCCC
8[111]	6[112]	ATAAATAAAATACCAACTGTATGCGGAAGAAC
8[175]	6[176]	ACTGCAGGTTGTAGGCGGAGAGCTGATGTA
8[207]	6[208]	TTCCCAACCGATCACCCTCGCAAATTCAAAGA

8[239]	6[240]	CATCACAGCCAAAACTCAAATCAACGAGCATA
8[271]	6[272]	CTTCTTCGCCAGCTTCTTTCCCGCTTCACT
8[303]	6[304]	TTGAGAAGCGCAAGCAGCTTGGCCTCAGGCAA
8[335]	6[336]	AGCTGCAGCGAAAAGTCAGGACGCCTTTACCT
9[160]	10[144]	CAATATCATCTCTCCGACCTGAGGTTTTCGTC
10[47]	8[48]	CCTTGATTCCCTGTTGAACCGCTCAATGGTTG
10[79]	8[80]	AGACAATACCCCCGTGTTATTCGTGGTTCC
10[111]	8[112]	GGGTGTGCATCGAAACTGCCTGTTCGTTTTCG
10[143]	9[159]	CCCAAATCTCAAACGACGATGTGGGCCGGTTA
10[175]	8[176]	GAGTTCTCTTGACAGGCTCTTTGCTCTATT
10[207]	8[208]	CTACATTCGCAATGGGAGTTTGTCCAGGTGAT
10[239]	8[240]	GGTTTCACAACGCCATCTACGGAGTGCACGCT
10[271]	8[272]	TCACCGCGTTTCGACTTGCGCGTAGCCATC
10[303]	8[304]	CACCTGAAAAGCAATTCTTCAACTGGCTGTCA
10[335]	8[336]	AAGAATCTTGTTCAATCAGCGCAATGCCACAA
11[32]	13[31]	GGTAGGCTTCGCCGTGGAAAGCTTGTTTAC
11[64]	13[63]	CATCAGGACATCATTTTCCGCAAAGTAGCAAA
11[96]	13[95]	GATGGCCACCAACCTGAGTCAGTTCGGCCAGT
11[128]	13[127]	GGCCGAATGCAACGGAATTTTTGGTGACGC
11[160]	12[144]	CCAAATGCCCAGGGCCAAGCTGACGGTATTCA
11[192]	13[191]	GACTGGCTCAGGAGAATGTTACCATGTACCAG
11[224]	13[223]	TGTCTCCGAACTCCAGGAGCTCTTGTGGCG
11[256]	13[255]	CTGAACACTCATTATTTCTCCAGGGTGAGCAG
11[288]	13[287]	GGCTCTGGAAGGAGACACTCGAAAGAAGGTCA

11[320]	13[319]	TTTACACTCCACTCCCACTCAATGCAGGTG
12[47]	10[48]	ATTTATTTTGATTTGTCAAACGCCGGGTTTCC
12[79]	10[80]	GCTGGCGGCGTAGCCAGACGGAAGTTCCTG
12[111]	10[112]	TATCCCTGCGGGATCGGCGCACCTGGCACTGT
12[175]	10[176]	GCAATGGGTCAAGGAATGCGCAAATTCTAC
12[207]	10[208]	CGAAAGATGTGTTTGAAGAACTTGAGGCGCAA
12[239]	10[240]	CCTCCCAGTGGCTGCCTGTCTTGGTCTGGCCT
12[271]	10[272]	CATTCTGACTTGCAACCCAAAGAATAACAC
12[303]	10[304]	GGCTGTACTTGACAGCCGAAGTGCATCTTGGA
12[335]	10[336]	GAAGTCTCCTATCCTCCTCCAGTCGACTCAAG
13[160]	14[144]	CTTGAAGAAGAGAGATCCTGGGCTCCAGTACA
14[47]	12[48]	TCTGTTCGCGCGTTAGTCATGGTGATACGGTA
14[79]	12[80]	TCCGGTCACATTCCCGAGCGCTGCAGTCAG
14[111]	12[112]	TTCCCCTGCGTATTCAGTGCGTCATGCCAGAA
14[143]	13[159]	AAAGCGGTACATCTGCAATCGCGGGTGGCAAC
14[175]	12[176]	ATCGATCATCGCTCTGCCTTCCCCCAGGTT
14[207]	12[208]	GCAGAAGATTTCGAGGACGCCGCTGCCAACGC
14[239]	12[240]	TCGGAGACTACAACAACAGCATGTAGCCGATC
14[271]	12[272]	GAACGAATTAGCATCTTGGCTAATCGAGCC
14[303]	12[304]	TCAGAGGTCAGCAGATCCTCGTCGTCTCAAGA
14[335]	12[336]	GCCGACCACAGATAGATTGCAGTCGTTTGAAG
15[32]	17[31]	GCTGTTGCTGCCGCGGCAGAGGGCATTCGT
15[64]	17[63]	CCCTTTCTTGACGCACTCCCCGCCCAGAATTT
15[96]	17[95]	CGACAGCTCCGCAGCCTCTGTCGCTGCACGAC

15[128]	17[127]	GCCGAATCTTTTGCTCTGCGATCTGATGAT
15[160]	16[144]	TCTGCCACTCTCGACCAGGAGCGCCGCCGCTT
15[192]	17[191]	TACTTCATCTCTGAAGGAATCTTAAGGAGTTT
15[224]	17[223]	TAATATGACTAAAGCATCTTCCCCCAACCC
15[256]	17[255]	CTGAGGTTAGAACAAGGACCTGATGCATCGTC
15[288]	17[287]	AGCAGGATACATTGGCTACTTCTCCCGCTCAG
15[320]	17[319]	CAATCCCACTGGCCAAACTCGATCACCATC
16[47]	14[48]	TTTTGCGTGTTGCACTGCTGAGCTTGGGCTTT
16[79]	14[80]	GATGCCGTCGTTGTGTCCGCATCTTGGCGT
16[111]	14[112]	CGCACTTCGAAGCTATATCTTCTGATTGTTTG
16[175]	14[176]	CAAATCTCTCGCTGAGTCTGCTTTTCATGG
16[207]	14[208]	CCAAGAGTTATCTCTGACCTCTCCGACTCCCA
16[239]	14[240]	TGAATCAGTACAGGTTCTCCCTGGCGTGGTGG
16[271]	14[272]	CATCCTGGGTTTCAAAGGGCACTTCCAGGA
16[303]	14[304]	TGCTGGAGTCTGAGAACACAGAACGCATTTCG
16[335]	14[336]	AAGATCCATGATCAGCTCCCTGCATGTTCAAG
17[160]	18[144]	AGATAAGACCCACGCTATGTGGTCCGCATCTC
18[47]	16[48]	TTTCGCCGGTTTTTGCCGCCCTGGGCCGCTGT
18[79]	16[80]	TGTTGTGACGTTGCCGAGGAAGCCTTCGTG
18[111]	16[112]	GGCCGCGGGATGATGCGTTCGTTTGCTGATAC
18[143]	17[159]	GTGCTGAACCTCTTTTGAGGCCACCAGATTGG
18[175]	16[176]	GCCCAAGGTCATCGACTATAACGTAAGCTG
18[207]	16[208]	TCTGTACCCCTGAGCACTAGGAATGCCCTCAT
18[239]	16[240]	TCATCCAGCCACCAGACGCTGCCACGAGTCTC

18[271]	16[272]	GGAACTCGTAATTTCACAACCACAGGGCAA
18[303]	16[304]	ACCAGCAAGGCCAGCTCCTGGCCCAGAATCTC
18[335]	16[336]	CACTTGAAAGAAACCCGATTTGGAAGGATAAG
19[32]	21[31]	TTCCGCTTCAGTTCAAGACGACATGGCACA
19[64]	21[63]	TCTGCGCCCACGGACGCGTTACGCGTTGTGAA
19[96]	21[95]	ATGCAGCCATTTCTTCGCGTCTGCGGCGTGCG
19[160]	20[144]	TCTCAGAACTACTTGTCCGCTGCACACAAGGG
19[192]	21[191]	AAGTCTGCAGAGGGCTCAGTTGGCTTATGCGA
19[224]	21[223]	AAGATTTGAGATTACTGAGGAGTGGATATC
19[256]	21[255]	ATCTGGAAGCGAACACAAGAGATGGGCGTAAA
19[288]	21[287]	ACCTCTCAAGCATCAGGTCTTTCCATATTGAT
19[320]	21[319]	CAATCCCCAGTATCAAGGTATTACACTTGA
20[47]	18[48]	TTCCACCATCAGTGGCCTTTGCTGTCTGACGT
20[79]	18[80]	TGTGCCACGGAGGACGCTTCCTGGGCTGAT
20[111]	18[112]	GCCGGCTGTGTCCGGCGGACGTGCGCTTTCGT
20[175]	18[176]	TTTATCACAGACATGTCTGGAGGTTTGGTG
20[207]	18[208]	CCTCCCAACGGCCTGGAGAAGTTTGAGCCAGT
20[239]	18[240]	AGGTCGTCCTCTGCACCGAGTCAGGACCTTGA
20[271]	18[272]	TCCGGACCGTCTGAAGGTCTGCCCTGCCCT
20[303]	18[304]	GCACGAAAATCGAGAACGCACTGGAGTGATCA
20[335]	18[336]	ATCCGGCAGCGTTTGCCCTTTCCTTGAAGACT
21[160]	22[144]	TTACCGGGGACTTGCAATGGAACTCATGCTGT
22[47]	20[48]	TACCGGTTTCATTCAGCGTCCCCGGCCACCTC
22[79]	20[80]	CGTCTGGCCACGGTGATGGTCCCCGTACTC

22[111]	20[112]	CACCGTGTAAACCGTCAACCTGCACTGGCATC
22[143]	21[159]	AACGCCCGGACCGTACTCCACATCGGCTGTTG
22[175]	20[176]	TTGGCTCACAACGAGTTGGTGATCGCAGCC
22[207]	20[208]	AAATTTATGGTCTGATATCGTAGACTGGCAAC
22[239]	20[240]	AGATCAAGCGAGCAAAGCAAAACAGAATGCTC
22[271]	20[272]	CTGCCACTTCCCAACAGCATTGCTTCGAGA
22[303]	20[304]	GCTGAAGGAATATCAATGAAGGAATTATGCGC
22[335]	20[336]	AACACTGAAAAGAGATTCTTGGCGCCGCTTTC
23[32]	25[31]	AAAAAGCCCTGTGGTTCTTCAACGGCGCTC
23[64]	25[63]	AACACTGTTTAGATCTCTTACCCAGAACTCGT
23[96]	25[95]	TAACCGACCCTGCACTCCACGCCATTAACGAG
23[128]	25[127]	CCAGTCGAACGACGGACGCTGCCTTAACGT
23[160]	24[144]	ATTCCGTTTGACAGCGAGAATCAACATAAGCA
23[192]	25[191]	CTACGAAAGAGTTATTTCAGGACACGCACGC
23[224]	25[223]	AGCAGAACGCAATCACTGAAGCACAGTTCG
23[256]	25[255]	AATTAAAGAAAGAGTTTGGATTAGTATTCCCA
23[288]	25[287]	GCTCCACTATATTACTGCTACAGGGTATCTGT
23[320]	25[319]	TGCGGCATACCCCCTAACCTTTACTTAAAT
24[47]	22[48]	CAGACGGTCCACCGGTTCCGGCGGCAGTTCTG
24[79]	22[80]	ATGCCGCAGTTACCTGGCTAATCGCTGTTA
24[111]	22[112]	TCTGTATACCCAACGATGAATCCGTCTGAGCC
24[175]	22[176]	CAAGCCTCACGTTACATTCATGGCACCATG
24[207]	22[208]	CCAACATTAAATCCTGGCCAGCAGATACCTGA
24[239]	22[240]	AGGCGAATGACACTCATAAATTACAGGTAAGG

24[271]	22[272]	GGACGGGTCAATAAGGAGGGGTCCATGTCG
24[303]	22[304]	AACAATGGTGAAGACATCACCACAAGGCAATA
24[335]	22[336]	CCGGGATTATGCTCAATGGATACAATCAGATC
25[160]	26[144]	GTACTTGAGTGTAATTGCGGAGACACGAGGAC
26[47]	24[48]	GCCTGTGTATTCCTCCAGCCTGTTACAGCTCG
26[79]	24[80]	CGTATGGGCCCCAAGCTCCACACCTGAGGT
26[111]	24[112]	ACGCTCTCAGATTCTTCCCATGGACGGATTAA
26[143]	25[159]	CTGCCTGCTTCATCACAGTTGAACTTTGCGAT
26[175]	24[176]	TTTCTTGCCACTTCAGGAGTGGAACAAACA
26[207]	24[208]	GAAAGAACTCCAAGAAGCCTTGAATATACAGG
26[239]	24[240]	CAGACGTAGTTCGGGAATGCAGGAGCAGAGAC
26[271]	24[272]	CCACTGGATAAGGATGGAAGAGAGACCCAA
26[303]	24[304]	TCGAAGCGTCAGCGGTAAAGGTTGATATAAGA
26[335]	24[336]	GAATGGAATAACAGGTGGCCTTTTCATGAGCG
27[32]	29[31]	CCTTGGTGGAAGATGTGTCGATGTGTCTCA
27[64]	29[63]	TGTCAACCAGAATACCGGACTCTGAATTGCGT
27[96]	29[95]	GGACTTCAGTCGAGAATCAGTGCTATCTGTTC
27[128]	29[127]	AGGCCCAGCTGCTTGACCACGTGAACAGAG
27[160]	28[144]	AAATCCCCAAGGTTTATAAGTCTAACTTGGAT
27[192]	29[191]	CTGATACCAGATATAGAGTCGGCAGCATTCGC
27[256]	29[255]	AACATATAGTGCAGATCCGGTGTCTTTCGCGT
27[288]	29[287]	AACAATCCCTGAGCAAAAAGATTGCTGTAGTG
27[320]	29[319]	TCTGAACAGTTTATAAATGAGTTCGGGTTG
28[47]	26[48]	GGGTCCTCACTCATAGGTGTACTGTCTTCATA

28[79]	26[80]	TCAGCTCCATCTGAACGCAACAGGCTGGAC
28[111]	26[112]	TGGATAATTGTAGAACCTCAACTGGTGCCTTA
28[175]	26[176]	ATTCCATGATTCTGCAATGTGCGGCGGTTT
28[207]	26[208]	GAAGGTATGTGCAAAATTATTATCGAAGAACT
28[239]	26[240]	ATGCAGACACGATACTTGCCCTCTAGCACATG
28[271]	26[272]	TCAATGTTGATGATTAAACCCAAGCTACGA
28[303]	26[304]	GTTAGAGTTCGCACTCGCGGGGATCAGAGTTA
28[335]	26[336]	TTGCAAGGTCGCTACGGCGGGTTTTCACAGGA
29[160]	30[144]	TCTGCATAATGGGTTCAGGATGCACTGCGTCC
30[47]	28[48]	CTTATTCTGCAGGGAGTGATCTTTTGTTACCT
30[79]	28[80]	AGCCAGGTTCTTCGGTAAACTTGTTCAGTT
30[111]	28[112]	GGGTCTTCACAGGATCTGTGATGGCGATGTTA
30[143]	29[159]	GGTTGTCACACAGCTCCCTGAGAGGGTGAGTA
30[175]	28[176]	GGAGGTCATGATGTCTGACGCTGTACAAAT
30[207]	28[208]	TCGACATCAGAGTGAGATCGGTTTATGTAGAT
30[239]	28[240]	GCAGTTAATTGTGAAAATGCTGAATTGTCTCC
30[271]	28[272]	TAATGCGCCAGCGATGCCAGAGTGGCGTTA
30[303]	28[304]	ACGCTGCGTGACCGTACTCAAACAATCAATGA
30[335]	28[336]	CGGGCGCTTCTTACTGTTTCTTTAGACATACA
31[32]	30[48]	TCATCGAAACAGACAGAGTTGATAAAGGGCAC
31[64]	30[80]	CAGCTTCTTGAGGTGAGACTGGTAGATCGTTC
31[96]	30[112]	TCGGCAAAATCCCTTATAAATCAACTGCTCCA
31[160]	30[176]	GAACAAGAGTCCACTATTAAAGAACGGGGTT
31[192]	30[208]	CCAACGTCAAAGGGCGAAAAACCGAGTAACTA

31[224]	30[240]	GGCGATGGCCCACTACGTGAACCAGATCCGCG
31[256]	30[272]	ATCAAGTTTTTTGGGGTCGAGGTGGCCGCGCT
31[288]	30[304]	CACTAAATCGGAACCCTAAAGGGATAGCGGTC
31[320]	30[336]	TTTAGAGCTTGACGGGGAAAGCCGGAAAGGAG
0[359]	1[359]	TTTTGCTGACACGGAAGAATATGCCAACGCCCCGGTTT
1[8]	0[8]	TTTAGACGTTGTGACGTTTTAGTAAATATAATGTGTTT
2[359]	3[359]	TTTTTTCTTCTGCCCAATTTGGCTGCATCGACAGTTTT
3[8]	2[8]	TTTGACAGATGTATGTAAGGGAGGGAAATACAACCTTT
4[359]	5[359]	TTTCTGAAAATGATGCTCTTTAGCTGATGCTAAAGTTT
5[8]	4[8]	TTTGGATTATTCCCTGGTGACGATAATAATATGAATTT
6[359]	7[359]	TTTATTAAGGAGCGTGGCGTGTGGCATTGCAGCAGTTT
7[8]	6[8]	TTTTACCCGCAGCGTTAACACGGTGTTTTCCGCGTTTT
8[359]	9[359]	TTTCCCTACAGCGTGAGCCCAAGAAGTCCATCTTTTTT
9[8]	8[8]	TTTAACCACTTGTGTGGGCGAACTGTTCATCCTTATTT
10[359]	11[359]	TTTTATGAATCACCACCTGAAGCCTCAAGATCACTTTT
11[8]	10[8]	TTTATGGATACGCGACAGCTCAGGAAGCACACCCGTTT
12[359]	13[359]	TTTGCCTGCAAGACCTTGGAGCTATGGCCCTCTTTTTT
13[8]	12[8]	TTTGTGTCGCATTCTTCGGCCTGCCAGCGCCGTCATTT
14[359]	15[359]	TTTTACATCTGCCGCTGCCGCACATGGTGTACGAATTT
15[8]	14[8]	TTTCGTTGCAGCAAGCGTTCCTTAACCGCCTTTGGTTT
16[359]	17[359]	TTTCTGCTCCAACAGGAGGAGAACAGAAGACTAAGTTT
17[8]	16[8]	TTTTGTTTCAGATGACCTGCGCTCCGTTCCGCTGCTTT

18[359]	19[359]	TTTCAGGCTACTGAGTGGCCATCGATCCCAGCTCATTT
19[8]	18[8]	TTTGGACTCTGCGGCTGCGCCGCCGCGTTTTTGATTT
20[359]	21[359]	TTTATCTACTCGTCGCGAAGAGAAACCATAATTGCTTTT
21[8]	20[8]	TTTGGCATCATCCTCCGTCGCAGCACCTCCGGCCGTTT
22[359]	23[359]	TTTGGCGTCAGCCAAGTTATAGACGAGGGCAAGGCTTT
23[8]	22[8]	TTTCTTTACTGCCATATTCTCAGGACTCCTGAAATTTT
24[359]	25[359]	TTTGGGAAGAAGGCGAAGTGAAGAGGATCAGAAATTTT
25[8]	24[8]	TTTTGGATCTGCATAGAGATGCGTCCTGGATGGTCTTT
26[359]	27[359]	TTTTAAATGCACTTCCGAGTGTTTTATGGAGATGATTT
27[8]	26[8]	TTTGTGTAAACGAGCTTGCCTTGTCAGTGAGTGGATTT
28[359]	29[359]	TTTACCGTTTAGCTGAAACCATAAACATTGCTGATTTT
29[8]	28[8]	TTTAATTTGTTCACCAGCTGTTAGTATACTGCTCCTTT
30[359]	31[359]	TTTAGGAAGGGAAGAAAGCGCGAACGTGGCGAGAATTT
31[8]	30[8]	TTTATTGCAGTAATGTGCTTCCCTCCAAGGACTTCTTT
3[160]	4[144]	ATTAATGACAACTGGAAGGAACCCGATTCGTG
4[111]	2[112]	GAGTTACCGACAAAATGAATAAAGCCTTGTGT
19[128]	21[127]	CGTGCTGTGAGCAGCTGATGCAGGATGACA
27[224]	29[223]	GGTGGAAATCACGAAGGTGTTTTGTAAAAG
1[32]	3[31]	AATAAAATCCTGCTGGCATTCTGCCAACGT
1[64]	3[63]	CACTTGATCTTAATAGTATTGGTTTTTGAAGT
1[96]	3[95]	ACCTGAGCTGATTTATGTCAAATAAGCAAGCG

1[128]	3[127]	ACGAGGGAATCTGGTCTGACCTAACAATCT
1[192]	3[191]	ATATTTATTTCCCGATGCTTTTTGTTGCCCAT
1[224]	3[223]	CGTCATTAGTAGCGAGTAGCATTTGTGAGC
1[256]	3[255]	CAGCCCTGTGAGCACATCCTGTAAGTATAAAT
1[288]	3[287]	GTCCGGGATGCCGGTTTGTTTTGAATCCATTT
1[320]	3[319]	TTTAGCGGATGGGTGATGGTTTTTAACAAC
5[32]	7[31]	ATCACCAAATGCAATGTTTTTGGGTGATGG
5[64]	7[63]	TAGTCTGTAATGCCATTATGCAAGAAAGAATG
5[96]	7[95]	CACTGTCATTCATTCTTCATCACTGATGCTGG
5[128]	7[127]	ATTACTGTCGTCCTGTTCGGAGGCATGTGC
5[192]	7[191]	CTGTTAATGAACGGTCAGAGAGATTATTTACT
5[224]	7[223]	TACAAAGAGCCTGATTGCAAAACTTGTCAC
5[256]	7[255]	AACTGAAGCTGGTTATGGTCAGTTCAGGCGCC
5[288]	7[287]	CATGCAGATGCAGCAATATCTGGGTTGGTGGG
5[320]	7[319]	TCGATGCTCGTGGTGATATCCGTGAAAGAC
8[143]	7[159]	TGACTGTGCTCACGGCTGTACCGGCATGGTTA
9[32]	11[31]	CCGCGGCGTCCAGCCTCGCATAAAGTTTTG
9[64]	11[63]	AGCATGAGCAGCACGACCGCTGGCTGCCCCTG
9[96]	11[95]	GTGCCGTAGACTTAATTCCATCCTCGATATCT
9[128]	11[127]	CCCGTAACGTACCGGATGCACTGCCGGAAA
9[192]	11[191]	AAGGACGTCTCAAGGTTGGACCAAGGAATGGT
9[224]	11[223]	GCTGGAGGCCGCATGCCAAGAGCACCTGGT
9[256]	11[255]	GATAACTACACATGCTCTGTTCTCAGAAGGAA
9[288]	11[287]	GCTATTTGCAAATCTCTAAGAAAGCACTTTCA

9[320]	11[319]	ATTGACGCATGCGCACATATGAACCCAAAC
12[143]	11[159]	AGAACATCTTCTCACCGGCCCCAATTGGCTAT
13[32]	15[31]	CCGCAAGTTTCATCCATTACCATCAGACGT
13[64]	15[63]	ATCTGGATGTGCCGGACTGTCCAGGCACTATC
13[96]	15[95]	TCATTCAGAGCGCGGTTGGTGCTGCTCAAGCG
13[128]	15[127]	GTCGATAGTTCGCAATCTGGGTCACGTTTT
13[192]	15[191]	ACTCTGTGTCAGGGACCAGTTGCCCAAAGATG
13[224]	15[223]	CACCTACATGTTGCGCAAAGCCAAGATCAA
13[256]	15[255]	GAATTTCCGGGATACCTTCACTGGCCGCCTTT
13[288]	15[287]	GCTTGTTCATGCATCCAGAATTGTCCAGAATA
13[320]	15[319]	CTCAGGCGCTGATGTTCGCCCTAGAGGATT
16[143]	15[159]	CTGCCGCATTTTTGCACGTATTGCGACGCATA
17[32]	19[31]	CTCGGACCACCGGCACTGGTGGGCACTTTT
17[64]	19[63]	TCTGCCGCCGCTGCAGCATTCGTTATGCCGCT
17[96]	19[95]	CGGCACTTTGGAGGCAGACGTGGCAGCTGACG
17[128]	19[127]	TTTGCTGGTGGCGGCCTCTGACTGGCGGCG
17[192]	19[191]	CGCCTGTTCTCTGTTGCGCAGAGAAGTGGACC
17[224]	19[223]	ATTCATTGAGATGGATGGCGTGGACCTGAG
17[256]	19[255]	ACAGAGGTCCGTCCGCTTCGGCAAAGCAGCTG
17[288]	19[287]	GACTGAGAGACTCCAACTTCATCACAGCGATG
17[320]	19[319]	CAACATGGGATTCCAGGCTGGATTATCTTG
20[143]	19[159]	CCGCGACCAGTCAGTTGCATCAGTCCTGAGTC
21[32]	23[31]	GAGAAAATTCCTGTGCCGTCTTACCCCACA
21[64]	23[63]	TCTTCATATTTCAGCTGAATGGTGGCTGTCAT

21[96]	23[95]	ATGGTGGATCACCACCACCGTGGTAGAATTTA
21[128]	23[127]	CTGTACTGCTTCATCCGGATTCTCAGTACG
21[192]	23[191]	AATGAAGTCAAGCAGCAGAATCATCTTGATCG
21[224]	23[223]	TTTATGTACTTACGGCTGACGAAGGAATCA
21[256]	23[255]	AATTGCCACGCGCAGCAAAATCAGATGAGCAA
21[288]	23[287]	GCTCTCGGCCATATAACAACAAACTGCCTGAT
21[320]	23[319]	TAAATGCGCGATGCATTCCGAGAAAGAAAT
24[143]	23[159]	ATATCAACCCACTGCCGGAGCCTTCAGTGATA
25[32]	27[31]	AGCACAGCAAGGTGAGGTAACATGGCATTT
25[64]	27[63]	CGAAGCAATTGCCTCCCAGGCCCAGAACTCGC
25[96]	27[95]	GCCAACAATGTCTTACCAGTGCCAGTGTGGTC
25[128]	27[127]	CGATTCCCCAGCAGACCTCCGACTTCTTCC
25[192]	27[191]	CAGCGACGACCAAGCCGCCTGATGAGAAATGA
25[224]	27[223]	TCGATGGACCAATATTCGAATTACGAGTAC
25[256]	27[255]	CCTCCGGTTGATATGGGCGCAGATCTGTACAA
25[288]	27[287]	TCCACGAAGAGAACATCAACGACTTATTACAT
25[320]	27[319]	CGACCAGTGGAGAGCCATTCAATTATTTTA
28[143]	27[159]	TTGGTTCAATCTTCGGTGGTGTGCATGAAGAC
29[32]	31[31]	ATCTGGTTGAATGGCACAACTTCTAGACTTC
29[64]	31[63]	TCTCCACATTCGACGTTATTGGACCCAGCAAA
29[96]	31[95]	GGCCAAGCTTCATCTCCAGGGCCACCCCGAAA
29[128]	31[127]	AAGGGTCCGCACTCCTTCAGCAGAAGAATAG
29[192]	31[191]	ATCAAAGGATTACGCATCGCTATTCGTGGACT
29[224]	31[223]	ATAACGCTCGAACAAGACCCGTTTCTATCAG

29[256]	31[255]	CGTCTTCACGCTACAGGGCGCGTGTCACCCAA
29[288]	31[287]	TCAGATGACGTAACCACCACACCCCCGTAAAG
29[320]	31[319]	AGTATTAAGGGCGCTGGCAAGTGGCCCCCGA
31[128]	31[159]	CCCGAGATAGGGTTGAGTGTTGTTCCAGTTTG
4[143]	3[159]	TAAAAAATATCCCTCCGTGGATCTAGAAGTAT

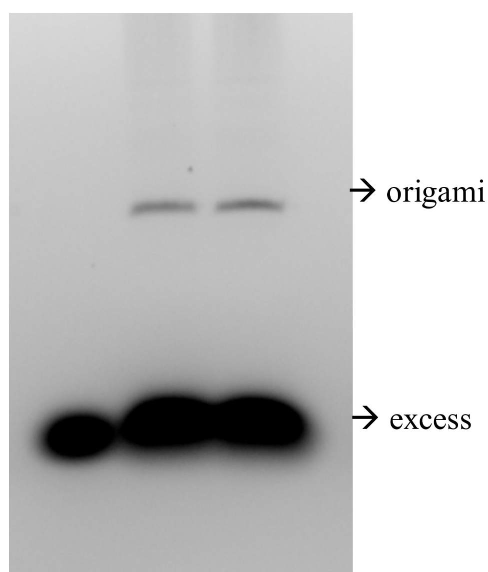


Figure 2S.1. Gel electrophoresis of DNA origami synthesized with the pScaf-11054.1. This origami was extracted from the gel and and analyzed by AFM.

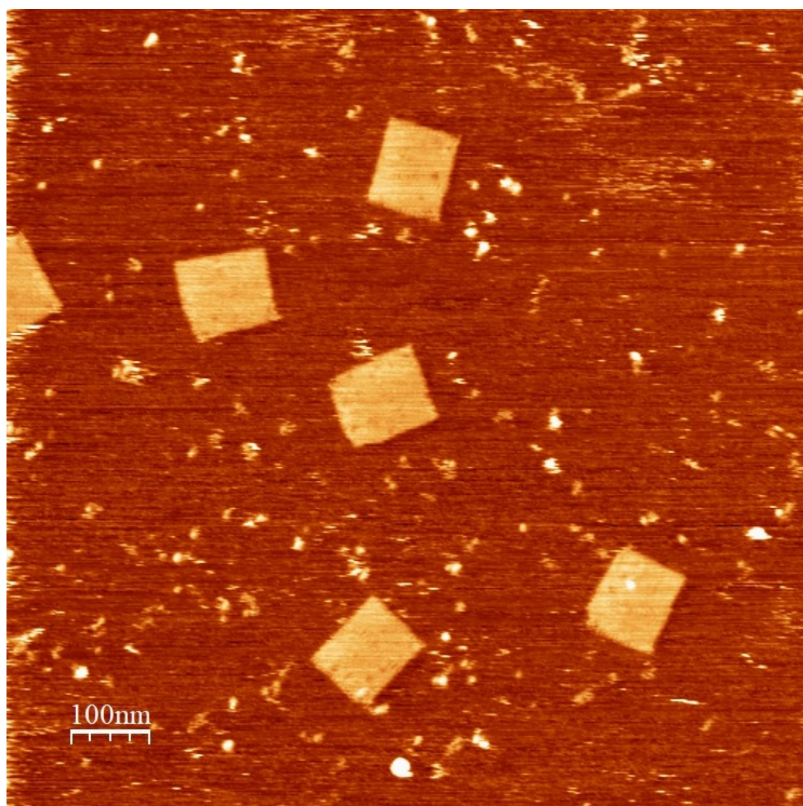


Figure 2S.2. Larger field of view image of several large origami rectangles synthesized using pScaf-11054.1.

Figure 2S.3. Larger field of view image of several large origami rectangles synthesized using pScaf-11054.1.

Appendix

<u>Ongoing Work- dNAM 2.0: Utilizing the New Node</u>

As previously described, we established a larger dNAM node with 67% more data points. To this end, first a larger origami node was designed in caDNAno to accommodate an 8x10 matrix with 10nm spacing between data points. However, the new origami design requires a larger scaffold of 11,054 nts. We produced this scaffold by cloning a phagemid to make ssDNA of that exact length and producing it in E. coli co-infected with helper phage. Next, we confirmed that this scaffold formed the 2D rectangular origami that was ~50% larger than the original node via AFM. Finally, the 8x10 matrix of data points was validated by DNA-PAINT. This establishes a new origami that is now available to be used as a dNAM node for data storage. The following sections describe the ongoing work and progress that has been made towards using the new node to encode a message.

<u>Encoding “Data is in our DNA!\n”</u>

The new PSCAF-11054.1 node was then used to encode the same message from the dNAM prototype: “Data is in our DNA!\”. To this end, we first generated a random mapping scheme ensuring all the mapping follows some core requirements: 1. The mapping relation between each parity and data bit must be rotation invariant, which is achieved by mirroring over the center axis. 2. The mapping must be evening distributed, so each parity contains a certain predefined number of data bits. Next, we ranked the mapping based on the distance of parity and data relationship, with the higher the distance being considered better. First, the Euclidean distance between a parity bit and their associated data bits are calculated. This is later normalized by total number of

associated data bits. Then, all these distances are summed together and normalized by total number of parity bits. With this encoding scheme, the PSCAF-11054.1 node contains 29 bits/node, compared to the original M13 node, which only included 16 bits/node (Figure 2A.1). Using the new node and encoding scheme, we will be able to encode this message in 6 PSCAF-11054.1 origami nodes instead of 15 (Figure 2A.2).

0	1	0	0	0	1	0	0	0	1
1	0	1	0	1	1	0	1	1	1
1	0	1	0	1	0	1	1	1	0
0	1	1	0	1	0	0	1	0	1
0	1	0	1	0	1	1	0	1	1
0	1	1	0	1	0	0	1	0	1
1	0	1	1	0	1	0	1	1	0
0	0	0	1	1	0	0	0	0	0

data parity checksum orientation index

Figure 2A.1 PSCAF-11054.1 8x10 matrix encoding scheme.

Making and Reading Node 0

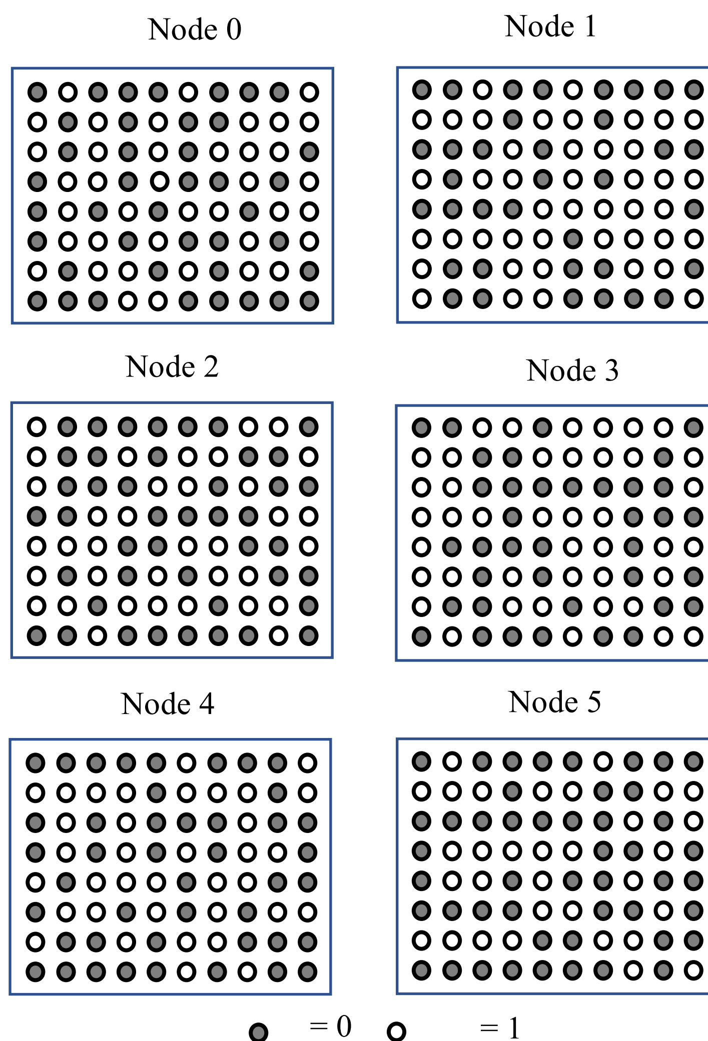


Figure 2A.2 The nodes to encode "Data is in our DNA!\n" using the PSCAF-11054.1 node with an 8x10 matrix.

To encode the message using PSCAF-11054.1 we started by assembling node 0. In a 50uL reaction, we combined ~10nM PSCAF-11054.1 scaffold with 30x staple strands with extended staple strands at the designated locations (Figure 2A.3), annealed them in the thermocycler overnight, and then gel purified. Samples were then analyzed by DNA-PAINT TIRF SRM, and images averaged to produce the most complete matrix

possible. From the first trial on the first node, we were able to recover ~70% of data points (27/39 by eye, this may improve or worsen when using an algorithm to decode it). While this is good for our first attempt, this rate of recovery will need to be improved in order to move forward with the other nodes and successfully decode the message. However, there is a chance that other nodes may perform better, and this nodes performance was due to inherent issues with this pattern. In addition, moving forward we will optimize the origami synthesis and DNA-PAINT parameters in order to achieve better data recovery.

Attempting to Synthesize PSCAF-11054.1 Node 0 with Individual Staple Strands

Our first attempt to synthesize PSCAF-11054.1 node 0 used oPool staples (discussed further in chapter 3). In trying to improve performance we attempted to fold PSCAF-11054.1 node 0 using individually synthesized staples. These staples were purchased in pre-diluted plates and combined by hand to make one mixture that contained all staples not at the 80 potential docking sites, and then another mixture that contained the 80 points with or without the extended strands for node 0. For these syntheses, we used ~10nM PSCAF-11054.1 ssDNA scaffold with 30x staple strands, and 50x docking staple strands. Then annealed overnight in the thermocycler and gel purified. However, our first attempt did not show anything on the gel with our normal method using SybrGold dye. We saw the same thing in our second attempt and therefore post-stained with gelred to see if the dye was the problem. In fact, it was. After post-staining we could see clear origami bands. However, we could not successfully image these samples with DNA-PAINT (post-staining with GelRed appears to interfere with DNA-PAINT). We then prepared a very large sample, approximately 30nM PSCAF-11054.1

scaffold, ran the gel, but only used a small aliquot to stain with gelred and loaded it in the next lane. This way we could collect an unstained sample in the adjacent lane at the location of the stained band. Interestingly, this gel appeared to have excess scaffold instead of staples, even though the staples should still have been in huge excess. Our first attempt to do DNA-PAINT on these samples did not look good. However, we still have more samples to look at, and would like to do AFM to see how the structures look also. We also did another experiment with M13 and used the same SybrGold to stain it and it did appear to work fine on that gel. We have previously used SybrGold with PSCAF-11054.1 and pooled staples, so we do believe that there is some problem with the individual staples and SybrGold. We are also somewhat concerned now that there is a bigger problem with the plates of staples. However, there is further work to be done here.

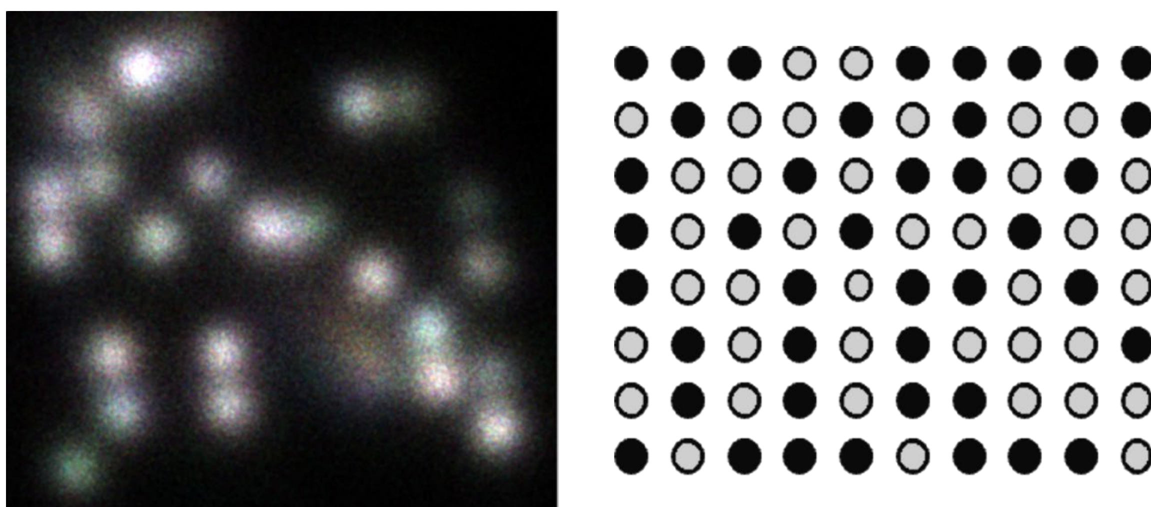


Figure 2A.3 PSCAF-11054.1 node 0- DNA-PAINT on the left shows an averaged image of PSCAF-11054.1 node 0. Right- is the pattern that should be encoded in node 0.

Discussion

One of the limiting factors to optimizing the PSCAF-11054.1 node is the rate of ssDNA production. We are currently producing the scaffold in 20mL cultures making ~25ng/mL. We generally make 8-12 20mL cultures at once, but this is generally only enough to do one origami synthesis that still results in a very low concentration sample. We have attempted to increase our production in many ways including increasing the culture size and growth conditions, changing at what OD we infect with helper phage, using a helper plasmid instead of helper phage, to name a few. Improving the ssDNA production would be a huge benefit to optimizing the system downstream.

Another area that merits further consideration is the staple strands used for origami synthesis. We switched to using oPools from IDT for which the oligos are all synthesized together, instead of getting plates of individually synthesized oligos and combining them ourselves. We saw this as a way to make the dNAM system more efficient and user friendly as it would not require the same amount time and number of liquid handling steps. However, we have only started to evaluate their performance, as will be described in chapter 4.

A big factor that may be contributing to the performance of the PSCAF-11054.1 node are DNA-PAINT parameters. We have not been able to spend a lot of time optimizing DNA-PAINT because of the lack of ssDNA available. However, there is the potential to improve resolution by changing the length of the imager probe which determines the on/off rate of the blinking. In addition, we can change the image acquisition parameters like duration of acquisition, frame speed, and focus controls. Finally, we could also potentially improve image post-processing, this would probably be

better with higher concentration samples also. If we can capture more origamis, we can average them to create a better final image.

Finally, if we can cannot optimize the system well enough by evaluating the staples or changing DNA-PAINT parameters, we may have to consider that there are inherent characteristics of the scaffold that are affecting performance. The dNAM name group has started making a program to evaluate and optimize hybridizing sequences, however it was not designed to handle sequences that are 11,054 nts long. We have also started collaborating with a colleague at New Castle University who can evaluate inter/intra sequence interactions for the scaffold and staples.

CHAPTER 3: TAKING ADVANTAGE OF ARRAY-BASED OLIGO SYNTHESIS TO ADVANCE DNAM

Abstract

The demand for data storage is beginning to outpace silicon supplies. Advances in nucleic acid synthesis technologies increase the potential of DNA as an alternative material for data storage. Digital Nucleic Acid Memory (dNAM) has been developed as a DNA-based data storage approach which uses DNA origami to spatially position DNA probe sequences that are then read by super-resolution microscopy. In our previous dNAM prototype, different data patterns were created when DNA origami were self-assembled by mixing a precise set of hundreds of DNA oligos (staples) with one large single-stranded DNA scaffold. The data set for the dNAM prototype required multiple different origami structures that were synthesized separately because they all use the same DNA scaffold, and therefore staples would overlap and not form the exact patterns necessary if mixed together. Each synthesis also required several hundreds of liquid handling steps that are inefficient and error-prone, even with automation. To take advantage of recent advances in pooled oligonucleotide synthesis methods and improve the efficiency of origami synthesis for dNAM, we set out to show a proof-of-concept multi-origami synthesis with two different dNAM nodes. We demonstrated that sequence orthogonality enables self-assembly of two different origami structures that use two different scaffold sequences. We demonstrated that custom scaffolds can be designed that use orthogonal sets of staple strands that do not interfere with the formation of unrelated

origami when mixed together. We also showed that both origami structures can form simultaneously when both scaffolds and all staples are mixed. These results demonstrate a pathway toward "random access" of data from large premixed staple pools, and "one-pot synthesis" of multiple dNAM origami structures. Further, they suggest that designing custom, orthogonal scaffold sequences could allow dNAM to take advantage of cutting-edge array-based oligo synthesis approaches to improve the efficiency of origami synthesis by assembling multiple origamis in a one-pot assembly. We further designed several ssDNA scaffolds that would use non-overlapping (orthogonal) DNA oligonucleotide staple strands. We used a stochastic Monte Carlo approach to design 6 orthogonal scaffolds for future production. In the future, these 6 orthogonal scaffolds will enable the encoding of our previous prototype message "Data is in our DNA!\n" in a one-pot synthesis, which dramatically increase the efficiency of dNAM synthesis by eliminating thousands of liquid handling steps.

Introduction

As global data production continues to increase, current technologies and materials will not be able to keep up with demand for storage[42]. DNA is currently being explored as a promising alternative data storage material because of its inherent data density, stability, and low energy requirements[43]. Digital Nucleic Acid Memory (dNAM) is one system for encoding data in DNA that is read via super-resolution microscopy instead of DNA sequencing . To accomplish this, dNAM uses a 2D rectangular DNA origami structure to spatially orient protruding strands of DNA that act as addressable sites at nanometer precision. These protruding "data strands" transiently bind to fluorophore-labeled "imager strands" to create a blinking effect that can be used to

achieve super resolution imaging (DNA-PAINT)[53]. Custom algorithms have been developed to encode data within the fluorescent patterns, and decode the data from the super-resolution microscopy data. “Data is in our DNA!\n” was successfully encode into 15 origami structures, read by DNA-PAINT, and then decoded.

The foundation of the dNAM system is DNA origami nanostructures that require hundreds of synthetic DNA oligonucleotides. Recent advances in oligonucleotide synthesis offer opportunities to employ vast pools of oligos[65]. Array-based synthesis has made it possible to simultaneously synthesize up to millions of different, specific oligonucleotides that come in pre-mixed pools. These could potentially be used as staples for numerous origami. However, oligo pools synthesized with this technology have not been fully evaluated for their efficiency in origami synthesis or in the dNAM system. In addition, the ability to make multiple non-overlapping origami structures from vast pools of oligos also requires the design of numerous different single stranded DNA scaffolds that do not compete for the same staple strands, but instead use non-interfering “orthogonal” sets of oligonucleotide staple strands.

The previously reported dNAM prototype used 15 different M13mp18-based origami. Each origami required ~150 staples per origami, and over 2,000 oligos in total. The synthesis of the required origami structures required the mixing of the different sets of oligos which required >2,000 error-prone and time consuming liquid handling steps. This number of DNA oligonucleotides can be easily obtained in a single pool-based synthesis, with a sufficient yield for origami production. However, it is not possible to possible to fold the 15 different origami structures simultaneously using a single M13-based scaffold because the staple strands will compete for binding sites, and the message

will be completely scrambled and information lost. In addition, the concentration of each individual DNA oligo in a pool synthesis is difficult to measure, and it remains uncertain how this might effect DNA origami synthesis from mixed pools of staple strands.

To take advantage of pooled oligos for staple strands, we set out to design new scaffolds not based on the M13mp18 sequence[20]. Recently, several groups have demonstrated approaches to produce custom scaffolds[15,19,22]. We decided to utilize a recently reported phagemid, termed “pScaf”, that includes an f’ origin for ssDNA production, an optimized termination sequence, and the sequences necessary for packaging and export of ssDNA for easy production and isolation[63]. We used this system to create a new custom scaffold that is ~11KB and that can make a large rectangular origami (see Ch. 2). Importantly, this large origami shares no staple strands in common with the origami in our prototype dNAM structures that used the M13mp18 ssDNA scaffold.

Here we set out to use these two “orthogonal” scaffolds to test or ability to use large pre-mixed pools of oligonucleotides in origami synthesis for dNAM. In order to take advantage of array-based oligo synthesis technology to advance the dNAM system, we evaluated several preliminary factors. We checked the functionality of pooled staples by doing DNA-PAINT on M13mp18-based origami with all docking sites for the 6x8 matrix used in the original dNAM prototype. In addition, we showed that using a specific scaffold could achieve random access of data from a large mixed pool of staples, and that multiple origami structures could be produced in a single synthesis reaction. Finally, we designed multiple orthogonal scaffolds using a de novo, Monte-Carlo-based program and started a library of scaffold sequences for future use with our new, larger dNAM node.

These advances would push dNAM towards more user-friendly, high-throughput applications.

Materials and Methods

Origami synthesis

Origami was synthesized using ~10nM scaffold with 30x staples (IDT oPools or plates) in 0.5xTAE with 18mM $MgCl_2$. The origami mixture was folded in the thermocycler [1 min 90 °C, 2 min 80 °C, then from 80 °C to 20 °C over 12 h]. Synthesis mixture was purified on 0.5xTAE 8mM $MgCl_2$ 1% agarose gel at 70V for ~2hrs. Bands were excised, gel crushed, and origami solution collected. Concentration was determined by nanodrop.

Atomic Force Microscopy

AFM analysis was conducted on freshly cut mica. 4 µL of a dNAM origami sample was deposited onto the substrate for 1 min and then 100 µL of AFM buffer added to form a droplet on top of the sample. AFM imaging was performed with a Dimension-FastScan system from Bruker (Billerica MA, USA) set to amplitude modulation mode. Imaging was carried out in liquid with a set-point ratio between the free amplitude and imaging amplitude of ~0.7. The FastScan D cantilever was supplied by Bruker, with a nominal spring constant of 0.25 N/m. Sub-nanometer amplitude was used to image DNA docking strand positions on every origami structure following the method of (10.3791/54924). Tilt correction (line or plane flattening) was performed using WSxM (10.1063/1.2432410) software package version 5.0 Develop 9.22 (Nanotec Electronica, Madrid, Spain) and a low-pass filter applied to remove noise.

Super Resolution Microscopy sample preparation

The formed DNA origami structures were deposited on glass substrates using a microfluidic cell (sticky-Slide, Ibidi GmbH) equipped with inlet and outlet tubes to allow buffer exchange and washing steps between different imager probes solutions. Borosilicate glass slides (25 x 75 and 22 x 22 mm, #1 Gold Seal Coverglass) were sonicated in 0.1% (v/v) Liquinox and nano-pure water (1 min in each) to remove contaminants and dried at 40 ºC for at least 30 min. The coverslips were then, rinsed with methanol and nano-pure water and stored at 40 ºC prior to use. The glow discharge technique previously described by Green et al. (Green et al) was used to deposit DNA origami onto glass coverslips using a PELCO easiGlow™ Glow Discharge Cleaning System (Ted Pella Inc.). Briefly, coverslips that had been cleaned were exposed to glow discharge generated using 20 mAmp at 0.5 mbar for 75 s. For DNA-PAINT analysis, the sticky-Slide flow cell (~50 µL channel volume) was glued to the coverslip and the DNA origami solution deposited by introducing 600 µL of 0.02 nM DNA origami (a mixture of dNAM origami, and sharp triangle origami added as additional fiducial markers, in deposition buffer) into the channel and incubated for 30 min at room temperature. After deposition, the flow chamber was rinsed with 3 mL of deposition buffer (no DNA origami) and mounted on the Fluorescence Microscope.

DNA PAINT Imaging

Immediately before imaging the imager probe solution (imager probe strand in imaging buffer) was supplemented of 5 mM PCA to initiate the oxygen scavenger reaction. DNA origami were imaged below the diffraction-limit of light via DNA-PAINT using an inverted Nikon Eclipse Ti2 microscope from Nikon Instruments in total internal reflectance

fluorescence (TIRF) mode. The images were acquired using an: integrated Perfect Focus System from Nikon Instruments; an oil-immersion CFI Apochromat 100x TIRF objective, with a 1.49 numerical aperture, plus an extra 1.5x magnification from Nikon Instruments; and a 405/488/561/647 nm Laser Quad Band Set TIRF filter cube from Chroma. A 561 nm laser source excited fluorescence from the DNA-PAINT imager strands within an evanescent field extending a few hundred nanometers above the surface of the glass coverslip. The emitted fluorescence was imaged onto the full chip with 512 x 512 pixels (1 pixel = 16µm) using a ProEM EMCCD camera from Princeton Instruments at selected exposure time (50-150-300ms). Images with blinking events were recorded into a stack (typically 40,000 frames per recording) using Nikon NIS-Elements version 5.20.00 from Nikon Instruments prior to processing and analysis.

Data Analysis

After recording a DNA-PAINT stack, the center position of signals (localizations) emitted by imager probes, transiently binding to DNA origami docking strands, were identified using the ImageJ ThunderSTORM plugin. The localizations were rendered and then drift corrected using the Picasso-Render software package, as described by Schnitzbauer et al. When needed, the rendered origami patterns were picked and averaged together using Picasso-Average software package. Data visualization and peak fitting of image data for PSF analysis were performed using OriginPro Version 2019b (OriginLab).

Results

M13 Comparison of Staple Performance

To test the performance of staples synthesized as oligo pools, we synthesized the same rectangular origami from the dNAM prototype using the M13mp18 scaffold and either staples synthesized as pools or individually synthesized staples (plates) that were mixed by liquid handling. Folding reactions were set up at the same concentrations of scaffold and staple stands and run on the thermocycler at the same time (see Methods). Samples were compared by agarose gel electrophoresis. Origami products were indistinguishable by gel electrophoresis in terms of size of origami and yield (Fig. 3.1). Each origami sample was further analyzed by AFM and showed well-formed structures of the same size (Fig. 3.2). We also looked at the performance of structures made with both types of staples using DNA-PAINT (Fig. 3.3). Both samples performed well, and showed most of the 6x8 matrix points. While there is a slight difference in resolution, we believe this could simply be due to microscope issues as samples were imaged on different days. Further analysis should be done to investigate batch-to-batch differences for the pooled staples. We conclude that the staple strands synthesized as pools perform as well as individually synthesized staple strands. This opens the door to harness the potential of

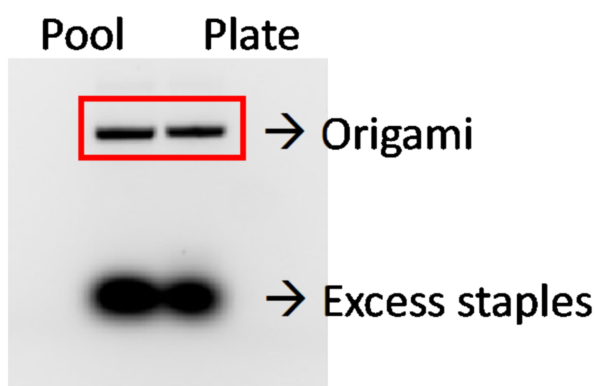


Figure 3.1 Gel comparing M13 origami synthesized with pooled staples (left) compared to individually synthesized staples (right).

huge staple pools for dNAM to enable “random access” retrieval of data and/or one-pot synthesis of multiple origami.

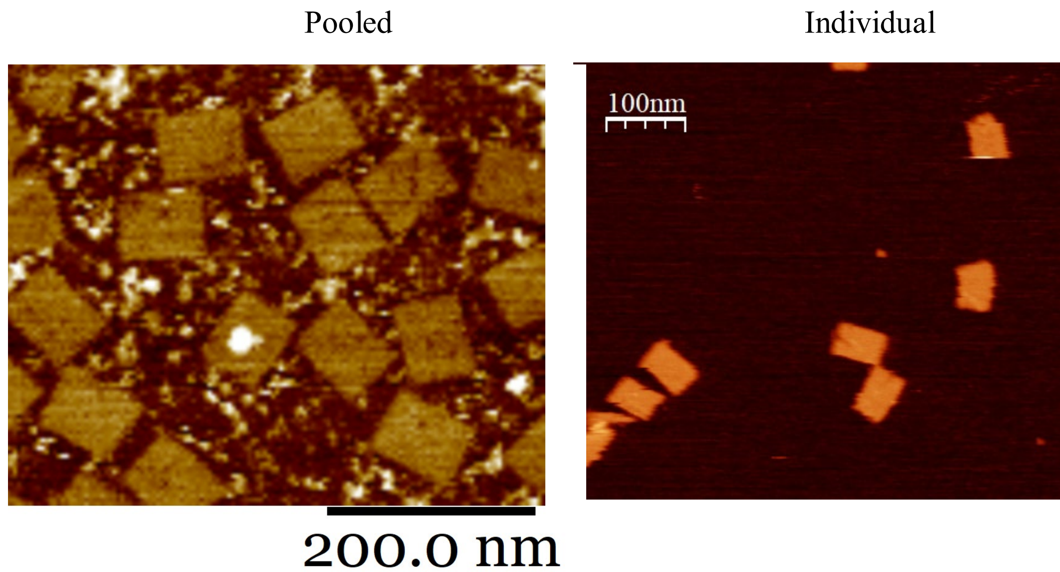


Figure 3.2 AFM comparing M13mp18 origami made with pooled staples (left) or individually synthesized staples (right).

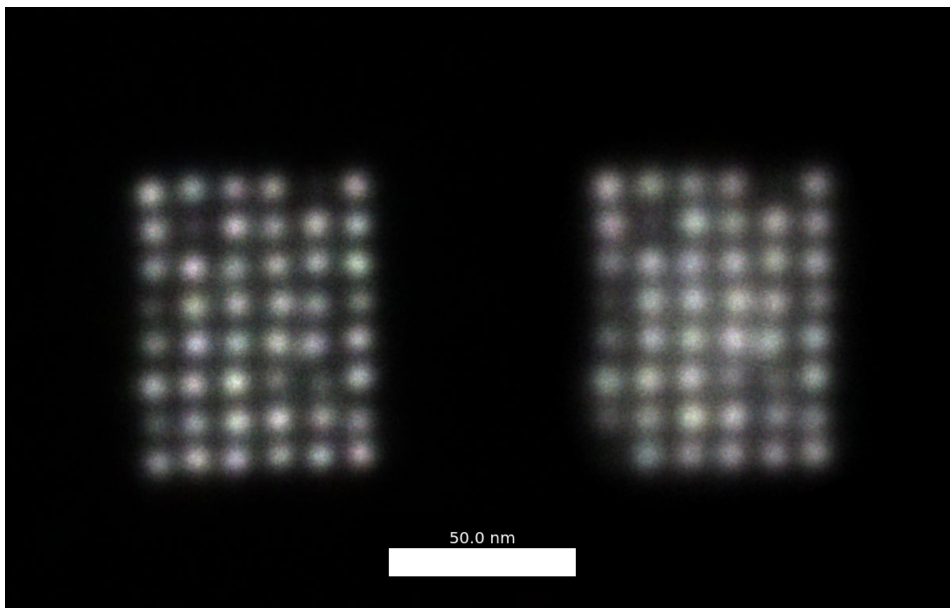
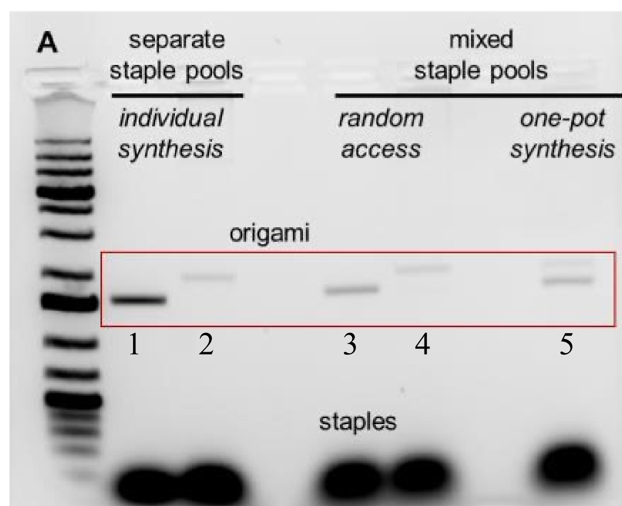


Figure 3.3 DNA-PAINT comparing M13mp18 origami made with pooled staples (right) or individually synthesized staples (left).

<u>One-Pot Multi-Origami Synthesis</u>

We next set out to evaluate the potential of orthogonal scaffolds in the context of mixed pools of oligos. We compared the ability of each scaffold to fold into the appropriate origami with individual and mixed sets of staple strands. We used pScaf-11054.1 scaffold (as described in chapter 2) and its staple strands along with the previously used staple strands and the M13mp18 scaffold of the dNAM prototype (both sets of staples were pools). We folded origami with either one scaffold and both sets of staples ("random access") or both scaffolds and both sets of staples ("one-pot synthesis"). We compared the resulting origami to the same structures formed by folding each scaffold with only the individual set of staple strands using gel electrophoresis (Fig. 3.4-A lanes 1 and 2). The origami are easily distinguishable because their different sizes are observable during gel electrophoresis and AFM. We found that both scaffolds folded correctly even in the presence of the unrelated staple strands, validating the ability of scaffolds to randomly access and present appropriate data from mixtures of DNA staples (Fig. 3.4-A lanes 3 and 4). We also found that both origami structures folded correctly when mixed together with both sets of staple strands (Fig. 3.4-A lane 5- two bands). We further analyzed this one pot synthesis product by AFM and found that both origami in the same mixture were well formed (Fig. 3.4-B). This suggests that orthogonal scaffolds can be used for both random access and one-pot synthesis, which is promising forusing large oligo pools for dNAM data storage. In order to further test this approach, we will need additional orthogonal scaffolds.



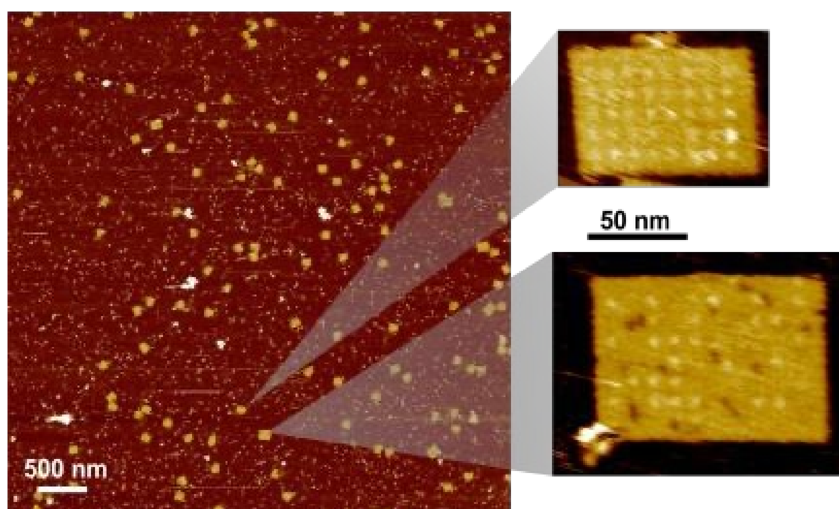


Figure 3.4 Random access and one-pot synthesis (A) Gel electrophoresis of origami from m13mp18 and a custom 11,054 nt scaffold. Origami bands were observed with one scaffold and one pool of staples (individual synthesis), two mixed pools of staples and a single scaffold (random access) and both staple pools and both scaffolds (one-pot synthesis). (B) AFM characterization confirmed two different sized DNA origami in the same "one-pot" synthesis.

Orthogonal Scaffolds Designs

Next, we set out to design multiple scaffolds that use orthogonal sets of staple strands. For this design challenge, we used a program called Scaffold Smith[19]. This program creates a scaffold sequence based on a caDNAno origami design input. This sequence can then be put back into caDNAno to design the appropriate set of staple sequences. Scaffold Smith begins at the user defined start site, then builds a sequence de

novo one base at a time using a stochastic Monte Carlo process. The user also defines the statistical weights of base pair steps, allowing for some control of thermodynamics properties of the scaffold. In addition, there are controls to exclude specific sequences or include specific sequences at a defined location. Scaffold Smith then analyzes the complete scaffold sequence to determine the degree of sequence redundancy, multiplicity, and actual base pair step weights.

Using Scaffold Smith, we created a set of six scaffold sequences, each 11,054 nts in length. This size was chosen because it can fold into a large 8x10 dNAM origami structure as demonstrated in chapter two of this dissertation. These sequences exclude the imager strand docking sequence, and include the f' origin sequence, terminator sequence, and packaging sequence necessary to produce the ssDNA in E. coli with M13KO7 helper phage. We created a database of these sequences stored on Benchling, which enables the design of the cloning experiments needed to produce the required phagemids. For the purpose of this dissertation, we designed and analyzed 6 scaffolds that are designed to encode the prototype message "Data is in our DNA!\n" using a single origami synthesis from a single oligo pool synthesis. In addition, the database can be expanded using Scaffold [69]Smith. Additional computational software advancements for design automation and sequence validation steps will be required to fully harness this approach.

There are not many guiding principles for designing an "optimal" scaffold sequence for DNA origami, however, we know that it should be low redundancy and, to use them together, orthogonal from the other scaffolds to be used. Scaffold Smith allows the user to set a "pseudo de Bruijn order" to limit the redundancy of the sequence it generates to a certain number of repeats of any sequence of a given fragment length. Five

of the six sequences were designed using a pseudo de Bruijn (DB) sequence order of 8 with a max redundancy of 5 (Scaffold Smith defaults). To test these settings, we created one sequence with a DB sequence order of 6 with a max of 3 duplicates (seq4) and compared the number of duplicates per fragment length within each sequence to M13mp18 (Fig. 3.5). All of the novel sequences were significantly less redundant than the standard M13mp18 scaffold, suggesting they would be suitable for folding origami. In addition, we looked at inter-sequence overlap using local and global pairwise alignments and a clustal omega alignment on Benchling. This showed very minimal sequence alignment, other than both ends which include the sequences necessary for single stranded DNA production. Not aligned regions can be seen in figure 3.6 in red, with overlaps in gray, with the maximum overlap of 9 nucleotides highlighted in the upper portion of the figure (Fig. 3.6). These sequences should be adequately orthogonal that they will not have common staples other than the necessary conserved portions. Together, this analysis suggests that these sequences could be used together in a one-pot synthesis to produce 6 different origami structures of the same size and shape.

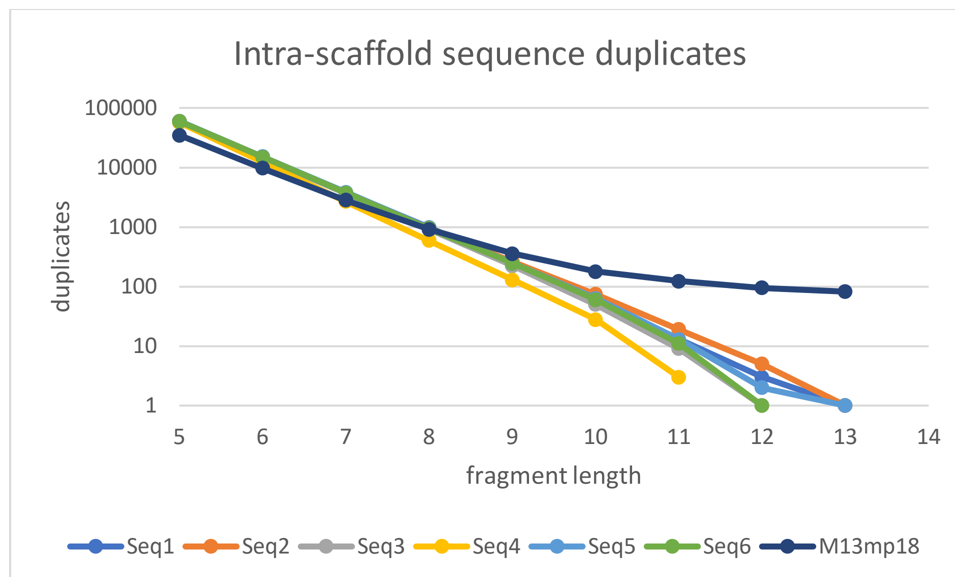


Figure 3.5 Intra-scaffold sequences duplicates based on fragment length for 6 scaffolds generated by Scaffold Smith (Seq1-Seq6) compared to M13mp18.

Figure 3.6 **Sequence alignments to compare inter-sequence overlap using Benchling alignment program. Highlighted is the longest overlap found.**

Discussion

Utilizing staple pools to make multiple origami structures requires that there is very little sequence overlap in the scaffolds used. However, the pScaf system requires that 393 nts remain constant in every scaffold in order to produce it using an E.coli/helper phage based system[63]. This means that those staples will also overlap and therefore could not be used for data points that are different in each origami. One option to address this issue would be to take advantage of this consistency for the parts of encoding that also remain the same for each origami design. For example, this section of the origami could be utilized for the orientation markers, which do not store data and can be the same for each dNAM node. Another option would be to not include this sequence in the origami structure. In this case, we would need to make the scaffold 393 nts longer, but the overlapping sequence could just be left single stranded.

Before the full potential of array-based oligo synthesis can be fully realized in dNAM, further experiments are needed to test the limits of random access and one-pot synthesis. For example, it is not yet know if increasing the number of scaffolds will decrease the yield of origami folding or create unwanted structures[71,72]. There are also current limitations to the yield of each oligo synthesized in the pool as the number of unique oligos is increased. As DNA synthesis advances, new experiments will become possible to test these remaining questions. Nevertheless, our current results advance the efficiency of origami synthesis for dNAM, and inspire future experiments.

In addition, to take full advantage of the potential of pooled staples and one-pot synthesis will require parallel advancements in SRM and data processing. Imaging may require multiple fields of view in order to acquire enough of each origami to create a

super-high resolution averaged image. It will also require more robust averaging algorithms to separate multiple origami patterns as the number of distinct patterns increases. Moreover, recent advances in DNA-PAINT technology could be utilized to make reading more efficient and enable us to distinguish multiple structures more easily. Overall, we envision that dNAM can readily be advanced by taking advantage of rapidly developing related technologies.

Another question that this research has raised is how to optimize scaffold sequence. We were curious to see that the origami made with M13mp18 had the same DNA-PAINT data points missing whether the structure was formed with pooled or individual staple strands. This suggests that it might be the scaffold sequence that is causing problems with the structure. Pooled staples would also be useful to investigate this, by making it quick and easy to purchase different sets of staples for different rotations of the scaffold. For example, instead of starting the scaffold sequence at the last helix, we could start it at the first helix; this way we would be using the same scaffold sequence but it would be arranged completely differently in the structure. Each structure would require a completely different set of staples, so purchasing pooled staples would allow us to test many more iterations of the same structure with a different scaffold rotation to test sequence effects on structure formation and performance. If the missing data points migrate throughout the structure with a specific sequence, we could avoid using that sequence at a data point. This scaffold rotation could also be useful to evaluate, and/or be guided by sequence analysis software that is currently being developed to examine intra-scaffold, intra-staple, and inter-scaffold-staple folding thermodynamics.[73]

Beyond using oligo pools for dNAM, we envision that this platform has room to expand into three dimensions. This could be achieved by including multiple imager binding sites per data strand, with the third dimension "axial" position being the distance from the surface of the origami, with a quencher embedded, that can be determined by fluorescence lifetime microscopy. 3D NAM will require the development of custom algorithms to encode and decode multiple bits per data strand including a deep neural network approach for image processing and a Bayes-optimal algorithm for data recovery and error correction. In addition, we will need to advance time-resolved super-resolution microscopy to read data at a resolution of 5x5x1 nm using custom imaging arrays to integrate DNA-PAINT and fluorescence lifetime microscopy. Together, with the work in this dissertation, these advancements would set the stage for dNAM to achieve TB levels of data storage capacity.

Appendix

Pool comparison stats-

Source	Oligo size (nts)	# oligos/pool	yield	error rate
Gen script	up to 170	NR	1 fmol	~0.5%
IDT	40-350	up to 20,000	up to 50 pmol	NR
Twist	up to 300	no limit	>0.2 fmol	up to 1:2000
Agilent	30-230	244,000	NR	NR

*NR- not reported per company website

Scaffold Smith user constraints-

INCLUDE
31,151|AATAGTGGACTCTTGTTCCAAACTGGAACAACACTCAACCCTATCTCG
GGCTATTCTTTTGATTTATAAGGGATTTTGCCGATTTCGGGGTACCAAA
30,266|AAAGGATCCACGCGCCCTGTAGCGGCGCATTAAGCGCGGCGGGTGTG
GTGGTTACGCGCAGCGTGACCGCTACACTTGCCAGCGCCCTAGCGCCCGCTC
CTTTCGCTTTCTTCCCTTCCTTTCTCGCCACGTTCGCCGGCTTTCCCCGTCAAG
CTCTAAATCGGGGGCTCCCTTTAGGGTTCCGATTTAGTGCTTTACGGCACCTC
GACCCCAAAAAACTTGATTTGGGTGATGGTTCACGTAGTGGGCCATCGCCCT
GATAGACGGTTTTTCGCCCTTTGACGTTGGAGTCCACGTTCTTT

EXCLUDE
TTGGGAGGA
AACCCTCCT
GGGAGGA
CCCTCCT

11KB sequences link- https://benchling.com/sekobernat/f_/N4b0s9gV-ss-scaffold-analysis/

Sequence alignment- https://benchling.com/sekobernat/f/lib_N4b0s9gV-ss-scaffold-analysis/seq_co4eZiCr-8x10nam_ss_pscafbb1/edit?alignment=seqanl_Ty8GMXxh

pScaf-11054.1 Scaffold and staple sequences can be found in Chapter 2 supplementary information.

M13mp18 scaffold sequence-

AATGCTACTACTATTAGTAGAATTGATGCCACCTTTTCAGCTCGCGCCCCAAA
TGAAAATATAGCTAAACAGGTTATTGACCATTTGCGAAATGTATCTAATGGTC
AAACTAAATCTACTCGTTCGCAGAATTGGGAATCAACTGTTATATGGAATGA
AACTTCCAGACACCGTACTTTAGTTGCATATTTAAAACATGTTGAGCTACAGC
ATTATATTCAGCAATTAAGCTCTAAGCCATCCGCAAAAATGACCTCTTATCAA
AAGGAGCAATTAAAGGTACTCTCTAATCCTGACCTGTTGGAGTTTGCTTCCGG
TCTGGTTCGCTTTGAAGCTCGAATTAAAACGCGATATTTGAAGTCTTTCGGGC
TTCCTCTTAATCTTTTTGATGCAATCCGCTTTGCTTCTGACTATAATAGTCAGG
GTAAAGACCTGATTTTTGATTTATGGTCATTCTCGTTTTCTGAACTGTTTAAAG
CATTTGAGGGGGATTCAATGAATATTTATGACGATTCCGCAGTATTGGACGCT
ATCCAGTCTAAACATTTTACTATTACCCCCTCTGGCAAAACTTCTTTTGCAAA
AGCCTCTCGCTATTTTGGTTTTTATCGTCGTCTGGTAAACGAGGGTTATGATA
GTGTTGCTCTTACTATGCCTCGTAATTCCTTTTGGCGTTATGTATCTGCATTAG
TTGAATGTGGTATTCCTAAATCTCAACTGATGAATCTTTCTACCTGTAATAAT
GTTGTTCCGTTAGTTCGTTTTATTAACGTAGATTTTTCTTCCCAACGTCCTGAC
TGGTATAATGAGCCAGTTCTTAAAATCGCATAAGGTAATTCACAATGATTAA
AGTTGAAATTAAACCATCTCAAGCCCAATTTACTACTCGTTCTGGTGTTTCTC
GTCAGGGCAAGCCTTATTCACTGAATGAGCAGCTTTGTTACGTTGATTTGGGT
AATGAATATCCGGTTCTTGTCAAGATTACTCTTGATGAAGGTCAGCCAGCCTA
TGCGCCTGGTCTGTACACCGTTCATCTGTCCTCTTTCAAAGTTGGTCAGTTCG
GTTCCCTTATGATTGACCGTCTGCGCCTCGTTCCGGCTAAGTAACATGGAGCA
GGTCGCGGATTTCGACACAATTTATCAGGCGATGATACAAATCTCCGTTGTAC
TTTGTTTCGCGCTTGGTATAATCGCTGGGGGTCAAAGATGAGTGTTTTAGTGT
ATTCTTTTGCCTCTTTCGTTTTAGGTTGGTGCCTTCGTAGTGGCATTACGTATT
TTACCCGTTTAATGGAAACTTCCTCATGAAAAAGTCTTTAGTCCTCAAAGCCT
CTGTAGCCGTTGCTACCCTCGTTCCGATGCTGTCTTTCGCTGCTGAGGGTGAC
GATCCCGCAAAAGCGGCCTTTAACTCCCTGCAAGCCTCAGCGACCGAATATA
TCGGTTATGCGTGGGCGATGGTTGTTGTCATTGTCGGCGCAACTATCGGTATC
AAGCTGTTTAAGAAATTCACCTCGAAAGCAAGCTGATAAACCGATACAATTA
AAGGCTCCTTTTGGAGCCTTTTTTTTGGAGATTTTCAACGTGAAAAAATTATT
ATTCGCAATTCCTTTAGTTGTTCCTTTCTATTCTCACTCCGCTGAAACTGTTGA
AAGTTGTTTAGCAAAATCCCATACAGAAAATTCATTTACTAACGTCTGGAAA
GACGACAAAACTTTAGATCGTTACGCTAACTATGAGGGCTGTCTGTGGAATG
CTACAGGCGTTGTAGTTTGTACTGGTGACGAAACTCAGTGTTACGGTACATGG
GTTCCTATTGGGCTTGCTATCCCTGAAAATGAGGGTGGTGGCTCTGAGGGTGG
CGGTTCTGAGGGTGGCGGTTCTGAGGGTGGCGGTACTAAACCTCCTGAGTAC
GGTGATACACCTATTCCGGGCTATACTTATATCAACCCTCTCGACGGCACTTA
TCCGCCTGGTACTGAGCAAAACCCCGCTAATCCTAATCCTTCTCTTGAGGAGT
CTCAGCCTCTTAATACTTTCATGTTTCAGAATAATAGGTTCCGAAATAGGCAG
GGGGCATTAACTGTTTATACGGGCACTGTTACTCAAGGCACTGACCCCGTTAA
AACTTATTACCAGTACACTCCTGTATCATCAAAAGCCATGTATGACGCTTACT
GGAACGGTAAATTCAGAGACTGCGCTTTCCATTCTGGCTTTAATGAGGATTTA
TTTGTTTGTGAATATCAAGGCCAATCGTCTGACCTGCCTCAACCTCCTGTCAA
TGCTGGCGGCGGCTCTGGTGGTGGTTCTGGTGGCGGCTCTGAGGGTGGTGGC

TCTGAGGGTGGCGGTTCTGAGGGTGGCGGCTCTGAGGGAGGCGGTTCCGGTG
GTGGCTCTGGTTCCGGTGATTTTGATTATGAAAAGATGGCAAACGCTAATAA
GGGGGCTATGACCGAAAATGCCGATGAAAACGCGCTACAGTCTGACGCTAAA
GGCAAACTTGATTCTGTCGCTACTGATTACGGTGCTGCTATCGATGGTTTCAT
TGGTGACGTTTCCGGCCTTGCTAATGGTAATGGTGCTACTGGTGATTTTGCTG
GCTCTAATTCCCAAATGGCTCAAGTCGGTGACGGTGATAATTCACCTTTAATG
AATAATTTCCGTCAATATTTACCTTCCCTCCCTCAATCGGTTGAATGTCGCCCT
TTTGTCTTTGGCGCTGGTAAACCATATGAATTTTCTATTGATTGTGACAAAAT
AAACTTATTCCGTGGTGTCTTTGCGTTTCTTTTATATGTTGCCACCTTTATGTA
TGTATTTTCTACGTTTGCTAACATACTGCGTAATAAGGAGTCTTAATCATGCC
AGTTCTTTTGGGTATTCCGTTATTATTGCGTTTCCTCGGTTTCCTTCTGGTAAC
TTTGTTCGGCTATCTGCTTACTTTTCTTAAAAAGGGCTTCGGTAAGATAGCTA
TTGCTATTTCATTGTTTCTTGCTCTTATTATTGGGCTTAACTCAATTCTTGTGG
GTTATCTCTCTGATATTAGCGCTCAATTACCCTCTGACTTTGTTCAGGGTGTTC
AGTTAATTCTCCCGTCTAATGCGCTTCCCTGTTTTTATGTTATTCTCTCTGTAA
AGGCTGCTATTTTCATTTTTGACGTTAAACAAAAAATCGTTTCTTATTTGGATT
GGGATAAATAATATGGCTGTTTATTTTGTAACTGGCAAATTAGGCTCTGGAAA
GACGCTCGTTAGCGTTGGTAAGATTCAGGATAAAATTGTAGCTGGGTGCAAA
ATAGCAACTAATCTTGATTTAAGGCTTCAAAACCTCCCGCAAGTCGGGAGGT
TCGCTAAAACGCCTCGCGTTCTTAGAATACCGGATAAGCCTTCTATATCTGAT
TTGCTTGCTATTGGGCGCGGTAATGATTCCTACGATGAAAATAAAAACGGCTT
GCTTGTTCTCGATGAGTGCGGTACTTGGTTTAATACCCGTTCTTGGAATGATA
AGGAAAGACAGCCGATTATTGATTGGTTTCTACATGCTCGTAAATTAGGATG
GGATATTATTTTTCTTGTTCAGGACTTATCTATTGTTGATAAACAGGCGCGTTC
TGCATTAGCTGAACATGTTGTTTATTGTCGTCGTCTGGACAGAATTACTTTAC
CTTTTGTCGGTACTTTATATTCTCTTATTACTGGCTCGAAAATGCCTCTGCCTA
AATTACATGTTGGCGTTGTTAAATATGGCGATTCTCAATTAAGCCCTACTGTT
GAGCGTTGGCTTTATACTGGTAAGAATTTGTATAACGCATATGATACTAAACA
GGCTTTTTCTAGTAATTATGATTCCGGTGTTTATTCTTATTTAACGCCTTATTT
ATCACACGGTCGGTATTTCAAACCATTAAATTTAGGTCAGAAGATGAAATTA
ACTAAAATATATTTGAAAAAGTTTTCTCGCGTTCTTTGTCTTGCGATTGGATTT
GCATCAGCATTTACATATAGTTATATAACCCAACCTAAGCCGGAGGTTAAAA
AGGTAGTCTCTCAGACCTATGATTTTGATAAATTCACTATTGACTCTTCTCAG
CGTCTTAATCTAAGCTATCGCTATGTTTTCAAGGATTCTAAGGGAAAATTAAT
TAATAGCGACGATTTACAGAAGCAAGGTTATTCACTCACATATATTGATTTAT
GTACTGTTTCCATTAAAAAAGGTAATTCAAATGAAATTGTTAAATGTAATTAA
TTTTGTTTTCTTGATGTTTGTTTCATCATCTTCTTTTGCTCAGGTAATTGAAATG
AATAATTCGCCTCTGCGCGATTTTGTAACTTGGTATTCAAAGCAATCAGGCGA
ATCCGTTATTGTTTCTCCCGATGTAAAAGGTACTGTTACTGTATATTCATCTGA
CGTTAAACCTGAAAATCTACGCAATTTCTTTATTTCTGTTTTACGTGCAAATA
ATTTTGATATGGTAGGTTCTAACCCTTCCATTATTCAGAAGTATAATCCAAAC
AATCAGGATTATATTGATGAATTGCCATCATCTGATAATCAGGAATATGATG
ATAATTCCGCTCCTTCTGGTGGTTTCTTTGTTCCGCAAAATGATAATGTTACTC
AAACTTTTAAAATTAATAACGTTCGGGCAAAGGATTTAATACGAGTTGTCGA
ATTGTTTGTAAAGTCTAATACTTCTAAATCCTCAAATGTATTATCTATTGACG
GCTCTAATCTATTAGTTGTTAGTGCTCCTAAAGATATTTTAGATAACCTTCCTC

AATTCCTTTCAACTGTTGATTTGCCAACTGACCAGATATTGATTGAGGGTTTG
ATATTTGAGGTTCAGCAAGGTGATGCTTTAGATTTTTCATTTGCTGCTGGCTCT
CAGCGTGGCACTGTTGCAGGCGGTGTTAATACTGACCGCCTCACCTCTGTTTT
ATCTTCTGCTGGTGGTTCGTTCGGTATTTTTAATGGCGATGTTTTAGGGCTATC
AGTTCGCGCATTAAAGACTAATAGCCATTCAAAAATATTGTCTGTGCCACGTA
TTCTTACGCTTTCAGGTCAGAAGGGTTCTATCTCTGTTGGCCAGAATGTCCCT
TTTATTACTGGTCGTGTGACTGGTGAATCTGCCAATGTAAATAATCCATTTCA
GACGATTGAGCGTCAAAATGTAGGTATTTCCATGAGCGTTTTTCCTGTTGCAA
TGGCTGGCGGTAATATTGTTCTGGATATTACCAGCAAGGCCGATAGTTTGAGT
TCTTCTACTCAGGCAAGTGATGTTATTACTAATCAAAGAAGTATTGCTACAAC
GGTTAATTTGCGTGATGGACAGACTCTTTTACTCGGTGGCCTCACTGATTATA
AAAACACTTCTCAGGATTCTGGCGTACCGTTCCTGTCTAAAATCCCTTTAATC
GGCCTCCTGTTTAGCTCCCGCTCTGATTCTAACGAGGAAAGCACGTTATACGT
GCTCGTCAAAGCAACCATAGTACGCGCCCTGTAGCGGCGCATTAAGCGCGGC
GGGTGTGGTGGTTACGCGCAGCGTGACCGCTACACTTGCCAGCGCCCTAGCG
CCCGCTCCTTTCGCTTTCTTCCCTTCCTTTCTCGCCACGTTCGCCGGCTTTCCC
CGTCAAGCTCTAAATCGGGGGCTCCCTTTAGGGTTCCGATTTAGTGCTTTACG
GCACCTCGACCCCAAAAAACTTGATTTGGGTGATGGTTCACGTAGTGGGCCA
TCGCCCTGATAGACGGTTTTTCGCCCTTTGACGTTGGAGTCCACGTTCTTTAAT
AGTGGACTCTTGTTCCAAACTGGAACAACACTCAACCCTATCTCGGGCTATTC
TTTTGATTTATAAGGGATTTTGCCGATTTCGGAACCACCATCAAACAGGATTT
TCGCCTGCTGGGGCAAACCAGCGTGGACCGCTTGCTGCAACTCTCTCAGGGC
CAGGCGGTGAAGGGCAATCAGCTGTTGCCCGTCTCACTGGTGAAAAGAAAAA
CCACCCTGGCGCCCAATACGCAAACCGCCTCTCCCCGCGCGTTGGCCGATTC
ATTAATGCAGCTGGCACGACAGGTTTCCCGACTGGAAAGCGGGCAGTGAGCG
CAACGCAATTAATGTGAGTTAGCTCACTCATTAGGCACCCCAGGCTTTACACT
TTATGCTTCCGGCTCGTATGTTGTGTGGAATTGTGAGCGGATAACAATTTCAC
ACAGGAAACAGCTATGACCATGATTACGAATTCGAGCTCGGTACCCGGGGAT
CCTCTAGAGTCGACCTGCAGGCATGCAAGCTTGGCACTGGCCGTCGTTTTACA
ACGTCGTGACTGGGAAAACCCTGGCGTTACCCAACTTAATCGCCTTGCAGCA
CATCCCCCTTTCGCCAGCTGGCGTAATAGCGAAGAGGCCCGCACCGATCGCC
CTTCCCAACAGTTGCGCAGCCTGAATGGCGAATGGCGCTTTGCCTGGTTTCCG
GCACCAGAAGCGGTGCCGGAAAGCTGGCTGGAGTGCGATCTTCCTGAGGCCG
ATACTGTCGTCGTCCCCTCAAACTGGCAGATGCACGGTTACGATGCGCCCATC
TACACCAACGTGACCTATCCCATTACGGTCAATCCGCCGTTTGTTCCCACGGA
GAATCCGACGGGTTGTTACTCGCTCACATTTAATGTTGATGAAAGCTGGCTAC
AGGAAGGCCAGACGCGAATTATTTTTGATGGCGTTCCTATTGGTTAAAAAAT
GAGCTGATTTAACAAAAATTTAATGCGAATTTTAACAAAATATTAACGTTTAC
AATTTAAATATTTGCTTATACAATCTTCCTGTTTTTGGGGCTTTTCTGATTATC
AACCGGGGTACATATGATTGACATGCTAGTTTTACGATTACCGTTCATCGATT
CTCTTGTTTGCTCCAGACTCTCAGGCAATGACCTGATAGCCTTTGTAGATCTC
TCAAAAATAGCTACCCTCTCCGGCATTAATTTATCAGCTAGAACGGTTGAATA
TCATATTGATGGTGATTTGCTGTCTCCGGCCTTTCTCACCCTTTTGAATCTTTA
CCTACACATTACTCAGGCATTGCATTTAAAATATATGAGGGTTCAAAAATTTT
TATCCTTGCGTTGAAATAAAGGCTTCTCCCGCAAAAGTATTACAGGGTCATAA

TGTTTTTGGTACAACCGATTTAGCTTTATGCTCTGAGGCTTTATTGCTTAATTT
TGCTAATTCTTTGCCTTGCCTGTATGATTTATTGGATGTT

M13mp18 staples-

AGA AAG GAA CAA CTA AAG GAA TTC AAA AAA A
AGG CTC CAG AGG CTT TGA GGA CAC GGG TAATTGGGAGGA
AAT ACG TTT GAA AGA GGA CAG ACT GAC CTT
CAT CAA GTA AAA CGA ACT AAC GAG TTG AGATTGGGAGGA
TTT AGG ACA AAT GCT TTA AAC AAT CAG GTC
TTT ACC CCA ACA TGT TTT AAA TTT CCA TATTTGGGAGGA
AAC AGT TTT GTA CCA AAA ACA TTT TAT TTC
AAC GCA AAA TCG ATG AAC GGT ACC GGT TGATTGGGAGGA
TAA TCA GCG GAT TGA CCG TAA TCG TAA CCG
TGC ATC TTT CCC AGT CAC GAC GGC CTG CAGTTGGGAGGA
GTC GAC TTC GGC CAA CGC GCG GGG TTT TTC
TTT TCA CTC AAA GGG CGA AAA ACC ATC ACCTTGGGAGGA
ACG GCT ACA AAA GGA GCC TTT AAT GTG AGA AT
GAC CAA CTA ATG CCA CTA CGA AGG GGG TAG CA
TAC GTT AAA GTA ATC TTG ACA AGA ACC GAA CT
ATC CCC CTA TAC CAC ATT CAA CTA GAA AAA TC
CTG TAG CTT GAC TAT TAT AGT CAG TTC ATT GA
TAA ATC GGG ATT CCC AAT TCT GCG ATA TAA TG
AAC AAG AGG GAT AAA AAT TTT TAG CAT AAA GC
ACA AAC GGA AAA GCC CCA AAA ACA CTG GAG CA
CCA GGG TTG CCA GTT TGA GGG GAC CCG TGG GA
TTA ATG AAC TAG AGG ATC CCC GGG TGG GTA ACG
CTC CAA CGC AGT GAG ACG GGC AAC CAG CTG CA
CAA ATC AAG TTT TTT GGG GTC GAA ACG TGG A
ACA ACT TTC AAC AGT TTC AGC GGA TGT ATC GG
TTT ATC AGG ACA GCA TCG GAA CGA CAC CAA CCTTGGGAGGA
TAA AAC GAG GTC AAT CAT AAG GGA ACC GGA TA
TTC ATT ACG TCA GGA CGT TGG GAA ATG CAG ATTTGGGAGGA
ACA TAA CGG GAA TCG TCA TAA ATA AAG CAA AG
CGG ATT GCA GAG CTT AAT TGC TGA AAC GAG TATTGGGAGGA
GAT TTA GTC AAT AAA GCC TCA GAG AAC CCT CA
TAT ATT TTG TCA TTG CCT GAG AGT GGA AGA TTTTGGGAGGA
GTA TAA GCC AAC CCG TCG GAT TCT GAC GAC AG
TAT CGG CCG CAA GGC GAT TAA GTT TAC CGA GCTTGGGAGGA
TCG AAT TCG GGA AAC CTG TCG TGC AGC TGA TT
GCC CTT CAG AGT CCA CTA TTA AAG GGT GCC GTTTGGGAGGA
CAG CGA AAC TTG CTT TCG AGG TGT TGC TAA
GCG CAG ACA AGA GGC AAA AGA ATC CCT CAG

TTA TAC CAC CAA ATC AAC GTA ACG AAC GAG
AAT ACT GCC CAA AAG GAA TTA CGT GGC TCA
GAT GGC TTA TCA AAA AGA TTA AGA GCG TCC
AAA TTA AGT TGA CCA TTA GAT ACT TTT GCG
GCT ATC AGA AAT GCA ATG CCT GAA TTA GCA
GCG AGT AAA AAT ATT TAA ATT GTT ACA AAG
GAT GTG CTT CAG GAA GAT CGC ACA ATG TGA
TTC CAG TCG TAA TCA TGG TCA TAA AAG GGG
TGG AAC AAC CGC CTG GCC CTG AGG CCC GCT
AAA GCA CTA AAT CGG AAC CCT AAT CCA GTT
TAA ATG AAT TTT CTG TAT GGG ATT AAT TTC TT
AAA CAG CTT TTT GCG GGA TCG TCA ACA CTA AATTGGGAGGA
ACA CTC ATC CAT GTT ACT TAG CCG AAA GCT GC
TCA TTC AGA TGC GAT TTT AAG AAC AGG CAT AGTTGGGAGGA
TAA GAG CAA ATG TTT AGA CTG GAT AGG AAG CC
CGA AAG ACT TTG ATA AGA GGT CAT ATT TCG CATTGGGAGGA
AAT GGT CAA CAG GCA AGG CAA AGA GTA ATG TG
TAG GTA AAC TAT TTT TGA GAG ATC AAA CGT TATTGGGAGGA
ATA TTT TGG CTT TCA TCA ACA TTA TCC AGC CA
GCT TTC CGA TTA CGC CAG CTG GCG GCT GTT TCTTGGGAGGA
CTG TGT GAT TGC GTT GCG CTC ACT AGA GTT GC
AGC AAG CGT AGG GTT GAG TGT TGT AGG GAG CCTTGGGAGGA
AAG GCC GCT GAT ACC GAT AGT TGC GAC GTT AG
GAC CTG CTC TTT GAC CCC CAG CGA GGG AGT TA
ATT ACC TTT GAA TAA GGC TTG CCC AAA TCC GC
AAT AGT AAA CAC TAT CAT AAC CCT CAT TGT GA
TTG CTC CTT TCA AAT ATC GCG TTT GAG GGG GT
TAA ATC ATA TAA CCT GTT TAG CTA ACC TTT AA
GAG GGT AGG ATT CAA AAG GGT GAG ACA TCC AA
TGT AGC CAT TAA AAT TCG CAT TAA ATG CCG GA
TCT TCG CTG CAC CGC TTC TGG TGC GGC CTT CC
CAC ATT AAA ATT GTT ATC CGC TCA TGC GGG CC
GCC CGA GAG TCC ACG CTG GTT TGC AGC TAA CT
CCC GAT TTA GAG CTT GAC GGG GAA AAA GAA TA
TCT AAA GTT TTG TCG TCT TTC CAG CCG ACA A
TGA CAA CTC GCT GAG GCT TGC ATT ATA CCATTGGGAGGA
AGC GCG ATG ATA AAT TGT GTC GTG ACG AGA
AAC ACC AAA TTT CAA CTT TAA TCG TTT ACCTTGGGAGGA
AGA CGA CAA AGA AGT TTT GCC ATA ATT CGA
GCT TCA ATC AGG ATT AGA GAG TTA TTT TCATTGGGAGGA
TTT GGG GAT AGT AGT AGC ATT AAA AGG CCG
GAG ACA GCT AGC TGA TAA ATT AAT TTT TGTTTGGGAGGA
TAA ATC AAA ATA ATT CGC GTC TCG GAA ACC

AGG CAA AGG GAA GGG CGA TCG GCA ATT CCATTGGGAGGA
CAC AAC AGG TGC CTA ATG AGT GCC CAG CAG
GCG AAA AAT CCC TTA TAA ATC AAG CCG GCGTTGGGAGGA
TCC ACA GAC AGC CCT CAT AGT TAG CGT AAC GA
TTA GGA TTG GCT GAG ACT CCT CAA TAA CCG AT
TTG ACA GGC CAC CAC CAG AGC CGC GAT TTG TA
GCA AGG CCT CAC CAG TAG CAC CAT GGG CTT GA
TTA TTA CGA AGA ACT GGC ATG ATT GCG AGA GG
AGA GAG AAA AAA ATG AAA ATA GCA AGC AAA CT
CCA ATA GCT CAT CGT AGG AAT CAT GGC ATC AA
GTA ATA AGT TAG GCA GAG GCA TTT ATG ATA TT
ATC GCA AGT ATG TAA ATG CTG ATG ATA GGA AC
AGA AAA CAA AGA AGA TGA TGA AAC AGG CTG CG
GCA ATT CAC ATA TTC CTG ATT ATC AAA GTG TA
TCA ATA TCG AAC CTC AAA TAT CAA TTC CGA AA
ATA TTC GGA ACC ATC GCC CAC GCA GAG AAG GA
TCA TCG CCA ACA AAG TAC AAC GGA CGC CAG CATTGGGAGGA
GAT GGT TTG AAC GAG TAG TAA ATT TAC CAT TA
CTT TTG CAG ATA AAA ACC AAA ATA AAG ACT CCTTGGGAGGA
CCA ACA GGA GCG AAC CAG ACC GGA GCC TTT AC
TTC TAC TAC GCG AGC TGA AAA GGT TAC CGC GCTTGGGAGGA
CAA CCG TTT CAA ATC ACC ATC AAT TCG AGC CA
GCC ATC AAG CTC ATT TTT TAA CCA CAA ATC CATTGGGAGGA
CAA CTG TTG CGC CAT TCG CCA TTC AAA CAT CA
AAG CCT GGT ACG AGC CGG AAG CAT AGA TGA TGTTGGGAGGA
TCG GCA AAT CCT GTT TGA TGG TGG ACC CTC AA
AAC GTG GCG AGA AAG GAA GGG AAA CCA GTA ATTGGGAGGA
TAT TAA GAA GCG GGG TTT TGC TCG TAG CAT
CAC CAG AAA GGT TGA GGC AGG TCA TGA AAG
CAG CAA AAG GAA ACG TCA CCA ATG AGC CGC
ATA CCC AAC AGT ATG TTA GCA AAT TAG AGC
TTA ACG TCT AAC ATA AAA ACA GGT AAC GGA
TTT TAT TTA AGC AAA TCA GAT ATT TTT TGT
CAT GTA ATA GAA TAT AAA GTA CCA AGC CGT
TAT AAC TAA CAA AGA ACG CGA GAA CGC CAA
CTG AGC AAA AAT TAA TTA CAT TTT GGG TTA
ATT ATC ATT CAA TAT AAT CCT GAC AAT TAC
ACC TTG CTT GGT CAG TTG GCA AAG AGC GGA
TAA AAG GGA CAT TCT GGC CAA CAA AGC ATC
TCA CCA GTA CAA ACT ACA ACG CCT AGT ACC AG
GCG GAT AAC CTA TTA TTC TGA AAC AGA CGA TTTTGGGAGGA
GGC CTT GAA GAG CCA CCA CCC TCA GAA ACC AT
CGA TAG CAT TGA GCC ATT TGG GAA CGT AGA AATTGGGAGGA

ATA CAT ACC GAG GAA ACG CAA TAA GAA GCG CA
TTA GAC GGC CAA ATA AGA AAC GAT AGA AGG CTTTGGGAGGA
TAT CCG GTC TCA TCG AGA ACA AGC GAC AAA AG
GTA AAG TAA TCG CCA TAT TTA ACA AAA CTT TTTTGGGAGGA
TCA AAT ATA ACC TCC GGC TTA GGT AAC AAT TT
CAT TTG AAG GCG AAT TAT TCA TTT TTG TTT GGTTGGGAGGA
ATT ATA CTA AGA AAC CAC CAG AAG TCA ACA GT
TGA AAG GAG CAA ATG AAA AAT CTA GAG ATA GATTGGGAGGA
TTT CGG AAG TGC CGT CGA GAG GGT GAG TTT CG
CCA CCC TCT ATT CAC AAA CAA ATA CCT GCC TA
TCA CCG ACG CAC CGT AAT CAG TAG CAG AAC CG
AAG GAA ACA TAA AGG TGG CAA CAT TAT CAC CG
ATC CCA ATG AGA ATT AAC TGA ACA GTT ACC AG
GTA CCG CAA TTC TAA GAA CGC GAG TAT TAT TT
AAT TGA GAA TTC TGT CCA GAC GAC TAA ACC AA
ACC TTT TTA TTT TAG TTA ATT TCA TAG GGC TT
CGC GCA GAT TAC CTT TTT TAA TGG GAG AGA CT
GCG GAA CAT CTG AAT AAT GGA AGG TAC AAA AT
AGC CAG CAA TTG AGG AAG GTT ATC ATC ATT TT
ACC CTT CTG ACC TGA AAG CGT AAG ACG CTG AG
AGG AAC CCA TGT ACC GTA ACA CTT GAT ATA A
GTA TAG CAA ACA GTT AAT GCC CAA TCC TCATTGGGAGGA
TTA AAG CCA GAG CCG CCA CCC TCG ACA GAA
TCA AGT TTC ATT AAA GGT GAA TAT AAA AGATTGGGAGGA
AAC GCA AAG ATA GCC GAA CAA ACC CTG AAC
AAA GTC ACA AAA TAA ACA GCC AGC GTT TTATTGGGAGGA
GCG AAC CTC CAA GAA CGG GTA TGA CAA TAA
ACA ACA TGC CAA CGC TCA ACA GTC TTC TGATTGGGAGGA
CCT AAA TCA AAA TCA TAG GTC TAA ACA GTA
CAT AAA TCT TTG AAT ACC AAG TGT TAG AACTTGGGAGGA
CTA CCA TAG TTT GAG TAA CAT TTA AAA TAT
CTT TAG GGC CTG CAA CAG TGC CAA TAC GTGTTGGGAGGA
GCC CGT ATC CGG AAT AGG TGT ATC AGC CCA AT
GCC TCC CTC AGA ATG GAA AGC GCA GTA ACA GT
GAA ATT ATT GCC TTT AGC GTC AGA CCG GAA CC
AAG TAA GCA GAC ACC ACG GAA TAA TAT TGA CG
GCC AGT TAG AGG GTA ATT GAG CGC TTT AAG AA
CTT ATC ATT CCC GAC TTG CGG GAG CCT AAT TT
AGT ATA AAG TTC AGC TAA TGC AGA TGT CTT TC
GAA TTT ATT TAA TGG TTT GAA ATA TTC TTA CC
CCT GAT TGC AAT ATA TGT GAG TGA TCA ATA GT
ATT TTA AAA TCA AAA TTA TTT GCA CGG ATT CG
TTA ACA CCA GCA CTA ACA ACT AAT CGT TAT TA

GCA CAG ACA ATA TTT TTG AAT GGG GTC AGT A
CCA CCC TCA TTT TCA GGG ATA GCA ACC GTA CT
CAG GAG GTG GGG TCA GTG CCT TGA GTC TCT GATTGGGAGGA
ATT TAC CGG GAA CCA GAG CCA CCA CTG TAG CG
CGT TTT CAA GGG AGG GAA GGT AAA GTT TAT TTTTGGGAGGA
TGT CAC AAT CTT ACC GAA GCC CTT TAA TAT CA
GAG AGA TAG AGC GTC TTT CCA GAG GTT TTG AATTGGGAGGA
GCC TTA AAC CAA TCA ATA ATC GGC ACG CGC CT
GTT TAT CAA TAT GCG TTA TAC AAA CCG ACC GTTTGGGAGGA
GTG ATA AAA AGA CGC TGA GAA GAG ATA ACC TT
GCT TCT GTT CGG GAG AAA CAA TAA CGT AAA ACTTGGGAGGA
AGA AAT AAA AAT CCT TTG CCC GAA AGA TTA GA
GCC GTC AAA AAA CAG AGG TGA GGC CTA TTA GTTTGGGAGGA
GTT TTA ACT TAG TAC CGC CAC CCA GAG CCA
AAA TCA CCT TCC AGT AAG CGT CAG TAA TAA
ACC GAT TGT CGG CAT TTT CGG TCA TAA TCA
AAT AGC TAT CAA TAG AAA ATT CAA CAT TCA
ACG CTA ACA CCC ACA AGA ATT GAA AAT AGC
TGT AGA AAT CAA GAT TAG TTG CTC TTA CCA
TTA GTA TCA CAA TAG ATA AGT CCA CGA GCA
CTT AGA TTT AAG GCG TTA AAT AAA GCC TGT
CTT TTA CAA AAT CGT CGC TAT TAG CGA TAG
CTC GTA TTA GAA ATT GCG TAG ATA CAG TAC
CAG AAG ATT AGA TAA TAC ATT TGT CGA CAA
CTT TAA TGC GCG AAC TGA TAG CCC CAC CAG

REFERENCES

1. Seeman, N. C. DNA Nanotechnology at 40. *Nano Lett.* **20**, 1477–1478 (2020).
2. Seeman, N. C. & Lukeman, P. S. Nucleic acid nanostructures: bottom-up control of geometry on the nanoscale. *Rep. Prog. Phys.* **68**, 237–270 (2005).
3. Ramakrishnan, S., Ijäs, H., Linko, V. & Keller, A. Structural stability of DNA origami nanostructures under application-specific conditions. *Comput. Struct. Biotechnol. J.* **16**, 342–349 (2018).
4. Wang, P. *et al.* Practical aspects of structural and dynamic DNA nanotechnology. *MRS Bull.* **42**, 889–896 (2017).
5. Ma, W. *et al.* The biological applications of DNA nanomaterials: current challenges and future directions. *Signal Transduct. Target. Ther.* **6**, 351 (2021).
6. Dey, S. *et al.* DNA origami. *Nat. Rev. Methods Primer* **1**, 13 (2021).
7. Wang, W., Arias, D. S., Deserno, M., Ren, X. & Taylor, R. E. Emerging applications at the interface of DNA nanotechnology and cellular membranes: Perspectives from biology, engineering, and physics. *APL Bioeng.* **4**, 041507 (2020).
8. Zadegan, R. M. & Norton, M. L. Structural DNA Nanotechnology: From Design to Applications. *Int. J. Mol. Sci.* **13**, 7149–7162 (2012).
9. Seeman, N. C. Nucleic acid junctions and lattices. *J. Theor. Biol.* **99**, 237–247 (1982).
10. Zhang, S., Fu, T. J. & Seeman, N. C. Symmetric immobile DNA branched junctions. *Biochemistry* **32**, 8062–8067 (1993).
11. Rothemund, P. W. K. Folding DNA to create nanoscale shapes and patterns. *Nature* **440**, 297–302 (2006).

12. Douglas, S. M. *et al.* Self-assembly of DNA into nanoscale three-dimensional shapes. *Nature* **459**, 414–418 (2009).

13. Douglas, S. M. *et al.* Rapid prototyping of 3D DNA-origami shapes with caDNAno. *Nucleic Acids Res.* **37**, 5001–5006 (2009).

14. Said, H. *et al.* M1.3 – a small scaffold for DNA origami. *Nanoscale* **5**, 284–290 (2013).

15. Marchi, A. N., Saaem, I., Vogen, B. N., Brown, S. & LaBean, T. H. Toward Larger DNA Origami. *Nano Lett.* **14**, 5740–5747 (2014).

16. Rothemund, P. W. K. Design of DNA origami. in *ICCAD-2005. IEEE/ACM International Conference on Computer-Aided Design, 2005.* 471–478 (IEEE, 2005). doi:10.1109/ICCAD.2005.1560114.

17. Zhang, F. *et al.* Complex wireframe DNA origami nanostructures with multi-arm junction vertices. *Nat. Nanotechnol.* **10**, 779–784 (2015).

18. Han, D. *et al.* Single-stranded DNA and RNA origami. *Science* **358**, eaao2648 (2017).

19. Engelhardt, F. A. S. *et al.* Custom-Size, Functional, and Durable DNA Origami with Design-Specific Scaffolds. *ACS Nano* **13**, 5015–5027 (2019).

20. Bush, J. *et al.* Synthesis of DNA Origami Scaffolds: Current and Emerging Strategies. *Molecules* **25**, 3386 (2020).

21. Nafisi, P. M., Aksel, T. & Douglas, S. M. Construction of a novel phagemid to produce custom DNA origami scaffolds. *Synth. Biol. Oxf. Engl.* **3**, (2018).

22. Chen, X. *et al.* Self-Assembly of Large DNA Origami with Custom-Designed Scaffolds. *ACS Appl. Mater. Interfaces* **10**, 24344–24348 (2018).

23. Wagenbauer, K. F. *et al.* How We Make DNA Origami. *Chembiochem Eur. J. Chem. Biol.* **18**, 1873–1885 (2017).

24. Doty, D., Lee, B. L. & Stérin, T. scadnano: A browser-based, scriptable tool for designing DNA nanostructures. Preprint at http://arxiv.org/abs/2005.11841 (2020).

25. Castro, C. E. *et al.* A primer to scaffolded DNA origami. *Nat. Methods* **8**, 221–229 (2011).

26. Sengar, A., Ouldridge, T. E., Henrich, O., Rovigatti, L. & Šulc, P. A Primer on the oxDNA Model of DNA: When to Use it, How to Simulate it and How to Interpret the Results. *Front. Mol. Biosci.* **8**, 693710 (2021).

27. Majikes, J. M. & Liddle, J. A. DNA Origami Design: A How-To Tutorial. *J. Res. Natl. Inst. Stand. Technol.* **126**, 126001 (2021).

28. National Human Genome Research Institute. https://www.genome.gov/about-genomics/teaching-tools/dna-origami.

29. Martin, T. G. & Dietz, H. Magnesium-free self-assembly of multi-layer DNA objects. *Nat. Commun.* **3**, 1103 (2012).

30. Halley, P. D., Patton, R. A., Chowdhury, A., Byrd, J. C. & Castro, C. E. Low-cost, simple, and scalable self-assembly of DNA origami nanostructures. *Nano Res.* **12**, 1207–1215 (2019).

31. Sobczak, J.-P. J., Martin, T. G., Gerling, T. & Dietz, H. Rapid Folding of DNA into Nanoscale Shapes at Constant Temperature. *Science* **338**, 1458–1461 (2012).

32. Guan, C., Zhu, X. & Feng, C. DNA Nanodevice-Based Drug Delivery Systems. *Biomolecules* **11**, 1855 (2021).

33. Aye, S. & Sato, Y. Therapeutic Applications of Programmable DNA Nanostructures. *Micromachines* **13**, 315 (2022).

34. Zhang, Q. *et al.* DNA Origami as an *In Vivo* Drug Delivery Vehicle for Cancer Therapy. *ACS Nano* **8**, 6633–6643 (2014).

35. Endo, M. & Sugiyama, H. DNA Origami Nanomachines. *Molecules* **23**, 1766 (2018).

36. Berengut, J. F. *et al.* Self-Limiting Polymerization of DNA Origami Subunits with Strain Accumulation. *ACS Nano* **14**, 17428–17441 (2020).

37. Mao, C., Sun, W., Shen, Z. & Seeman, N. C. A nanomechanical device based on the B±Z transition of DNA. **397**, 3 (1999).

38. Han, X., Zhou, Z., Yang, F. & Deng, Z. Catch and Release: DNA Tweezers that Can Capture, Hold, and Release an Object under Control. *J. Am. Chem. Soc.* **130**, 14414–14415 (2008).

39. Dickinson, G. D. *et al.* An alternative approach to nucleic acid memory. *Nat. Commun.* **12**, 2371 (2021).

40. DNA Data Storage Alliance. PRESERVING OUR DIGITAL LEGACY: AN INTRODUCTION TO DNA DATA STORAGE. (2021).

41. IDC. *Worldwide Global DataSphere Forecast, 2021–2025: The World Keeps Creating More Data— Now, What Do We Do with It All?*

42. Ang, J. *et al.* Decadal Plan Executive Committee. 21.

43. Zhirnov, V., Zadegan, R. M., Sandhu, G. S., Church, G. M. & Hughes, W. L. Nucleic acid memory. *Nat. Mater.* **15**, 366–370 (2016).

44. Organick, L. *et al.* Random access in large-scale DNA data storage. *Nat. Biotechnol.* **36**, 242–248 (2018).

45. Wang, Y. *et al.* High capacity DNA data storage with variable-length Oligonucleotides using repeat accumulate code and hybrid mapping. *J. Biol. Eng.* **13**, 89 (2019).

46. Heckel, R., Mikutis, G. & Grass, R. N. A Characterization of the DNA Data Storage Channel. *Sci. Rep.* **9**, 9663 (2019).

47. Meiser, L. C. *et al.* Synthetic DNA applications in information technology. *Nat. Commun.* **13**, 352 (2022).

48. Hao, Y., Li, Q., Fan, C. & Wang, F. Data Storage Based on DNA. *Small Struct.* **2**, 2000046 (2021).

49. Chen, K. *et al.* Digital Data Storage Using DNA Nanostructures and Solid-State Nanopores. *Nano Lett.* **19**, 1210–1215 (2019).

50. Tabatabaei, S. K. *et al.* DNA punch cards for storing data on native DNA sequences via enzymatic nicking. *Nat. Commun.* **11**, 1742 (2020).

51. Zhang, Y. *et al.* DNA origami cryptography for secure communication. *Nat. Commun.* **10**, 5469 (2019).

52. Jungmann, R. *et al.* Single-Molecule Kinetics and Super-Resolution Microscopy by Fluorescence Imaging of Transient Binding on DNA Origami. *Nano Lett.* **10**, 4756–4761 (2010).

53. Nieves, D., Gaus, K. & Baker, M. DNA-Based Super-Resolution Microscopy: DNA-PAINT. *Genes* **9**, 621 (2018).

54. Stein, J. *et al.* Toward Absolute Molecular Numbers in DNA-PAINT. *Nano Lett.* **19**, 8182–8190 (2019).

55. Eklund, A. S., Comberlato, A., Parish, I. A., Jungmann, R. & Bastings, M. M. C. Quantification of Strand Accessibility in Biostable DNA Origami with Single-Staple Resolution. *ACS Nano* **15**, 17668–17677 (2021).

56. Ramakrishnan, S. Atomic force microscopy studies of DNA origami nanostructures: from structural stability to molecular patterning. 138.

57. Strauss, M. T., Schueder, F., Haas, D., Nickels, P. C. & Jungmann, R. Quantifying absolute addressability in DNA origami with molecular resolution. *Nat. Commun.* **9**, 1600 (2018).

58. Lin, R., Clowsley, A. H., Lutz, T., Baddeley, D. & Soeller, C. 3D super-resolution microscopy performance and quantitative analysis assessment using DNA-PAINT and DNA origami test samples. *Methods* **174**, 56–71 (2020).

59. Strauss, S. & Jungmann, R. Up to 100-fold speed-up and multiplexing in optimized DNA-PAINT. *Nat. Methods* **17**, 789–791 (2020).

60. *Gartner Market Trends: Evolving Enterprise Data Requirements —How Much Is Not Enough?*

61. Fujifilm TCO tool.

62. Aghebat Rafat, A., Pirzer, T., Scheible, M. B., Kostina, A. & Simmel, F. C. Surface-Assisted Large-Scale Ordering of DNA Origami Tiles. *Angew. Chem. Int. Ed.* **53**, 7665–7668 (2014).

63. Nafisi, P. M., Aksel, T. & Douglas, S. M. Construction of a novel phagemid to produce custom DNA origami scaffolds. *Synth. Biol.* **3**, ysy015 (2018).

64. Vieira, J. & Messing, J. [1] Production of single-stranded plasmid DNA. in *Recombinant DNA Part D* vol. 153 3–11 (Academic Press, 1987).

65. Kosuri, S. *et al.* Scalable gene synthesis by selective amplification of DNA pools from high-fidelity microchips. *Nat. Biotechnol.* **28**, 1295–1299 (2010).

66. Kick, B., Praetorius, F., Dietz, H. & Weuster-Botz, D. Efficient Production of Single-Stranded Phage DNA as Scaffolds for DNA Origami. *Nano Lett.* **15**, 4672–4676 (2015).

67. Green, C. M. Nanoscale Optical and Correlative Microscopies for Quantitative Characterization of DNA Nanostructures. (Boise State University, 2019).

68. Ovesný, M., Křížek, P., Borkovec, J., Švindrych, Z. & Hagen, G. M. ThunderSTORM: a comprehensive ImageJ plug-in for PALM and STORM data analysis and super-resolution imaging. *Bioinformatics* **30**, 2389–2390 (2014).

69. Schnitzbauer, J., Strauss, M. T., Schlichthaerle, T., Schueder, F. & Jungmann, R. Super-resolution microscopy with DNA-PAINT. *Nat. Protoc.* **12**, 1198–1228 (2017).

70. Dickinson, G. D. *et al.* An alternative approach to nucleic acid memory. *Nat. Commun.* **12**, 2371 (2021).

71. Majikes, J. M., Nash, J. A. & LaBean, T. H. Competitive annealing of multiple DNA origami: formation of chimeric origami. *New J. Phys.* **18**, 115001 (2016).

72. Dunn, K. E. *et al.* Guiding the folding pathway of DNA origami. *Nature* **525**, 82–86 (2015).

73. Ben Shirt-Ediss, Jordan Connolly, Emanuela Torelli, Silvia Navarro, Juan Elezgaray, Jaume Bacardit and Natalio Krasnogor. Multi-Objective Sequence Selection for Scaffolded Origami Nanostructures. in.