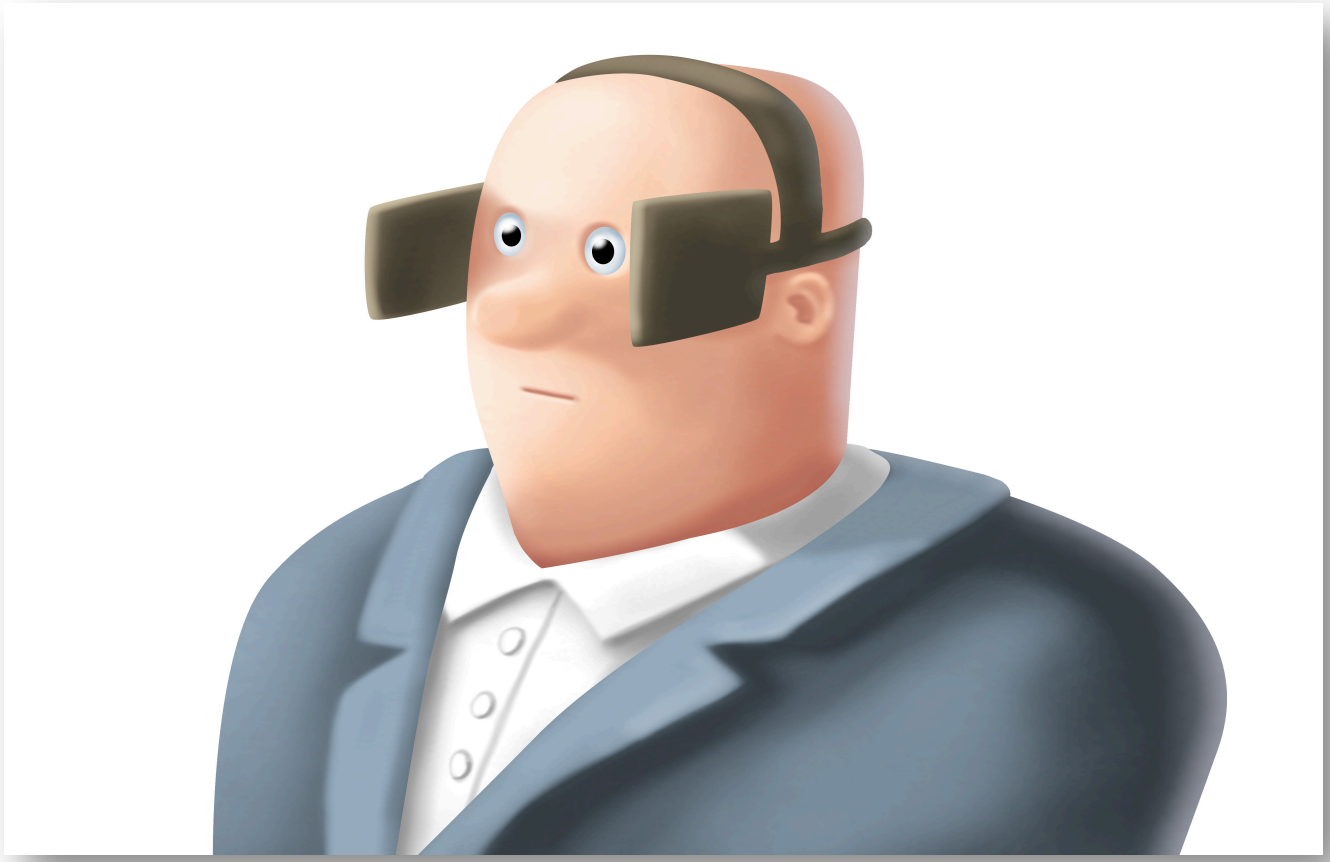
Eric Henderson

Santa Fe Institute | 2017

Scientific Adventurism



[don’t be this fellow…]

Hendersoniism

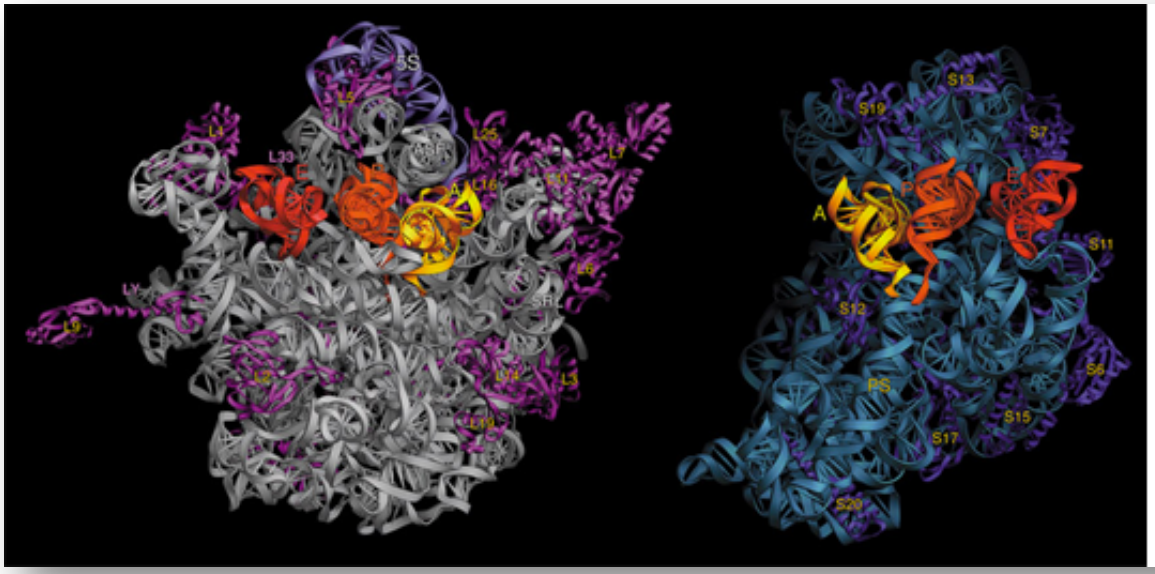
I’m no genius…

I just like making stuff!

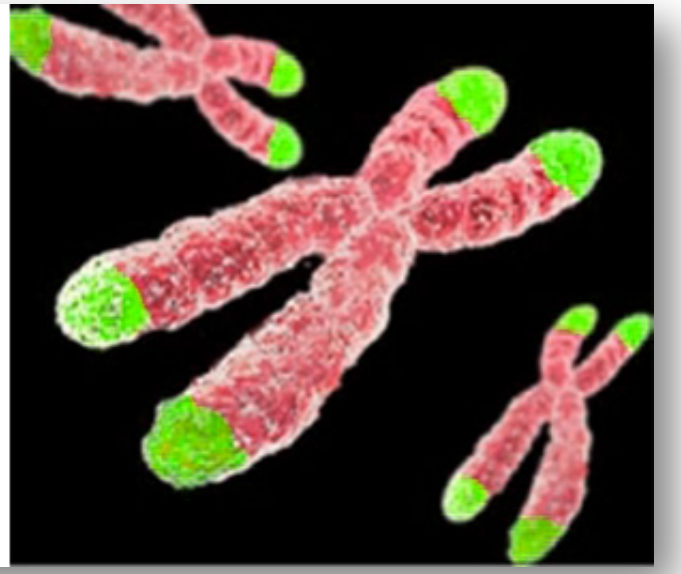
And I really don’t give a rat’s hindquarters about rules

Ribosomes & Telomeres

Two mind-boggling biological nanosystems

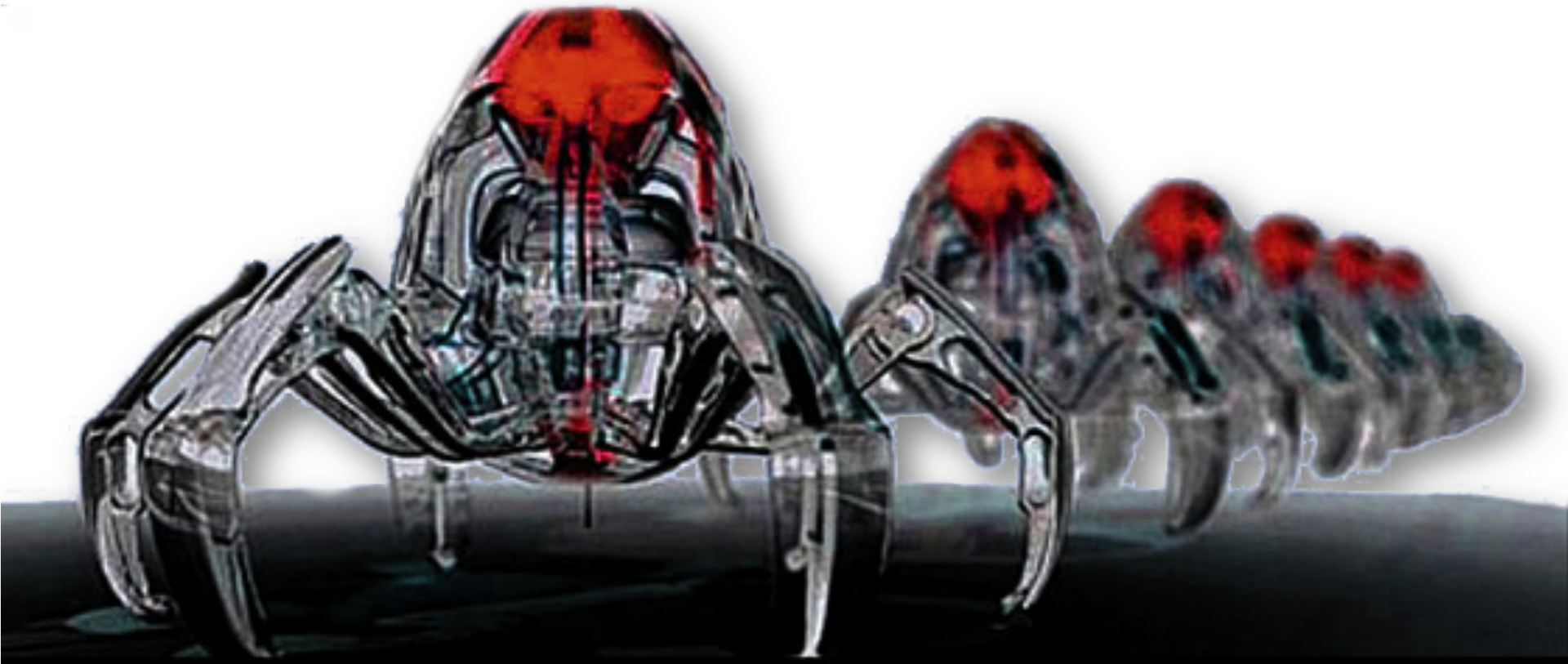


2 billion billion/human



3 million billion/human

DNA Nanobots



Currently 0/human, but not for long…

DNA Does Not Care About You

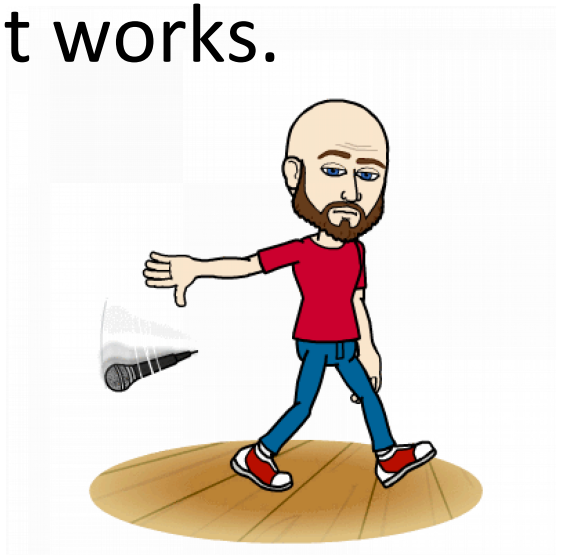
(or anything)

DNA is a non-thinking, non-caring piece of chemistry that propagates through time using disposable organic vessels (e.g., us).

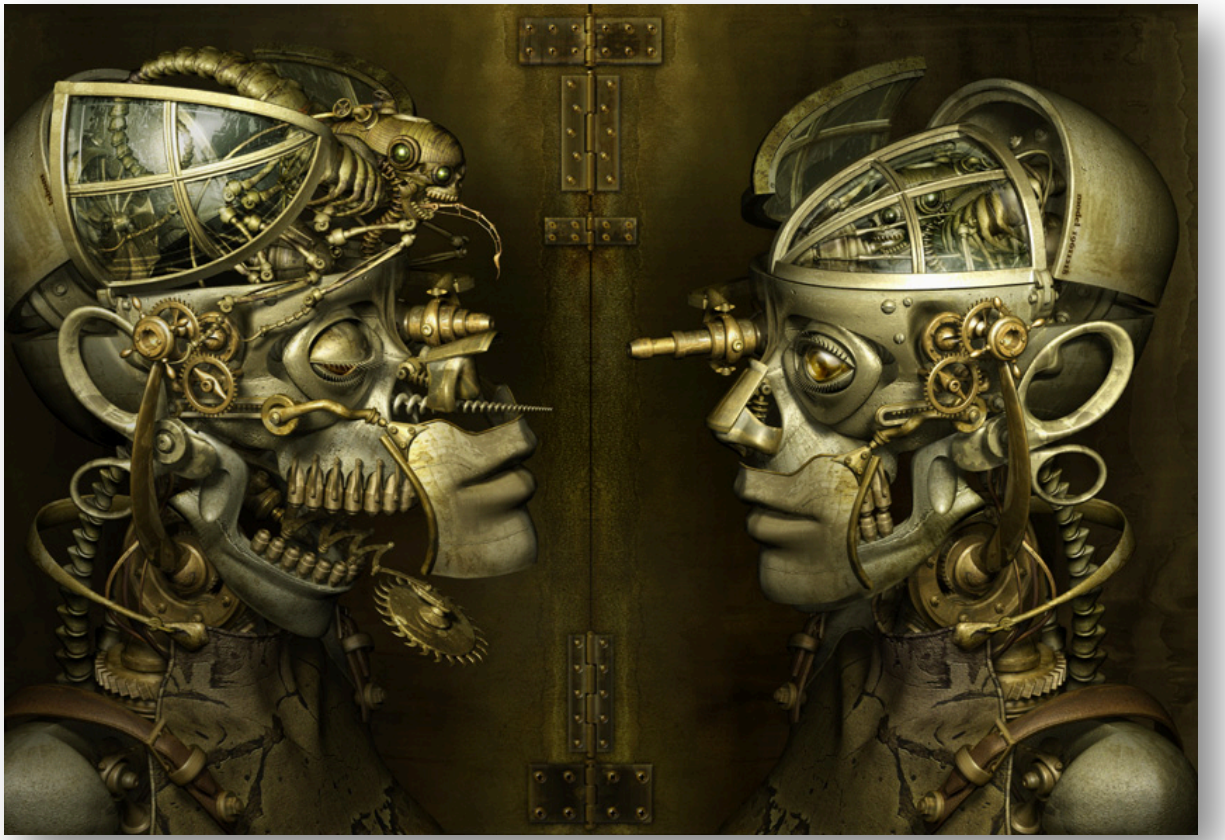
Likewise, evolution is a non-thinking, non-caring process with no agenda.

The significance of life is that it works.

Drop the mic.



Life is Autonomous Computation



One Classic

equipment, and to Dr. G. E. R. Deacon and the captain and officers of R.R.S. *Discovery II* for their part in making the observations.

[1] Young, F. B., Gerrard, H., and Jevons, W., *Phil. Mag.*, **40**, 149 (1920).

[2] Longuet-Higgins, M. S., *Mon. Not. Roy. Astro. Soc., Geophys. Supp.*, **5**, 285 (1949).

[3] Von Arx, W. S., Woods Hole Papers in Phys. Oceanog. Meteor., **11** (3) (1956).

[4] Ekman, V. W., *Arkiv. Mat. Astron. Fysik.* (Stockholm), **2** (11) (1905).

MOLECULAR STRUCTURE OF NUCLEIC ACIDS

A Structure for Deoxyribose Nucleic Acid

WE wish to suggest a structure for the salt of deoxyribose nucleic acid (D.N.A.). This structure has novel features which are of considerable biological interest.

A structure for nucleic acid has already been proposed by Pauling and Corey[1]. They kindly made their manuscript available to us in advance of publication. Their model consists of three intertwined chains, with the phosphates near the fibre axis, and the bases on the outside. In our opinion, this structure is unsatisfactory for two reasons: (1) We believe that the material which gives the X-ray diagrams is the salt, not the free acid. Without the acidic hydrogen atoms it is not clear what forces would hold the structure together, especially as the negatively charged phosphates near the axis will repel each other. (2) Some of the van der Waals distances appear to be too small.

Another three-chain structure has also been suggested by Fraser (in the press). In his model the phosphates are on the outside and the bases on the inside, linked together by hydrogen bonds. This structure as described is rather ill-defined, and for this reason we shall not comment on it.

This figure is purely diagrammatic. The two ribbons symbolize the two phosphate—sugar chains, and the horizontal rods the pairs of bases holding the chains together. The vertical line marks the fibre axis

We wish to put forward a radically different structure for the salt of deoxyribose nucleic acid. This structure has two helical chains each coiled round the same axis (see diagram). We have made the usual chemical assumptions, namely, that each chain consists of phosphate diester groups joining β-D-deoxyribofuranose residues with 3′,5′ linkages. The two chains (but not their bases) are related by a dyad perpendicular to the fibre axis. Both chains follow right-handed helices, but owing to the dyad the sequences of the atoms in the two chains run in opposite directions. Each chain loosely resembles Furberg's[2] model No. 1; that is, the bases are on the inside of the helix and the phosphates on the outside. The configuration of the sugar and the atoms near it is close to Furberg's 'standard configuration', the sugar being roughly perpendicular to the attached base. There is a residue on each chain every 3·4 A. in the *z*-direction. We have assumed an angle of 36° between adjacent residues in the same chain, so that the structure repeats after 10 residues on each chain, that is, after 34 A. The distance of a phosphorus atom from the fibre axis is 10 A. As the phosphates are on the outside, cations have easy access to them.

The structure is an open one, and its water content is rather high. At lower water contents we would expect the bases to tilt so that the structure could become more compact.

The novel feature of the structure is the manner in which the two chains are held together by the purine and pyrimidine bases. The planes of the bases are perpendicular to the fibre axis. They are joined together in pairs, a single base from one chain being hydrogen-bonded to a single base from the other chain, so that the two lie side by side with identical *z*-co-ordinates. One of the pair must be a purine and the other a pyrimidine for bonding to occur. The hydrogen bonds are made as follows: purine position 1 to pyrimidine position 1; purine position 6 to pyrimidine position 6.

If it is assumed that the bases only occur in the structure in the most plausible tautomeric forms (that is, with the keto rather than the enol configurations) it is found that only specific pairs of bases can bond together. These pairs are: adenine (purine) with thymine (pyrimidine), and guanine (purine) with cytosine (pyrimidine).

In other words, if an adenine forms one member of a pair, on either chain, then on these assumptions the other member must be thymine; similarly for guanine and cytosine. The sequence of bases on a single chain does not appear to be restricted in any way. However, if only specific pairs of bases can be formed, it follows that if the sequence of bases on one chain is given, then the sequence on the other chain is automatically determined.

It has been found experimentally[3,4] that the ratio of the amounts of adenine to thymine, and the ratio of guanine to cytosine, are always very close to unity for deoxyribose nucleic acid.

It is probably impossible to build this structure with a ribose sugar in place of the deoxyribose, as the extra oxygen atom would make too close a van der Waals contact.

The previously published X-ray data[2,3] on deoxyribose nucleic acid are insufficient for a rigorous test of our structure. So far as we can tell, it is roughly compatible with the experimental data, but it must be regarded as unproved until it has been checked against more exact results. Some of these are given in the following communications. We were not aware of the details of the results presented there when we devised our structure, which rests mainly though not entirely on published experimental data and stereochemical arguments.

It has not escaped our notice that the specific pairing we have postulated immediately suggests a possible copying mechanism for the genetic material.

Full details of the structure, including the conditions assumed in building it, together with a set of co-ordinates for the atoms, will be published elsewhere.

We are much indebted to Dr. Jerry Donohue for constant advice and criticism, especially on interatomic distances. We have also been stimulated by a knowledge of the general nature of the unpublished experimental results and ideas of Dr. M. H. F. Wilkins, Dr. R. E. Franklin and their co-workers at

1953

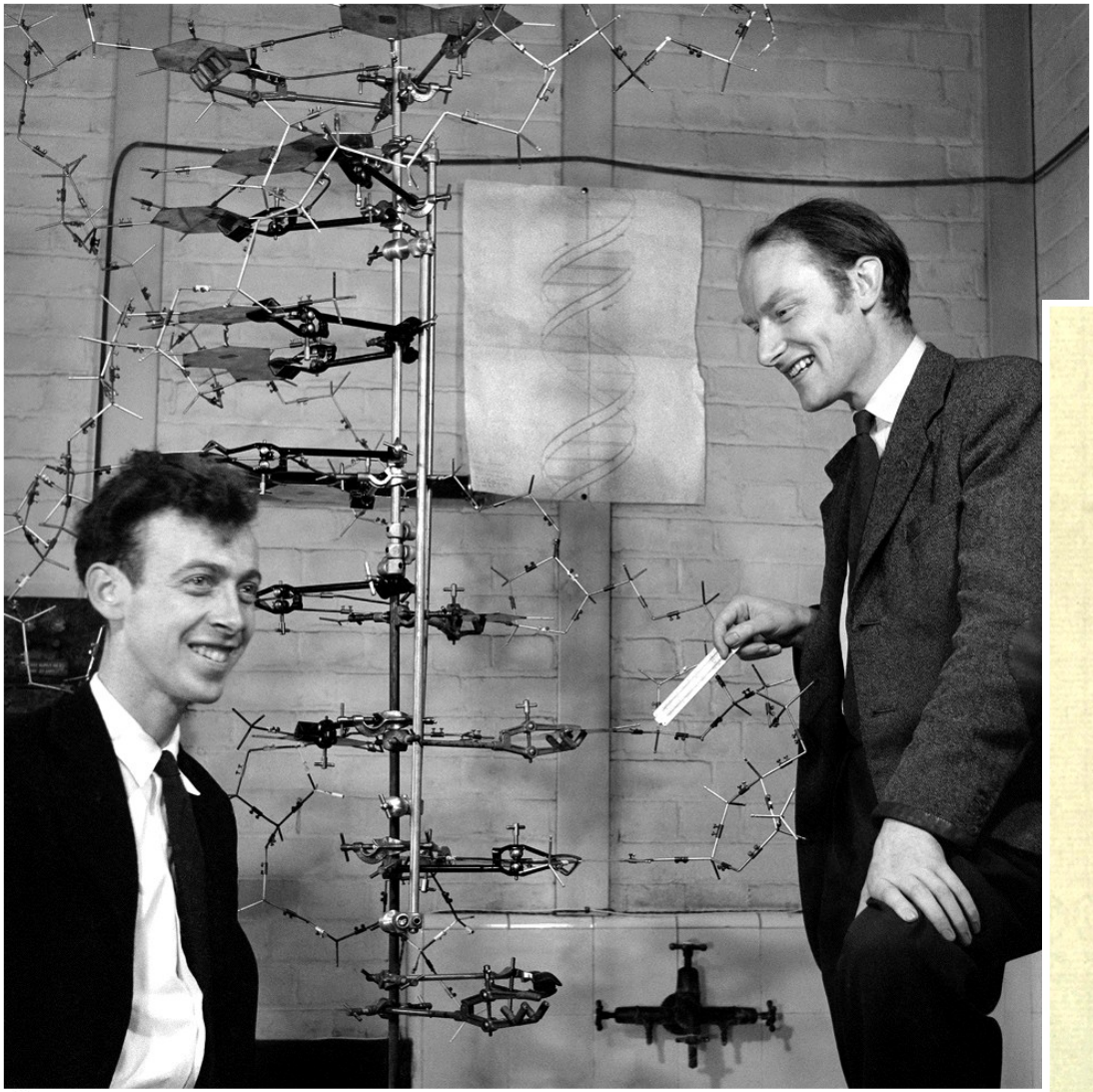


Figure 5-3a
What Is Life? A Guide To Biology

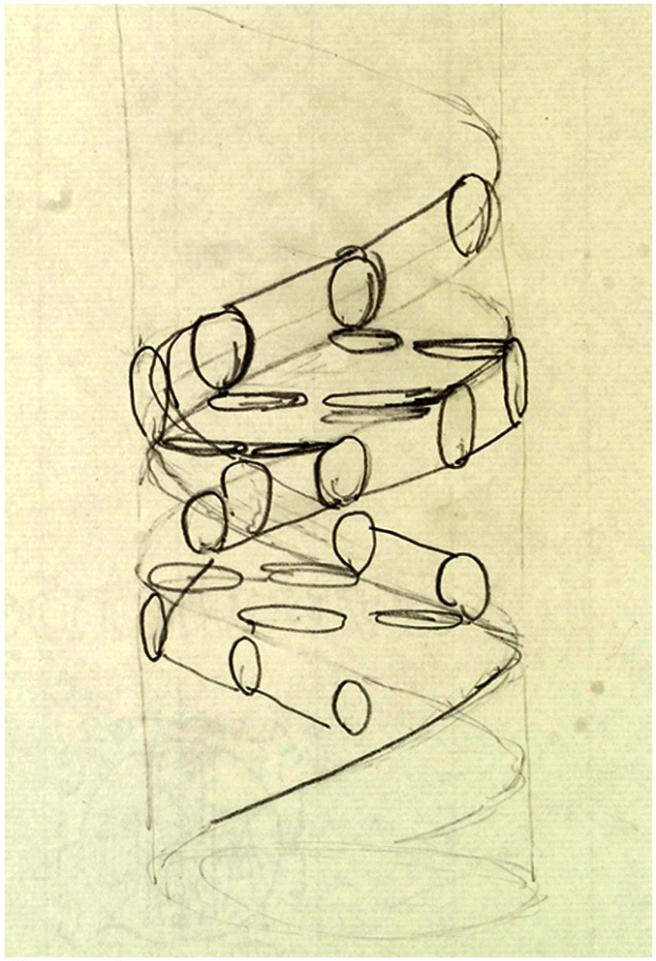
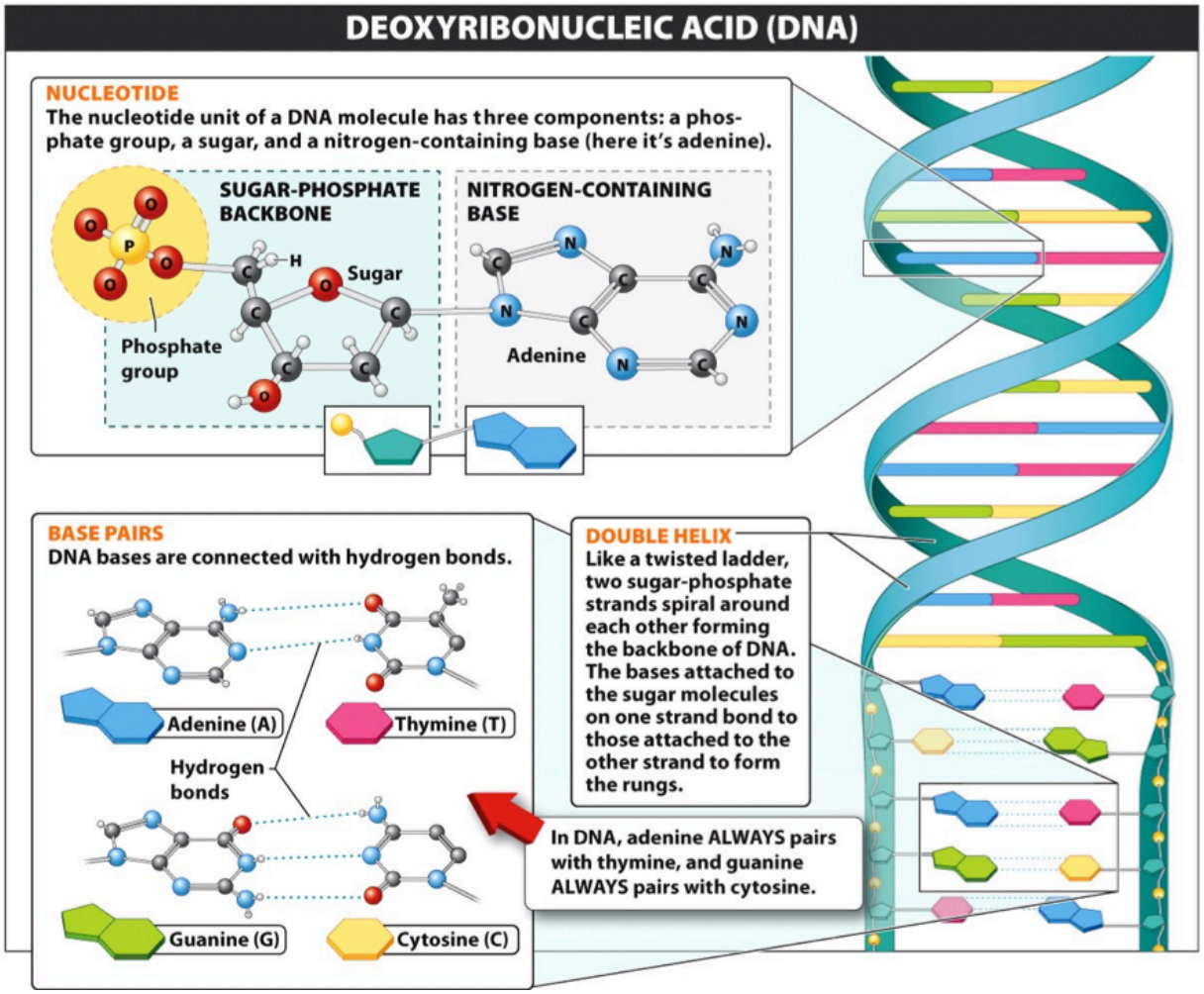


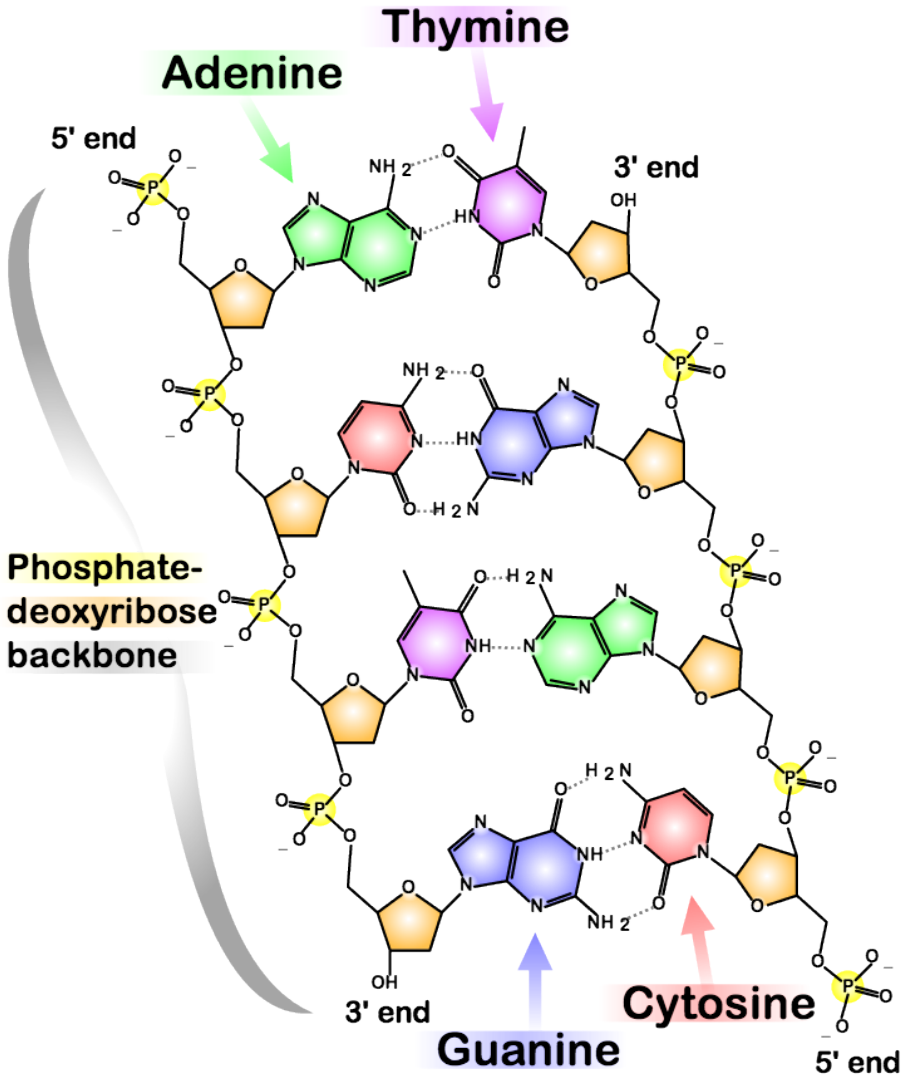
Figure 5-3b
What Is Life? A Guide To Biology

DNA "Double Helix"

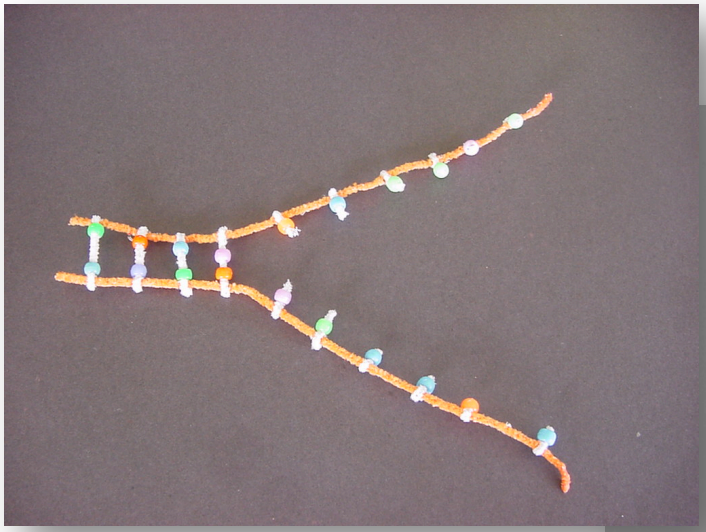


Nucleic acids and nucleotides

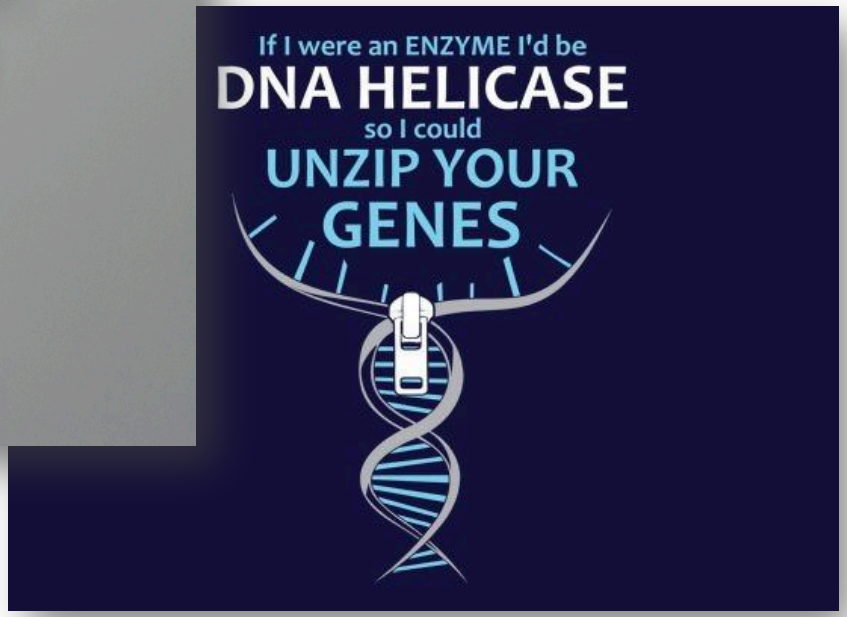
Sugars, Phosphates, and Bases



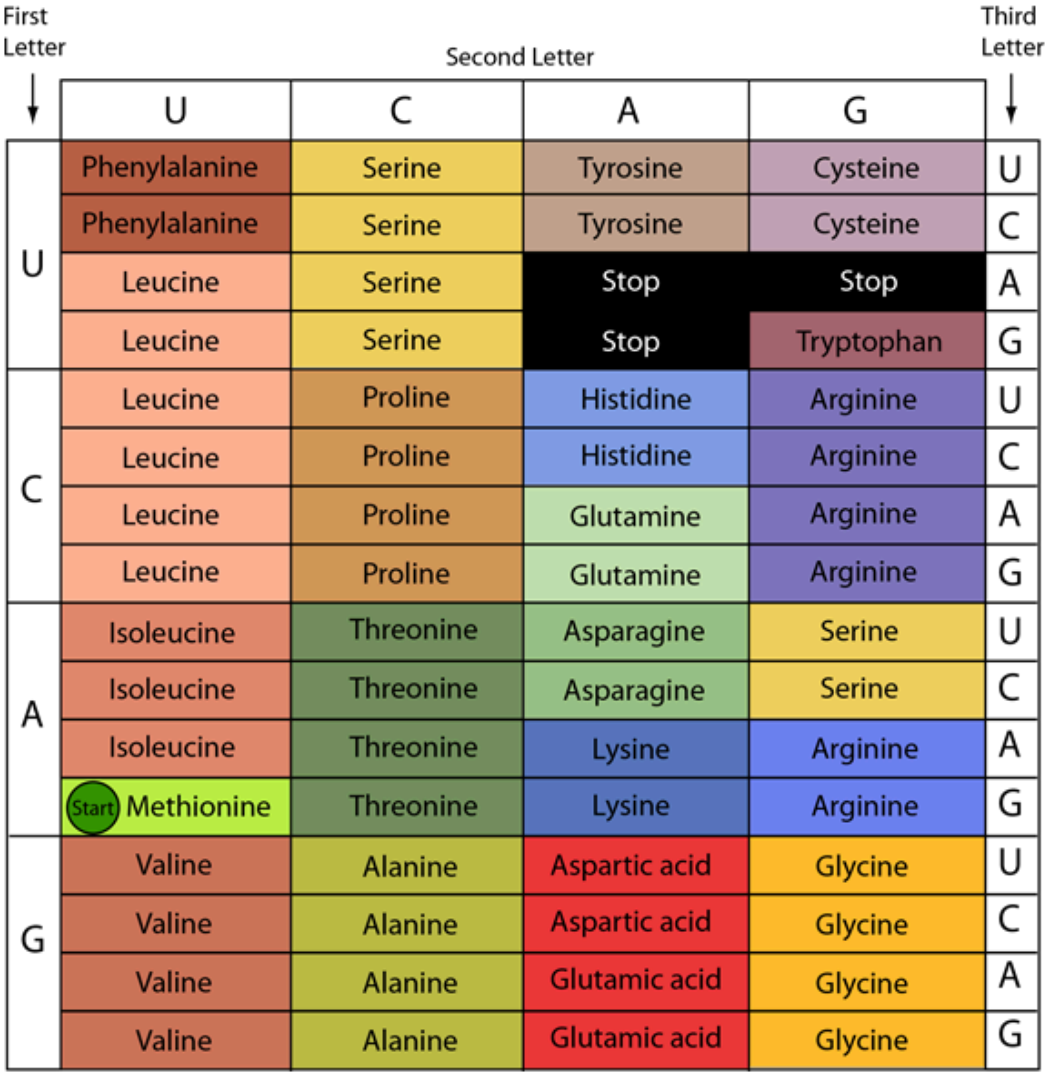
The Magic of DNA



Zip-Zip



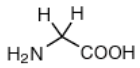
The Genetic Code



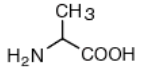
First Letter	Second Letter: U	Second Letter: C	Second Letter: A	Second Letter: G	Third Letter
U	Phenylalanine	Serine	Tyrosine	Cysteine	U
	Phenylalanine	Serine	Tyrosine	Cysteine	C
	Leucine	Serine	Stop	Stop	A
	Leucine	Serine	Stop	Tryptophan	G
C	Leucine	Proline	Histidine	Arginine	U
	Leucine	Proline	Histidine	Arginine	C
	Leucine	Proline	Glutamine	Arginine	A
	Leucine	Proline	Glutamine	Arginine	G
A	Isoleucine	Threonine	Asparagine	Serine	U
	Isoleucine	Threonine	Asparagine	Serine	C
	Isoleucine	Threonine	Lysine	Arginine	A
	Start Methionine	Threonine	Lysine	Arginine	G
G	Valine	Alanine	Aspartic acid	Glycine	U
	Valine	Alanine	Aspartic acid	Glycine	C
	Valine	Alanine	Glutamic acid	Glycine	A
	Valine	Alanine	Glutamic acid	Glycine	G

Amino Acids

Small

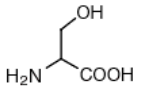


Glycine (Gly, G)
MW: 57.05

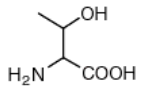


Alanine (Ala, A)
MW: 71.09

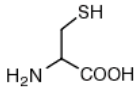
Nucleophilic



Serine (Ser, S)
MW: 87.08, pK_a ~ 16

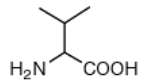


Threonine (Thr, T)
MW: 101.11, pK_a ~ 16

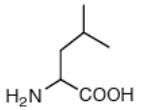


Cysteine (Cys, C)
MW: 103.15, pK_a = 8.35

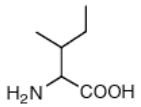
Hydrophobic



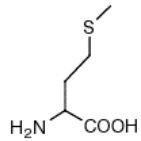
Valine (Val, V)
MW: 99.14



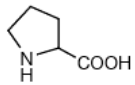
Leucine (Leu, L)
MW: 113.16



Isoleucine (Ile, I)
MW: 113.16

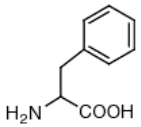


Methionine (Met, M)
MW: 131.19

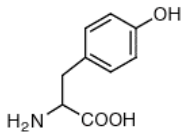


Proline (Pro, P)
MW: 97.12

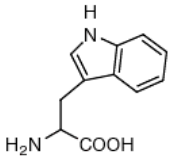
Aromatic



Phenylalanine (Phe, F)
MW: 147.18

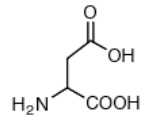


Tyrosine (Tyr, Y)
MW: 163.18

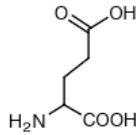


Tryptophan (Trp, W)
MW: 186.21

Acidic

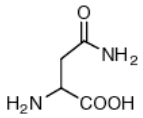


Aspartic Acid (Asp, D)
MW: 115.09, pK_a = 3.9

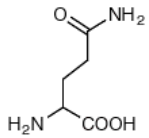


Glutamic Acid (Glu, E)
MW: 129.12, pK_a = 4.07

Amide

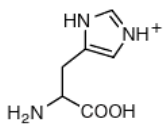


Asparagine (Asn, N)
MW: 114.11

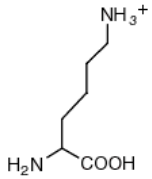


Glutamine (Gln, Q)
MW: 128.14

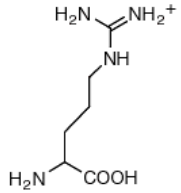
Basic



Histidine (His, H)
MW: 137.14, pK_a = 6.04

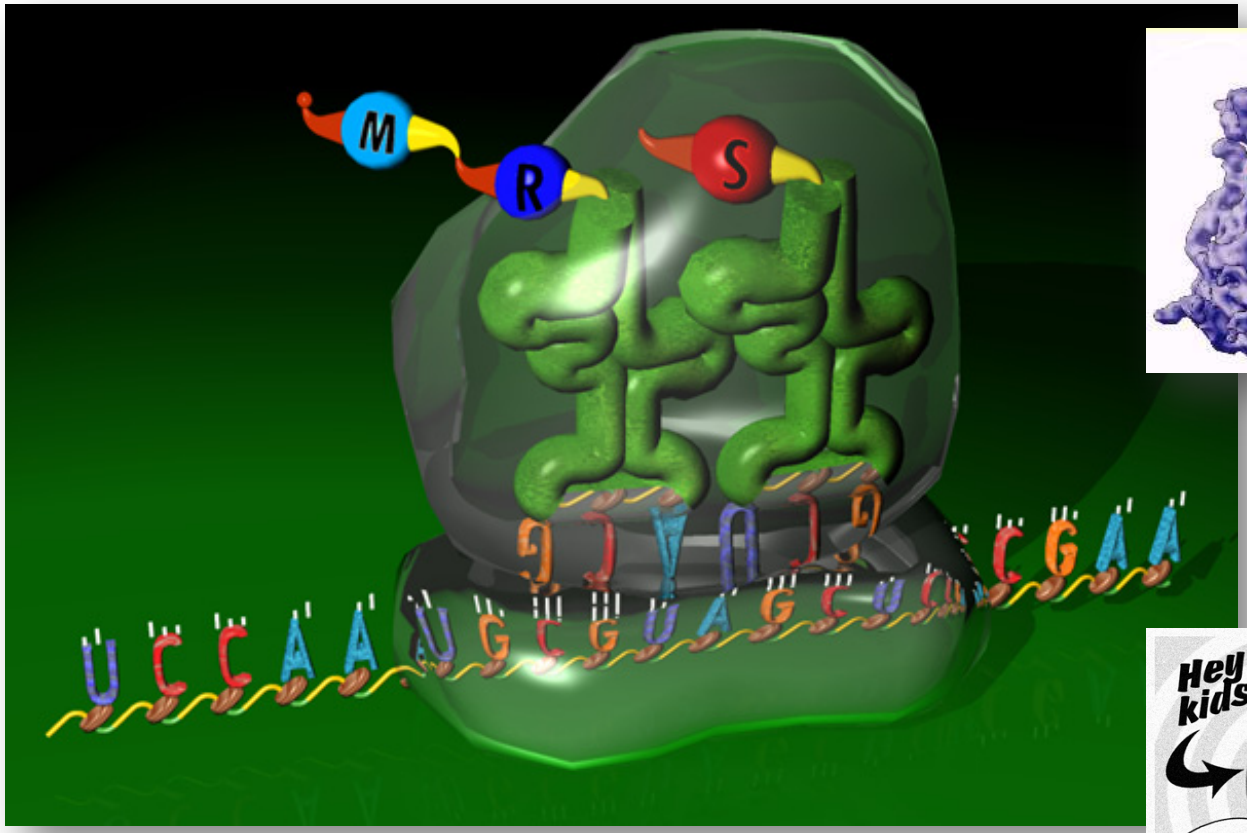


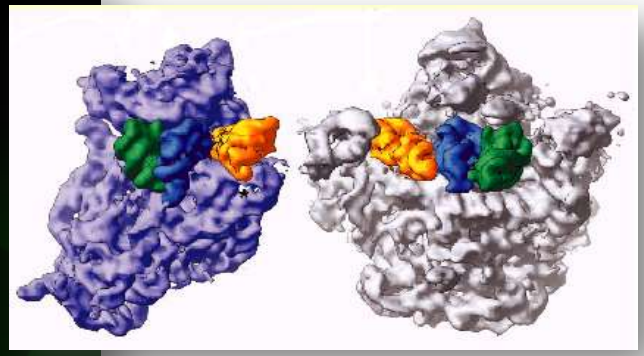
Lysine (Lys, K)
MW: 128.17, pK_a = 10.79



Arginine (Arg, R)
MW: 156.19, pK_a = 12.48

Self Assembling Nano Machine That Makes You!
(piece by tiny piece)

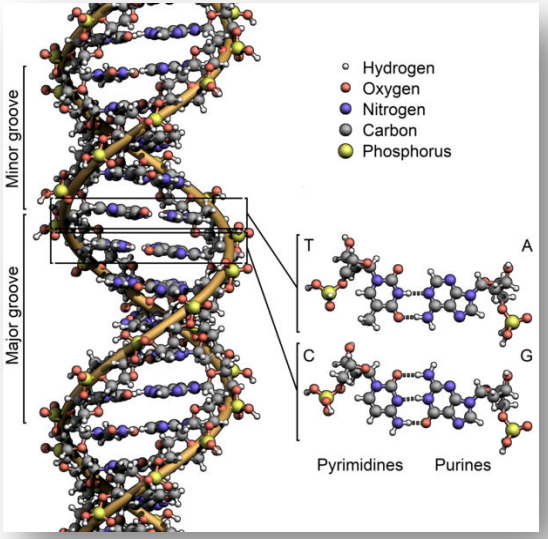




(Life's "decoder ring")

DNA Has a Multidimensional Code(s)

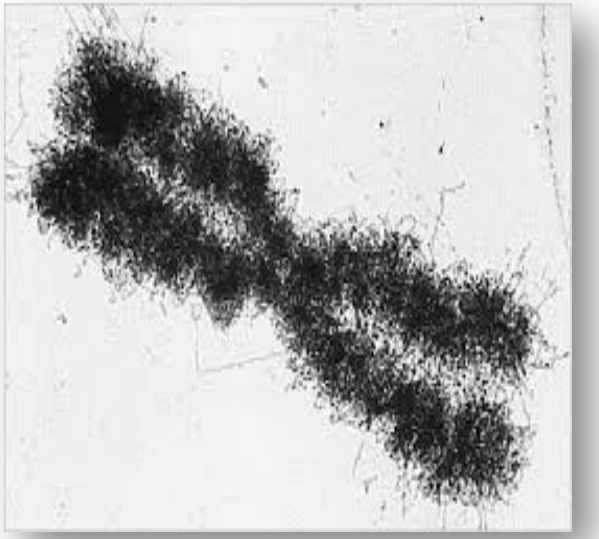
Chemical



Genetic

	U	C	A	G	
U	Phe	Ser	Tyr	Cys	U
	Phe	Ser	Tyr	Cys	C
	Leu	Ser	STOP	STOP	A
	Leu	Ser	STOP	Trp	G
C	Leu	Pro	His	Arg	U
	Leu	Pro	His	Arg	C
	Leu	Pro	Gln	Arg	A
	Leu	Pro	Gln	Arg	G
A	Ile	Thr	Asn	Ser	U
	Ile	Thr	Asn	Ser	C
	Ile	Thr	Lys	Arg	A
	Met	Thr	Lys	Arg	G
G	Val	Ala	Asp	Gly	U
	Val	Ala	Asp	Gly	C
	Val	Ala	Glu	Gly	A
	Val	Ala	Glu	Gly	G

Hint…?



1/23rd meter
~4.34 cm
Into 1-2 µm

Another Classic

Design of DNA origami

Paul W.K. Rothemund
Computer Science and Computation and Neural Systems
California Institute of Technology, Pasadena, CA 91125
pwkr@dna.caltech.edu

***Abstract*— The generation of arbitrary patterns and shapes at very small scales is at the heart of our effort to miniaturize circuits and is fundamental to the development of nanotechnology. Here I review a recently developed method for folding long single strands of DNA into arbitrary two-dimensional shapes using a raster fill technique – 'scaffolded DNA origami'. Shapes up to 100 nanometers in diameter can be approximated with a resolution of 6 nanometers and decorated with patterns of roughly 200 binary pixels at the same resolution. Experimentally verified by the creation of a dozen shapes and patterns, the method is easy, high yield, and lends itself well to automated design and manufacture. So far, CAD tools for scaffolded DNA origami are simple, require hand-design of the folding path, and are restricted to two dimensional designs. If the method gains wide acceptance, better CAD tools will be required.**

I. Introduction

Top-down methods for patterning at the nanoscale have been very successful. Methods range from photolithography, which allows routine patterning at the 90-nanometer scale, to more exotic methods like electron beam lithography, dip-pen lithography [1], atomic force microscopy (AFM) [2] and scanning tunnelling microscopy (STM) [3], [4] that allow patterning at length scales from 20 nm down to 0.1 nm. Top-down methods, however, have several drawbacks. To reach finer length scales, it appears that photolithography will require fabrication equipment of steeply increasing cost. The remaining techniques are serial; they require that patterns be created by drawing one line or one pixel at a time. Except for dip-pen lithography and AFM, top-down methods require ultra-high vacuum, ultra-clean conditions, or cryogenic temperatures.

Self-assembly, the spontaneous organization of matter by attractive forces, has been put forth as an inexpensive, parallel method for the synthesis of nanostructures that does not require expensive equipment and extreme conditions [5]. At the molecular scale many different classes of molecules have been advanced as the basic units of self-assembly, from relatively small organic molecules like porphyrins [6] or short peptides [7] to proteins [8] or whole viral particles [9]. Much progress has been made in these systems but the resulting structures are relatively simple and generally periodic in nature.

The problem is that to create complex structures using self-assembly, one must be able to program complex attractive interactions into the basic units: the interactions between the basic units must be highly specific and the geometry between units, once bonded, must be well-defined. An important difficulty is that of creating many different types of 'specific glue'. I give an example without defining any formal notions of components or what it means for them to stick together. If components of type **A**, **B**, **C** and **D** are to stick together into a linear structure **ABCD** then three specific attractive interactions—glues—must be built into the components, one for each of the pairwise interactions **AB**, **BC** and **CD**. By *specific* I mean that there is no cross-interaction between the specific glues—no pairs **AC** form, for example. For most classes of molecules, creating more than a few types of components and a few types of specific glue is a difficult research project. Creating components with complex geometry, for example squares with four edges, each capable of carrying a specific glue, is beyond our reach for most classes of molecules; for proteins it may take a decade or more before we can engineer such components.

DNA, however, is readily engineered to create complex components for self-assembly. The use of DNA for this purpose is encompassed by the field of 'DNA nanotechnology' [10], [11] which uses the exquisite molecular recognition of Watson-Crick binding to program the self-assembly of complex structures. DNA nanotechnologists rely on the principle that, to first order, a DNA sequence composed of the 'A', 'G', 'C', 'T' binds most strongly to its perfect complement. For example '5-ACCGGGTTTT-3' binds most strongly to '3-TGGCCCAAAA-5', somewhat less strongly to a sequence with a Hamming distance of 1 from the perfect complement '3-TGGCCCAAA**C**-5', even less strongly to a sequence of Hamming distance 2, such as '3-TGGC**A**CAAA**C**-5', etc.[1] The ordering of binding strengths is only approximately governed by Hamming distance and actually depends on the sequences in question [12]; much progress can be made with this approximation, however. Further, while the energy of binding decreases roughly linearly with Hamming distance, the tendency of two strands to bind, as measured by the equilibrium constant, changes exponentially—making it possible to design many different DNA glues of extraordinary specificity.

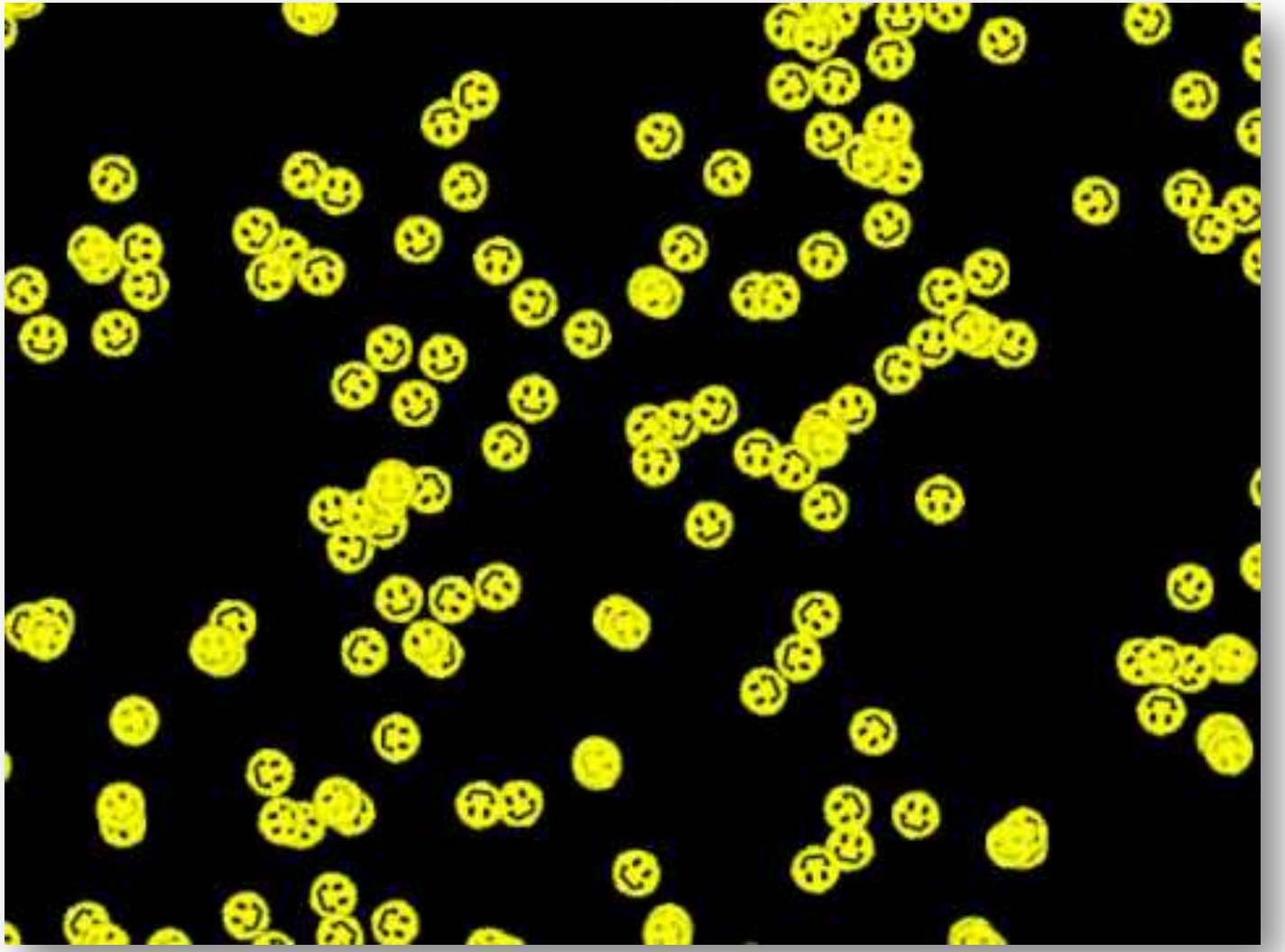
A second major principle, upon which DNA nanotechnologists rely, is that DNA has many rigid, well-characterized forms that are not a linear double helix. Of particular interest are branched forms of DNA, wherein three or more double helices intersect at a common vertex, as in Fig. 1a. This is accomplished by giving each of three different DNA sequences partially complementary sequences. The first half of strand **1** complements the last half of strand **2**, the first half of strand **2** complements the last half of strand **3** and the first half of strand **3** complements the last half of strand **1**. Fig. 1d and e show an important example, a 'double-crossover molecule' the first rigid, engineered DNA structure [13]. In this molecule 5 strands are used to create a structure in which two double helices are held in a rigid parallel arrangement. Note how some strands (2,3 and 4) participate in both helices—they wind along one helix, then switch to another through a structure called a 'crossover' (small black triangles). It is the crossovers that hold the helices together.

Over the last 15 years, such techniques have been used to create a diverse set of arbitrary DNA shapes and patterns (Fig. 2 reproduces some of them). Shapes include a cube [14], a truncated octahedron [15], and an octahedron [16]. The most complex pattern demonstrated to date is a 4x4 array of 16 addressable pixels [17]. All these designs represent milestones in the creation of DNA nanostructures; each took significant effort to design and synthesize (on the order of 1-2 years). A question becomes, how may the lessons learned from

[1]DNA sequences have an orientation denoted here by the addition of a '5' and a '3' label to its ends. Thus a sequence is not equivalent to its reverse. Further, strands in a double helix are anti-parallel and thus the complement of a DNA seqence has its '5' and '3' ends reversed.

2006

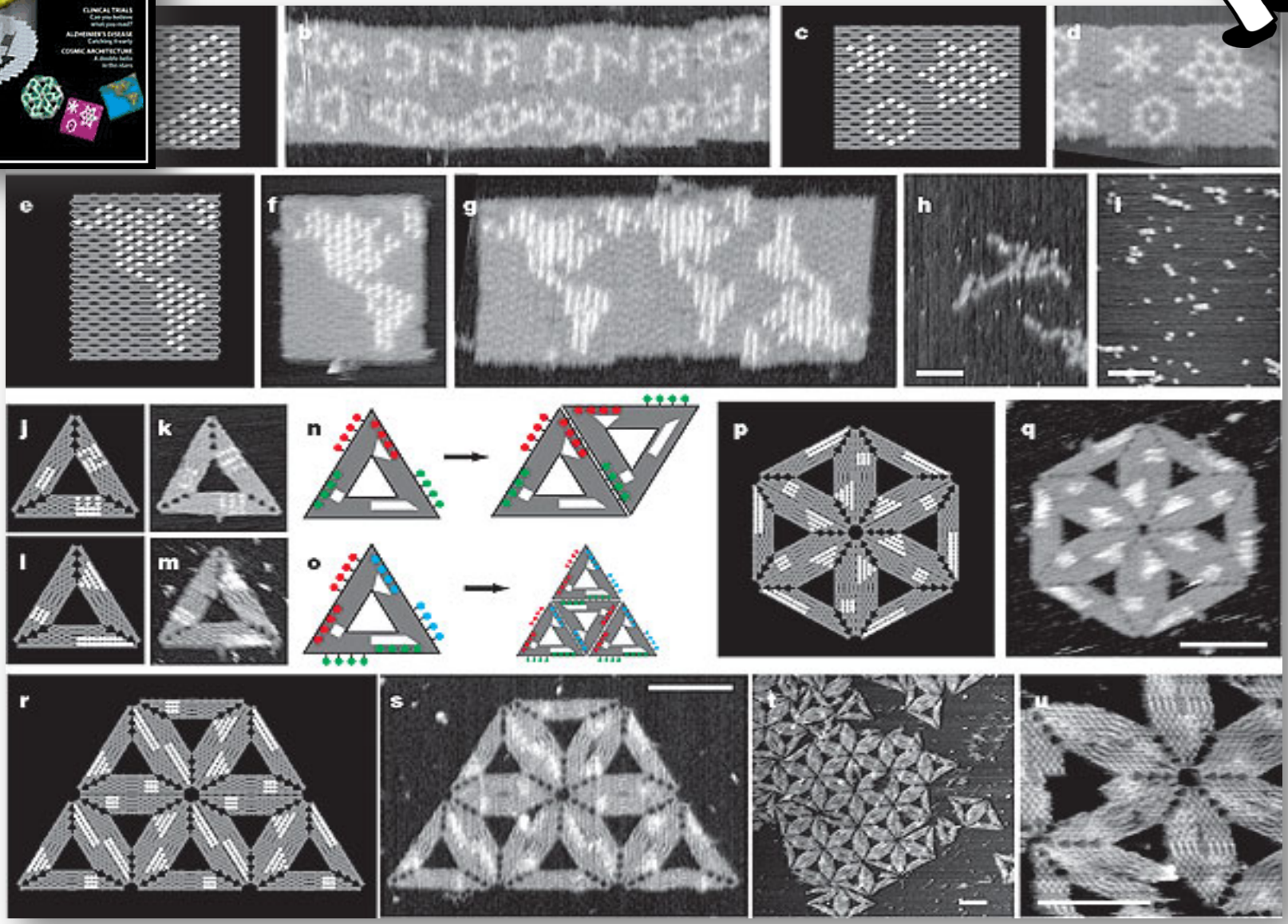
DNA's Engineering Code is No Laughing Matter :)



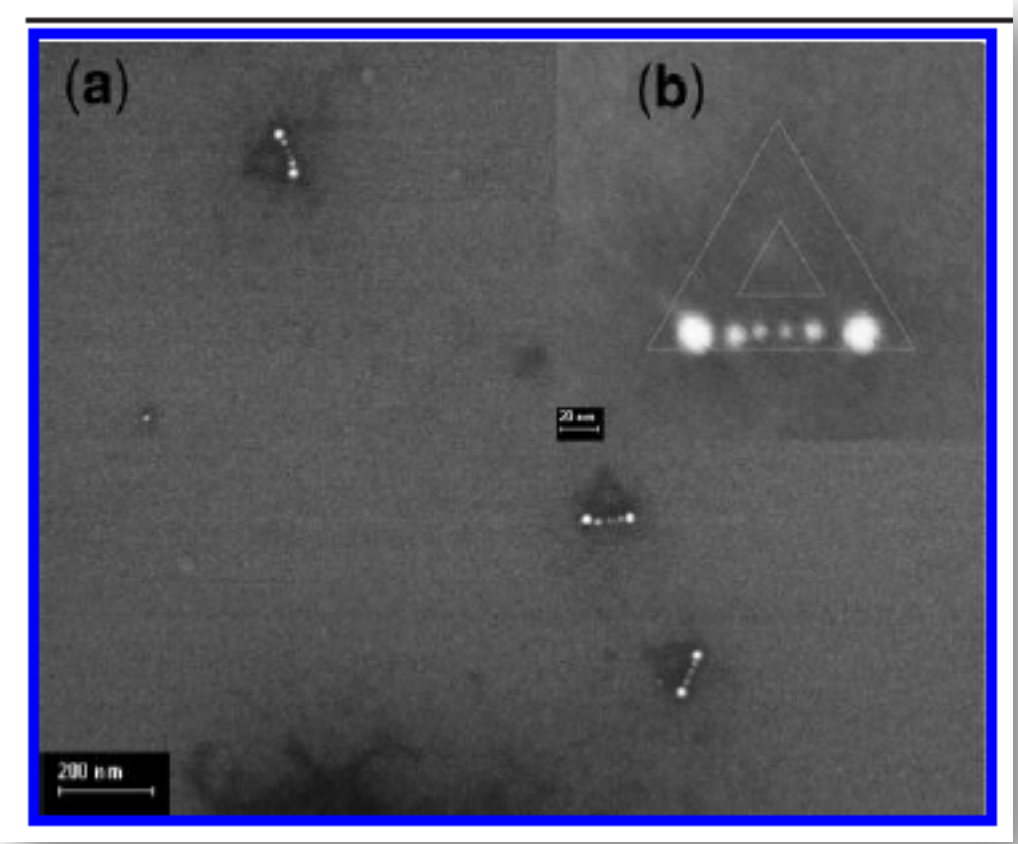
DNA Origami

(breaking "rules", awesome!)

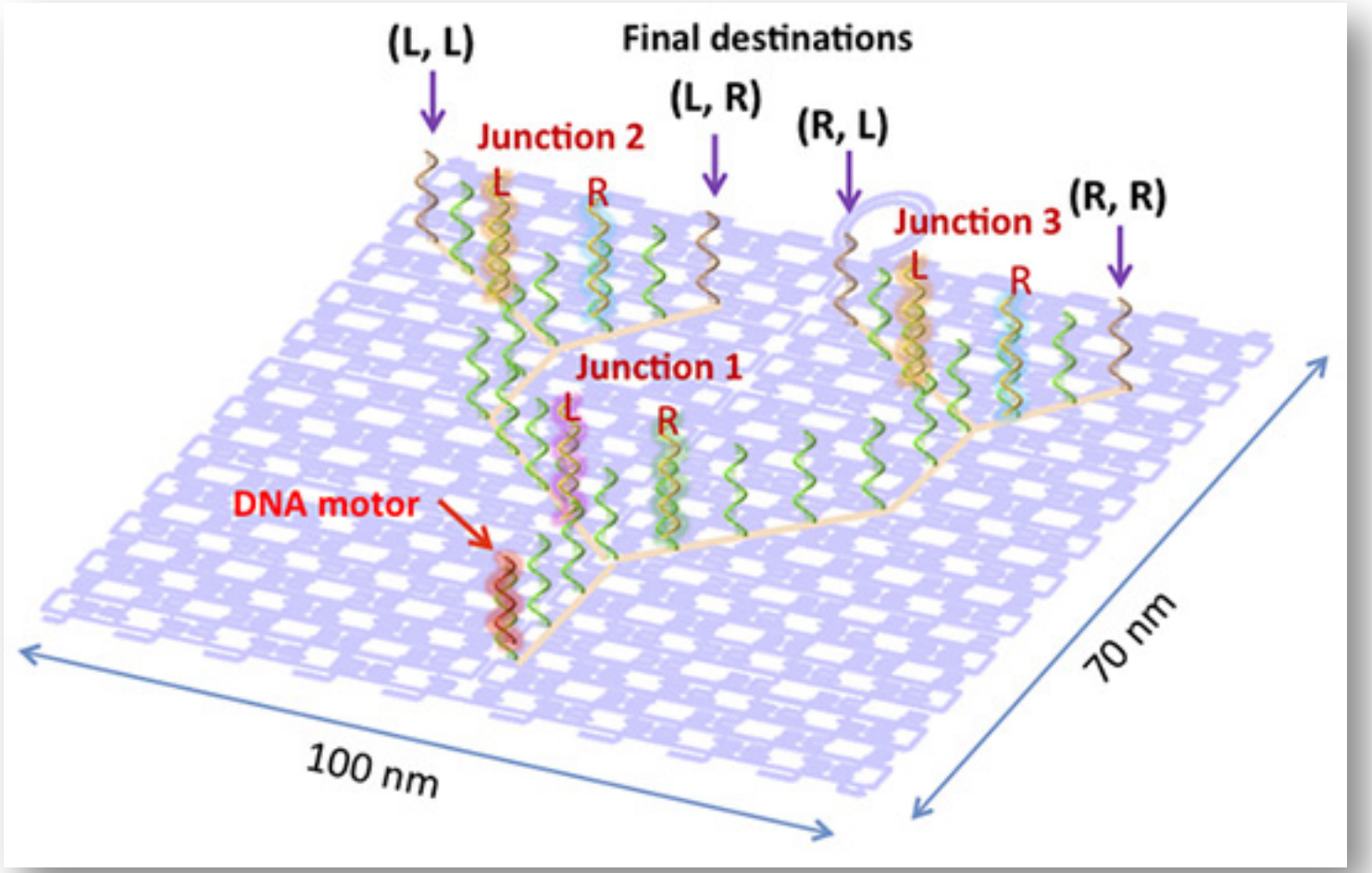
Paul Rothemund



Precise Positioning

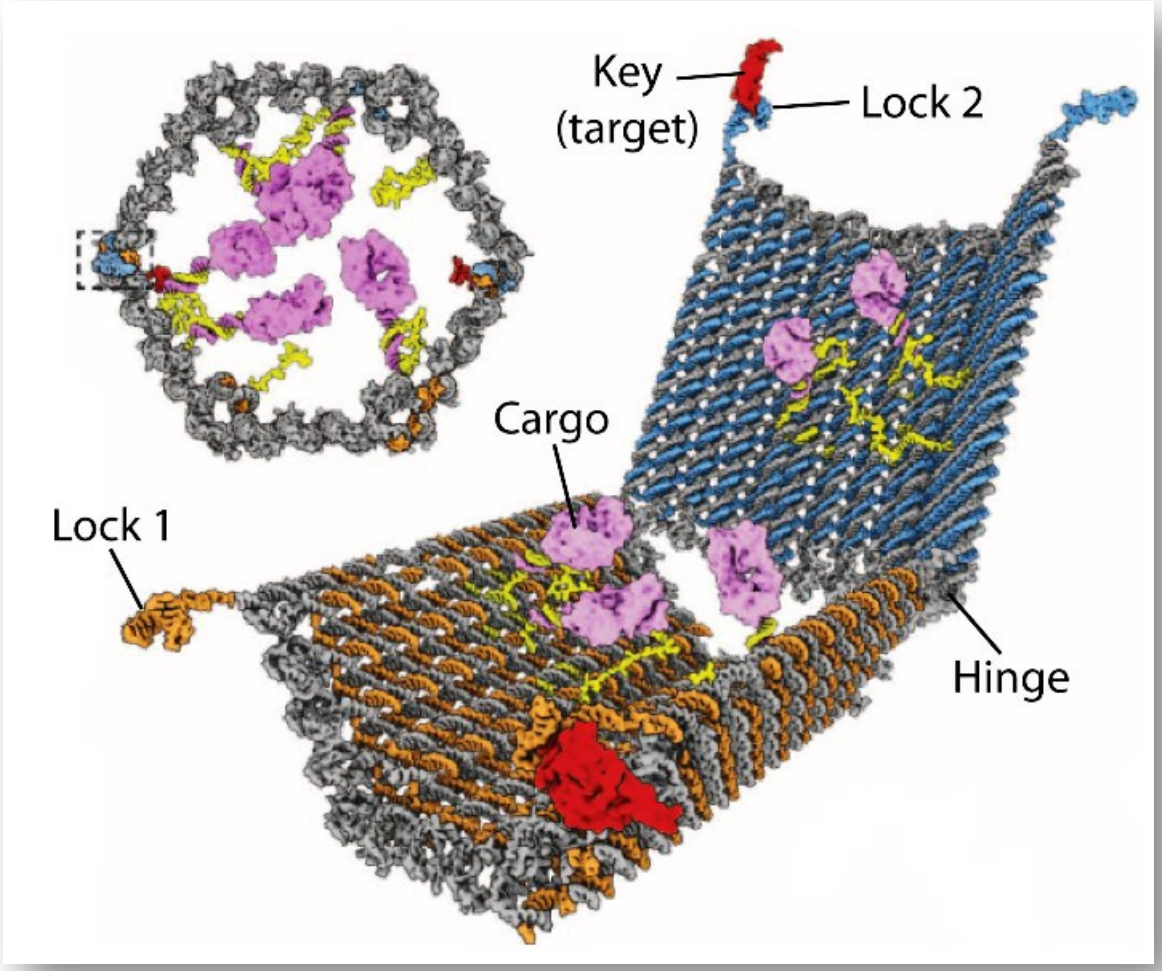


Orchestrating Molecular Events



Nadrian Seeman

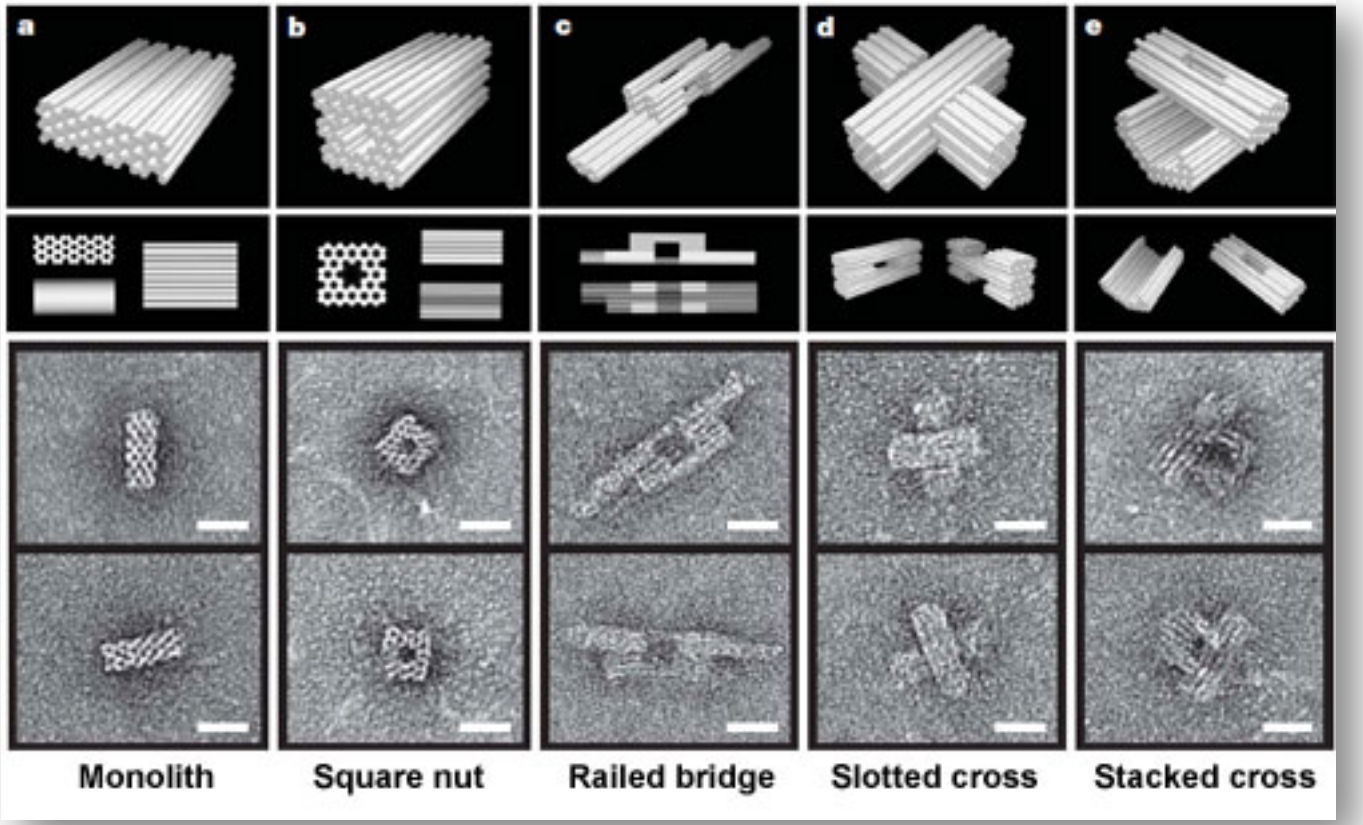
Delivering Drugs



Shawn Douglas

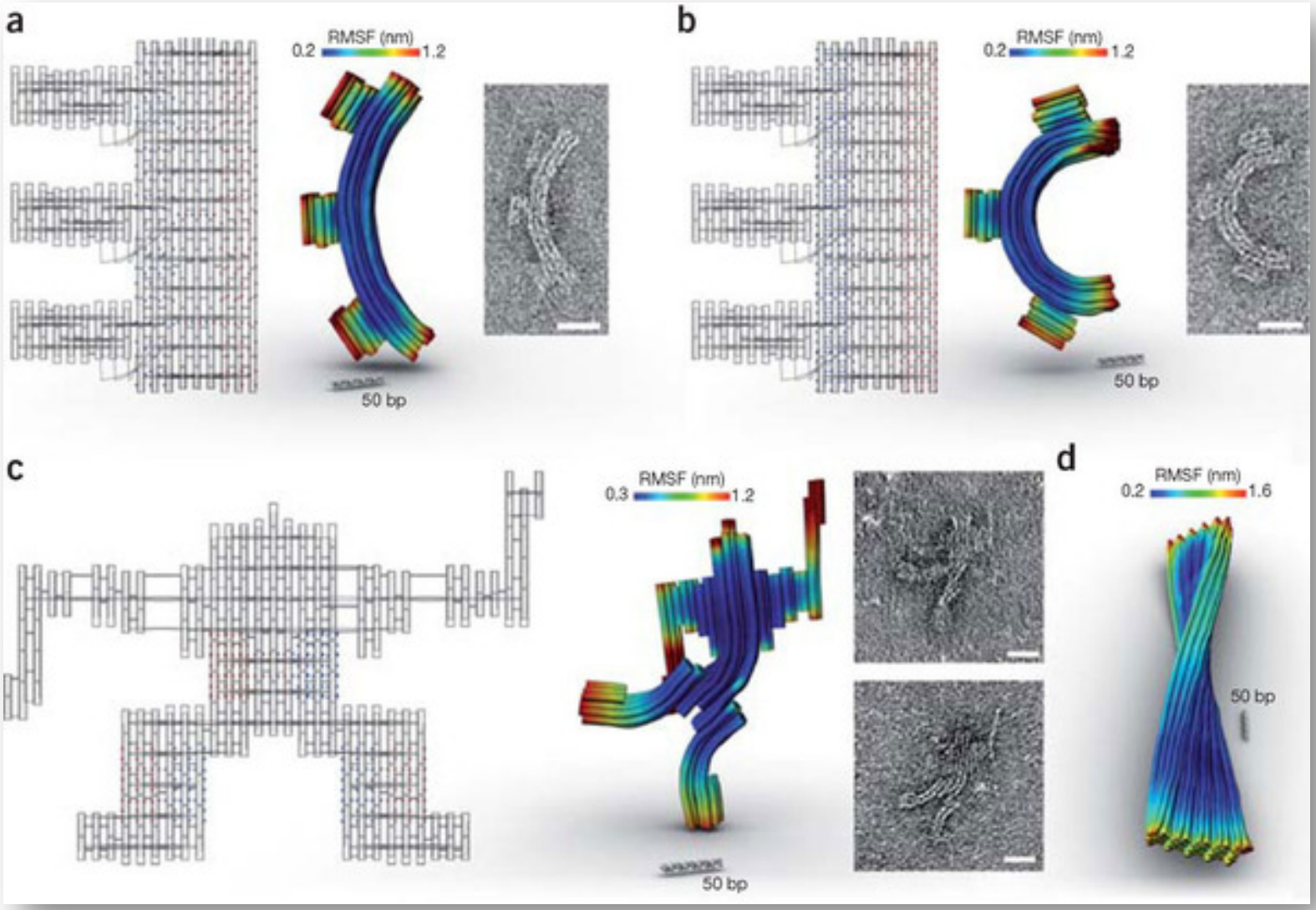
Going 3D: Part 1

(and breaking more "rules")

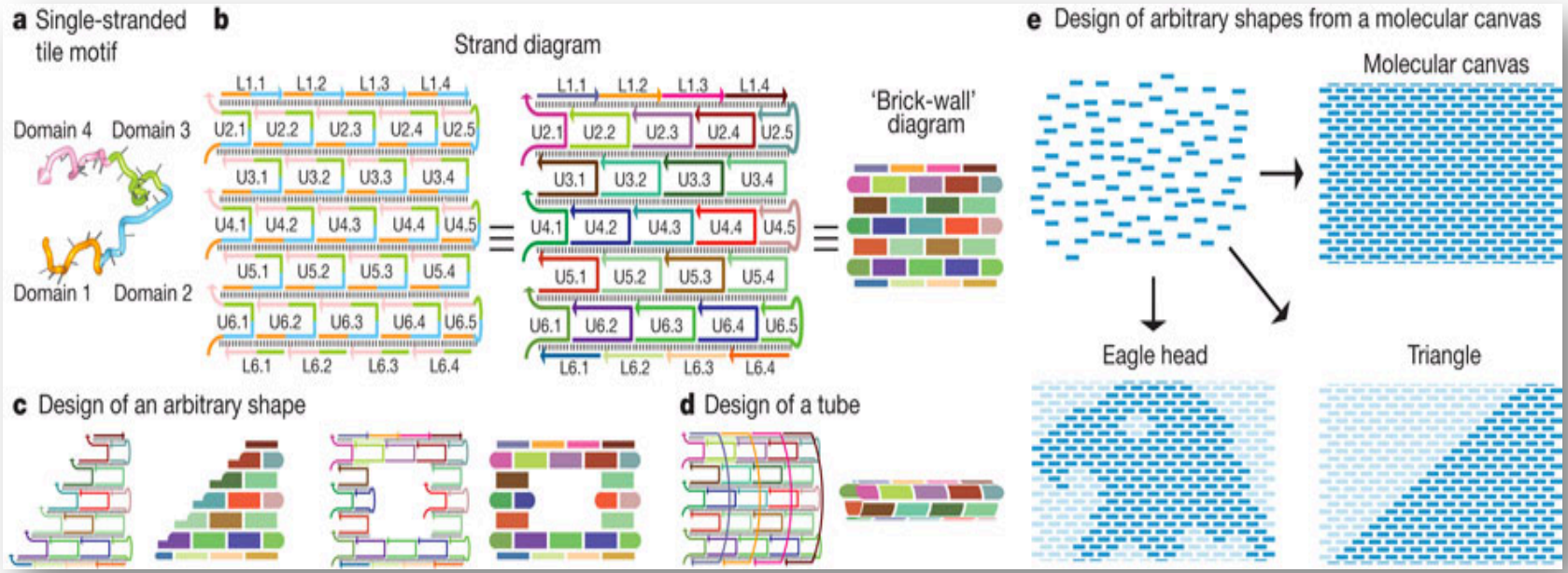


William Shih

Amazing “Gee-Whiz” Stuff



More Gee-Whiz Stuff

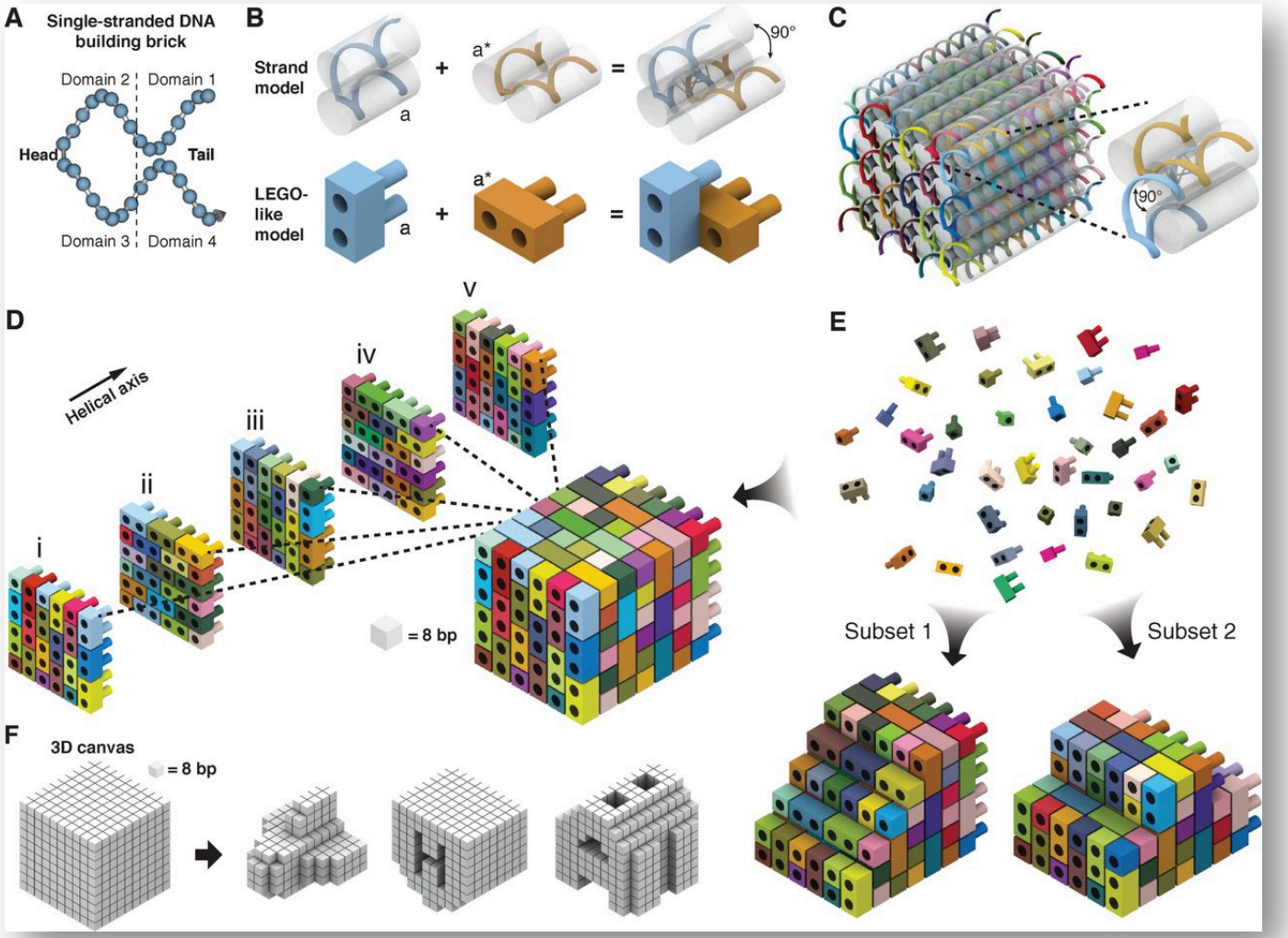


Fully automated process now exists…

Peng Yin

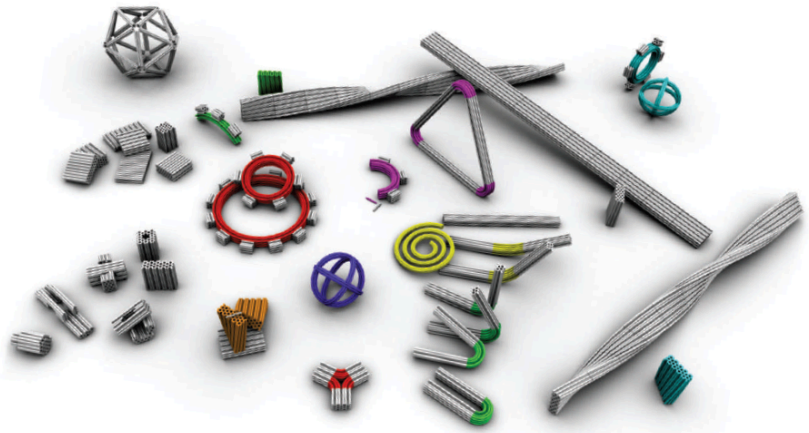
More Gee-Whiz Stuff

Going 3D: Part II



OMG (in 3D)

Finally! Good Question!



What to make with DNA origami

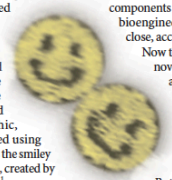
Chemists looking to create complex self-assembling nanostructures are turning to DNA. **Katharine Sanderson** looks at the science beneath the fold.

DNA is the kind of polymer that chemists dream about. Because its complementary sequences can bind to one another, individual molecules of the right sequence will assemble all by themselves into intricate shapes and structures at the nanoscale. DNA can weave together and bind other molecules, allowing it to serve as a scaffold for complex nanomachinery.

DNA nanoengineering is dreamy, but difficult. Researchers have been putting together carefully chosen segments of DNA to form sheets, tubes, even simple machines such as tweezers since the early 1980s. But back then, designing these structures could take months to years. And because researchers were focused on designing them from scratch, they could use only the short segments, no more than 150-base-pairs long, that DNA synthesizers could manufacture. This in turn constrained the size and complexity of the designs. "The problem is that we don't just want to make small stuff, we want to make complicated small stuff, cheaply and easily," says Paul Rothemund, a computational bioengineer at the California Institute of Technology in Pasadena.

Rothemund wondered whether he could create the complicated stuff using a longer, naturally occurring piece of DNA, such as the genome of a virus, and folding it over on itself. So in 2004 and 2005 he spent months, he says, programming in his underpants, trying to work out a way to bend a 7,000-base-pair viral genome to his will. In his design he visualized how the genome could be folded into a predetermined, two-dimensional shape. Knowing the sequence of the virus at every twist and turn, he was able to write complementary DNA sequences, about 16-base-pairs long, that would essentially staple the folds in place. He ordered the 'staples' from a DNA-synthesis company, mixed them with his virus in a buffer that stabilized the DNA and then heated and cooled the mixture, allowing the single stranded viral DNA to bind with the staples (see graphic, opposite). The result, viewed using atomic-force microscopy, was the smiley face and several other shapes, created by what he called DNA origami[1].

P. W. K. ROTHEMUND



The ease of DNA origami was a breakthrough, dispensing with the intricacies of precise DNA engineering and other metamaterials development. "It's like being able to bake a cake and not pay attention to the ingredient ratios," says Rothemund. But with the right ingredients complex structures can be built with the kind of precision that many people have been looking for. Origami scaffolds, sheets or bricks of folded DNA, are packed with known sequences that could be used to position DNA-binding molecules just a few nanometres apart. And the new, larger structures can contain upwards of 200 sites for affixing such molecules, compared with only a handful on pre-origami structures. This type of precision engineering could be a boon to nanoengineers wanting to position components on nanoelectronic circuits or for bioengineers looking to place proteins in close, accurate proximity to one another.

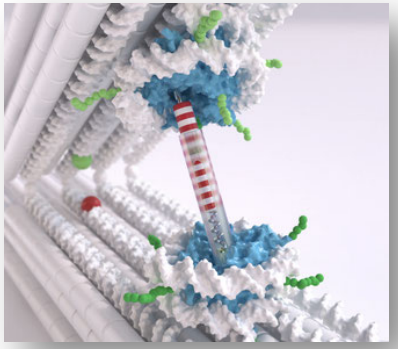
H. DIETZ/TECHNICAL UNIV. MUNICH

Now the challenge is to go beyond the novelty of Rothemund's smileys and a dozen or so other demonstration patterns and build structures with a practical purpose. Here's what several researchers are dreaming of doing.

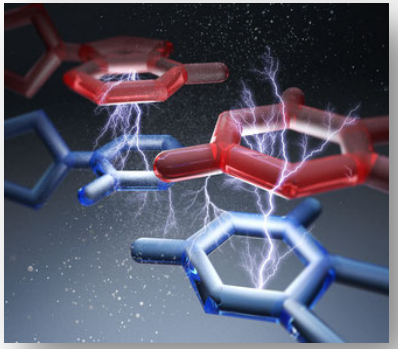
Make a ruler

Rothemund's technique was a door opener for Friedrich Simmel, a biophysicist at the Technical University of Munich in Germany. Suddenly, Simmel says, he was able to have even "rather sloppy" physics students making DNA structures with ease. Simmel has used DNA origami to make a ruler to measure distances between single molecules and

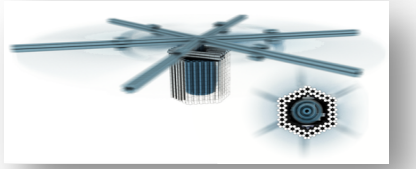
The Dietz Lab at Technische Universität München



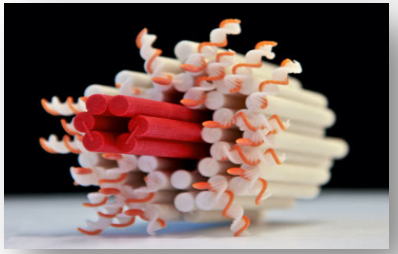
Uncovering the forces between nucleosomes using DNA origami



Single molecule dissection of stacking forces in DNA

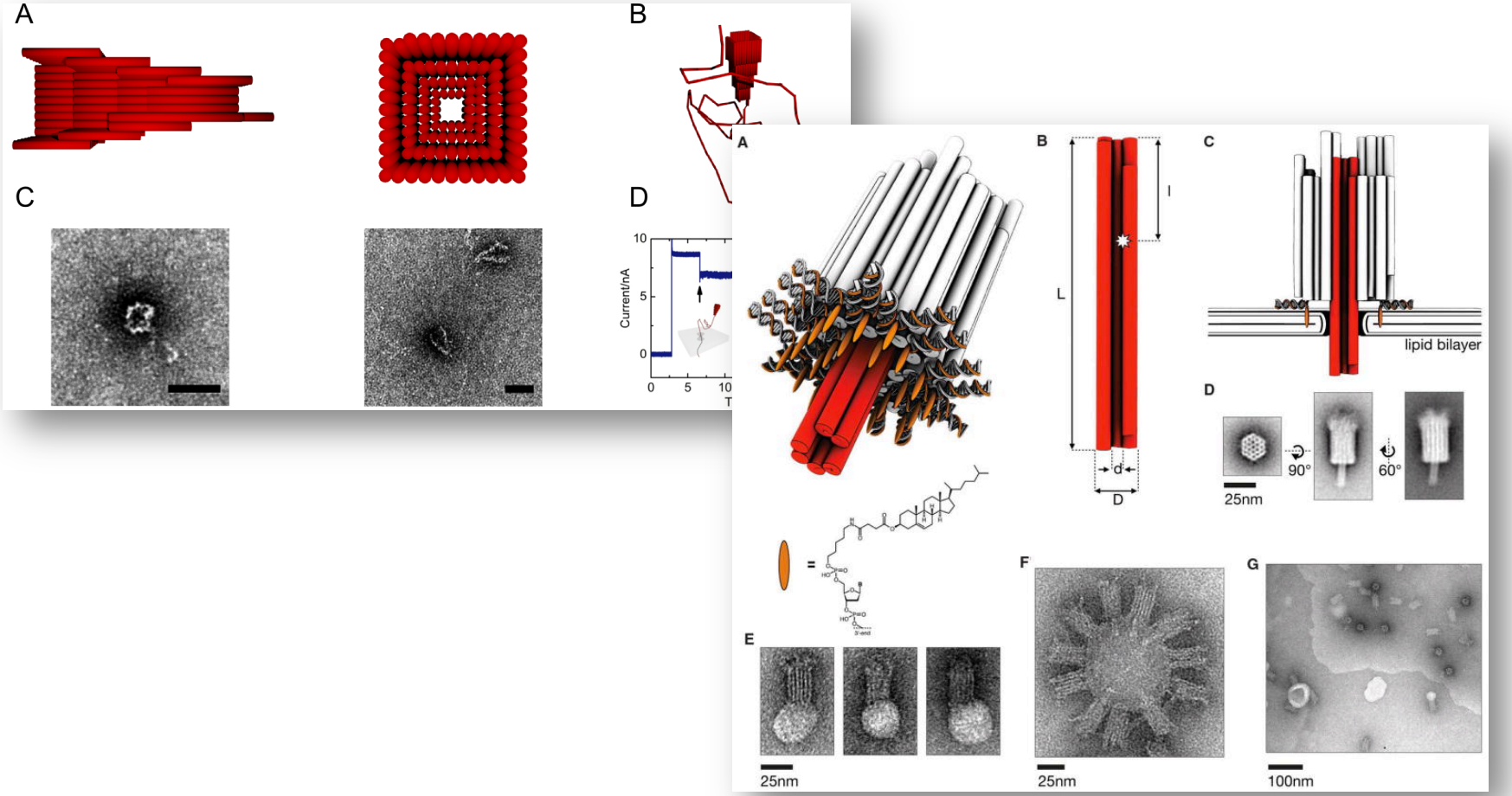


Nanoscale rotary apparatus formed from tight-fitting 3D DNA components



Synthetic lipid membrane channels formed by designed DNA nanostructures

Amazing Useful Stuff



So what's so great about DNA anyway?

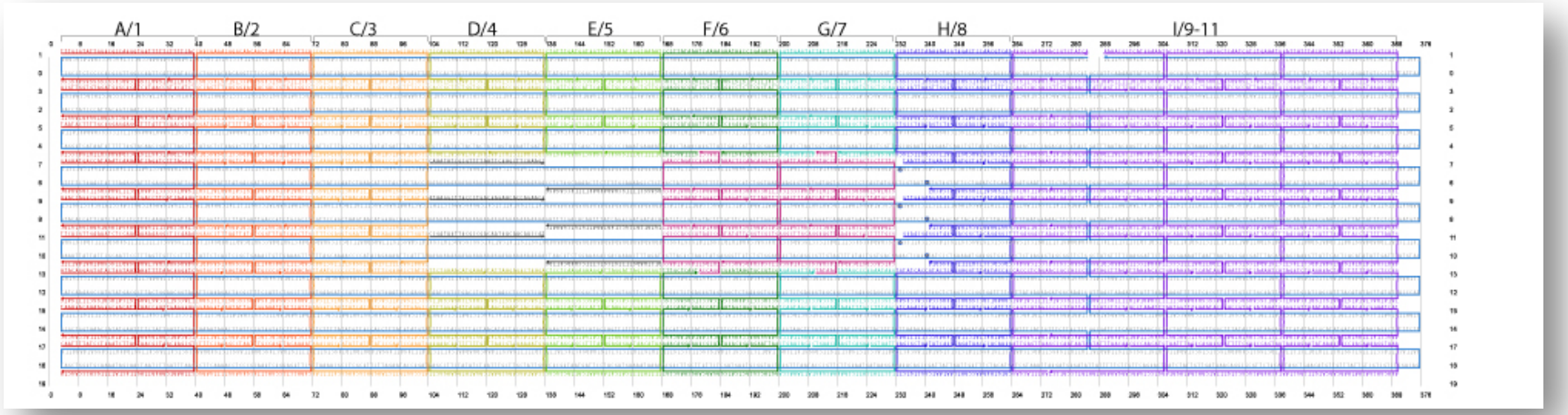
- Cheap and available
- Chemically and physically malleable
- "Simple" and well understood
- Dynamic and controllable
- Biologically compatible
- Works in solid and liquid phase

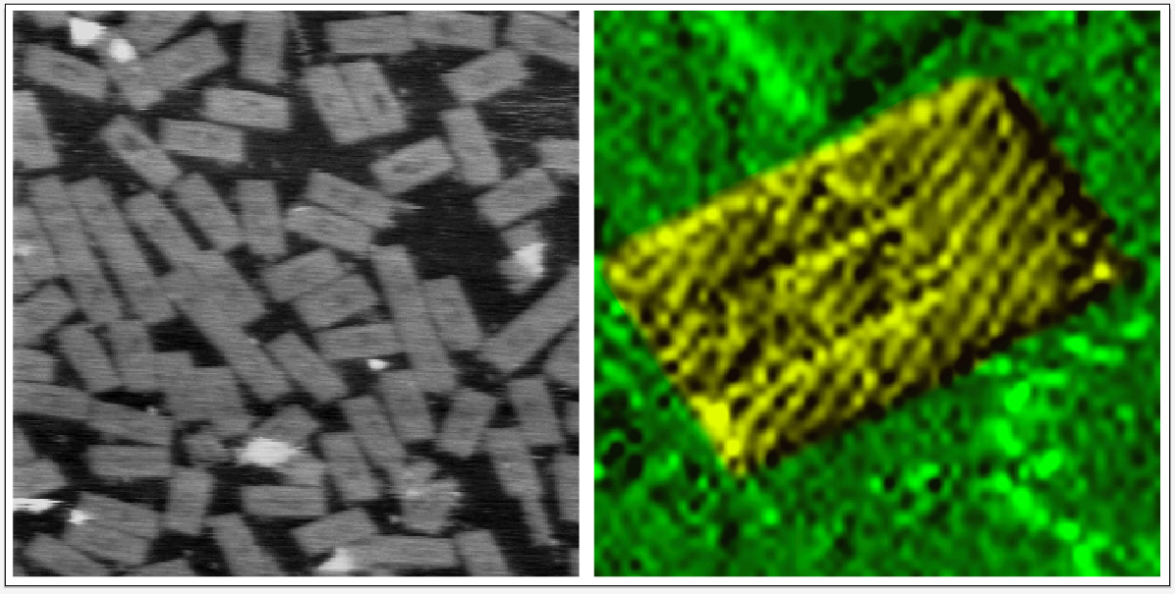
Hey, here's an idea (ca. 2010, let's build a machine that detects and measures molecular forces, kind of like an AFM, but at 4 trillionths of a dollar per machine that's like 1300000000000000(± .06) times cheaper!

Start with one of these…

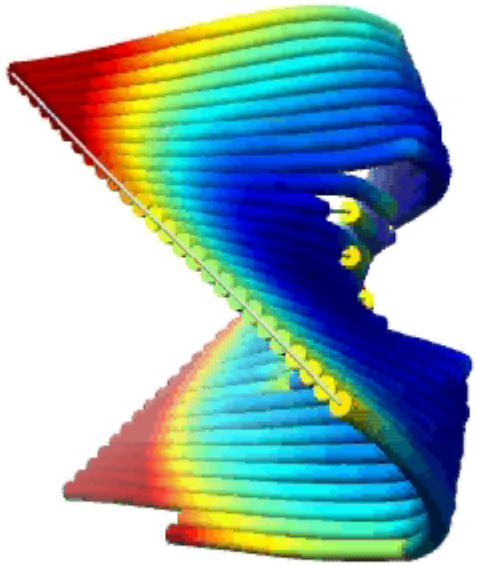
but a billion times smaller, and cooler.

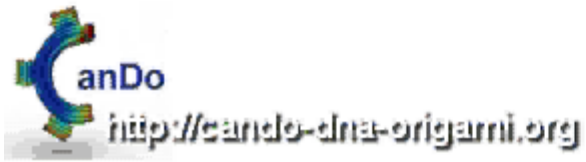
Slider v.1.x



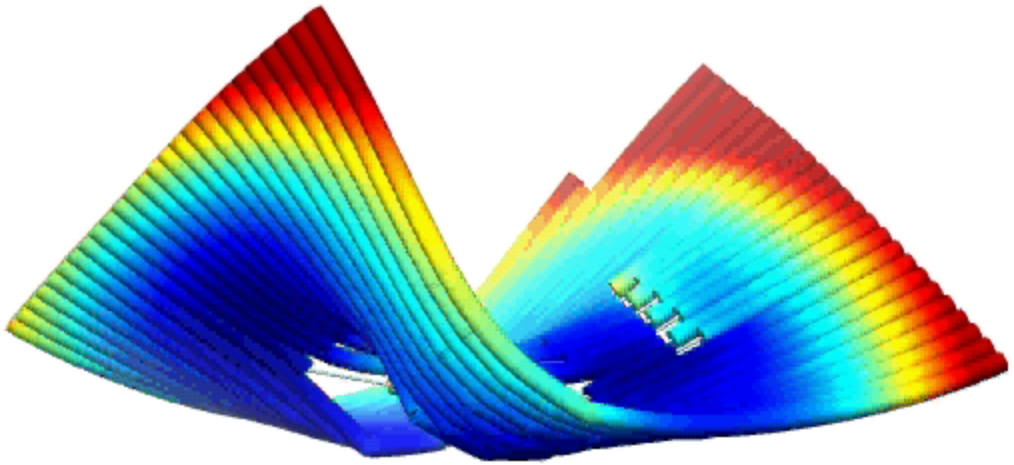


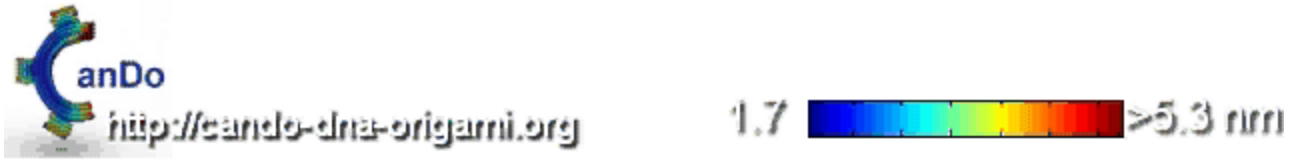
Slider v.1.x



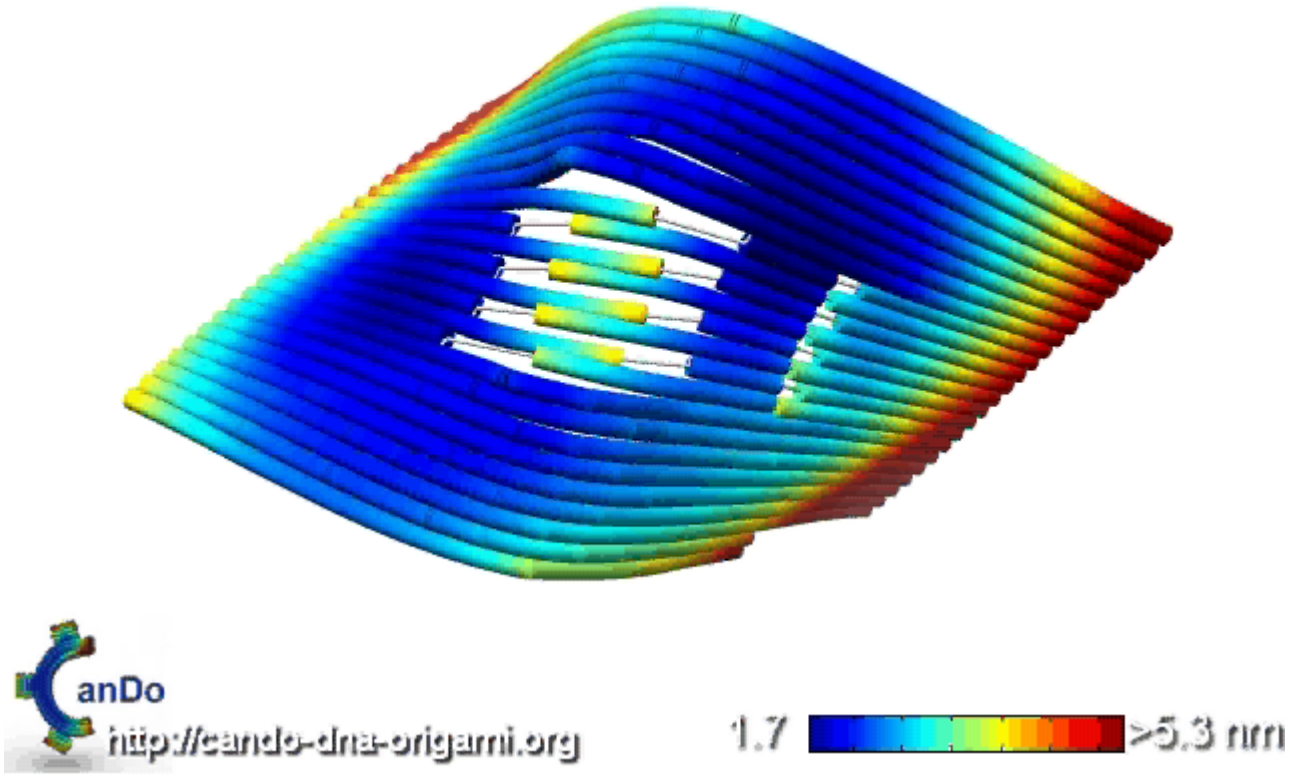


Slider v.1.x

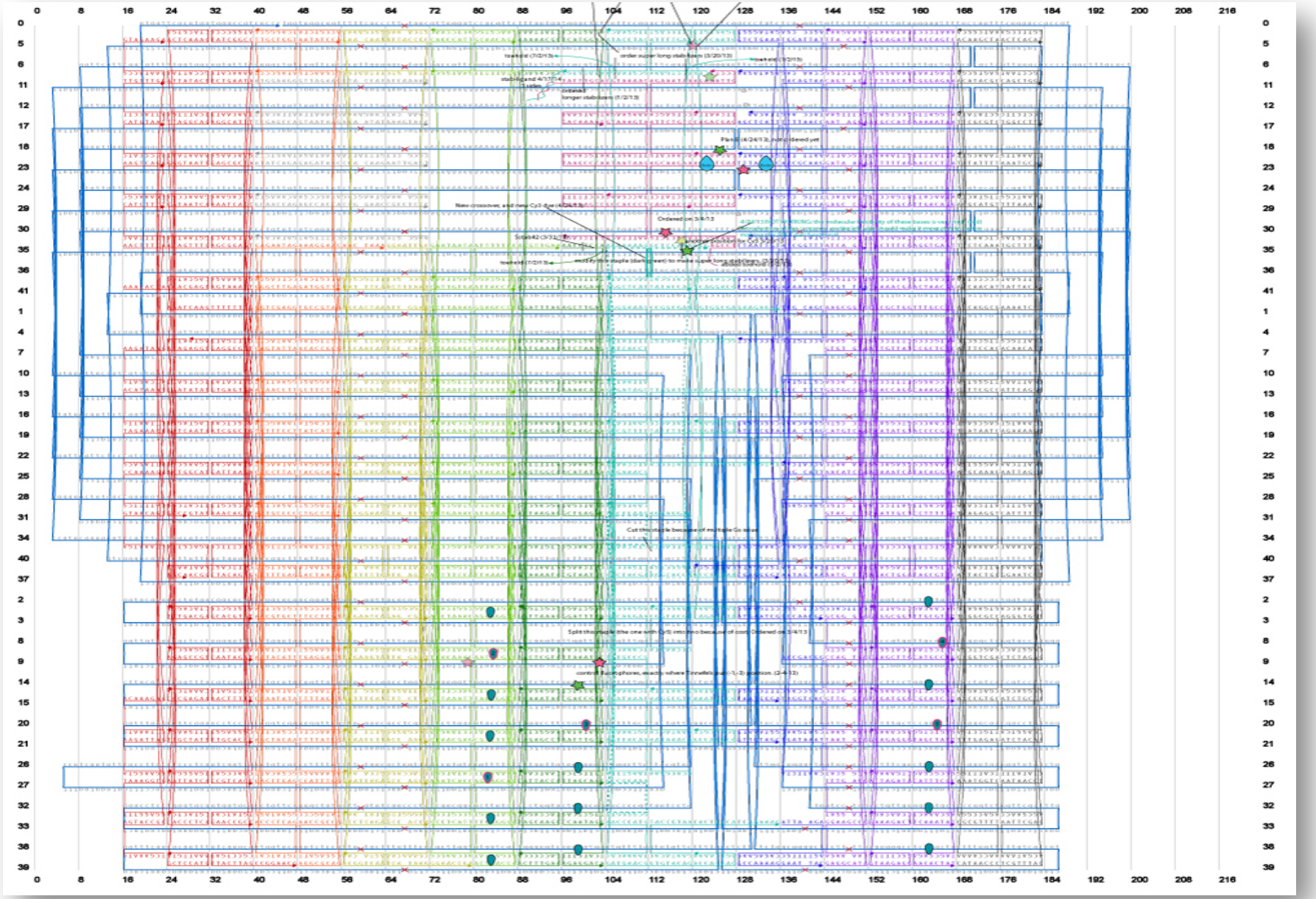




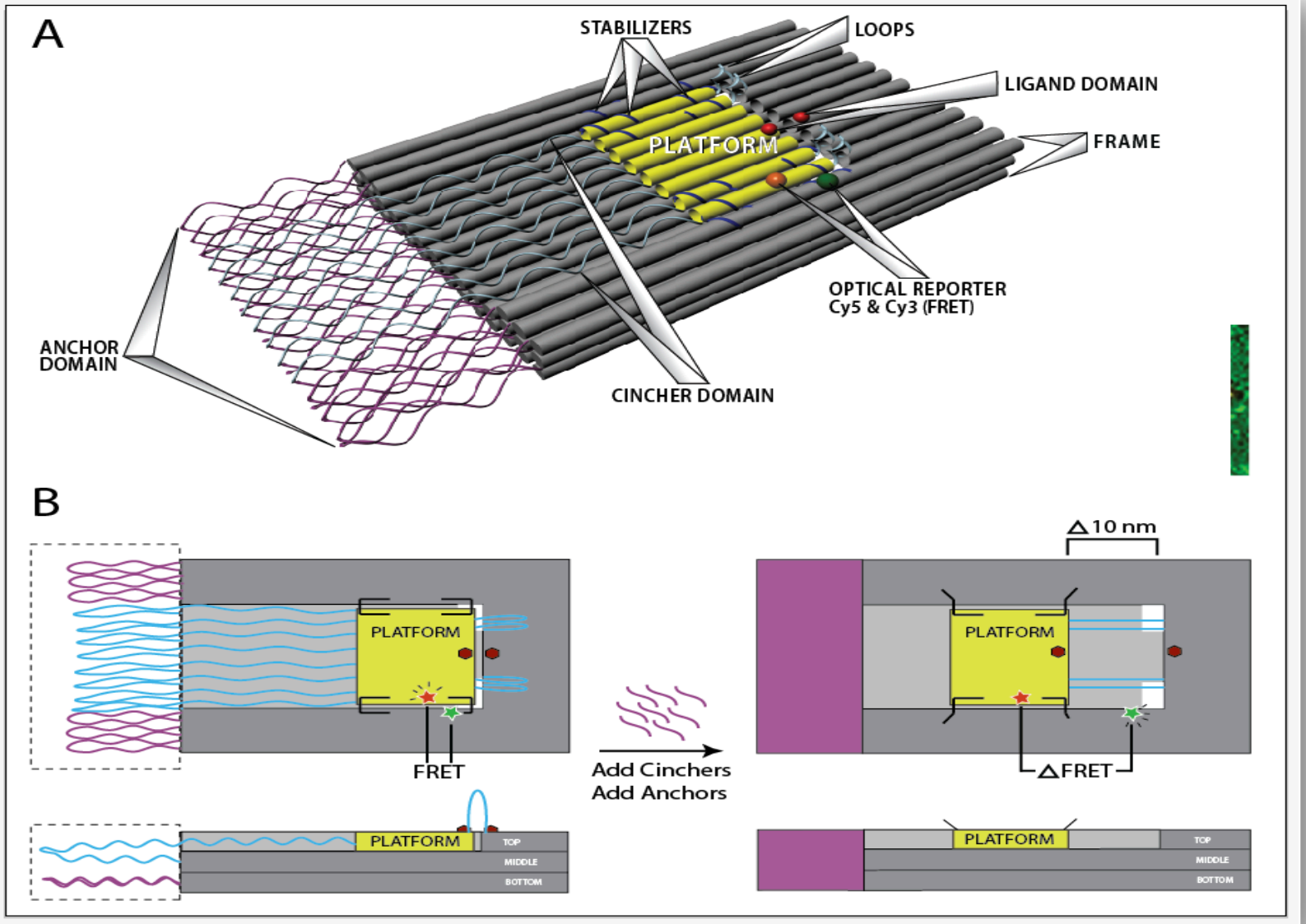
Slider v.1.x



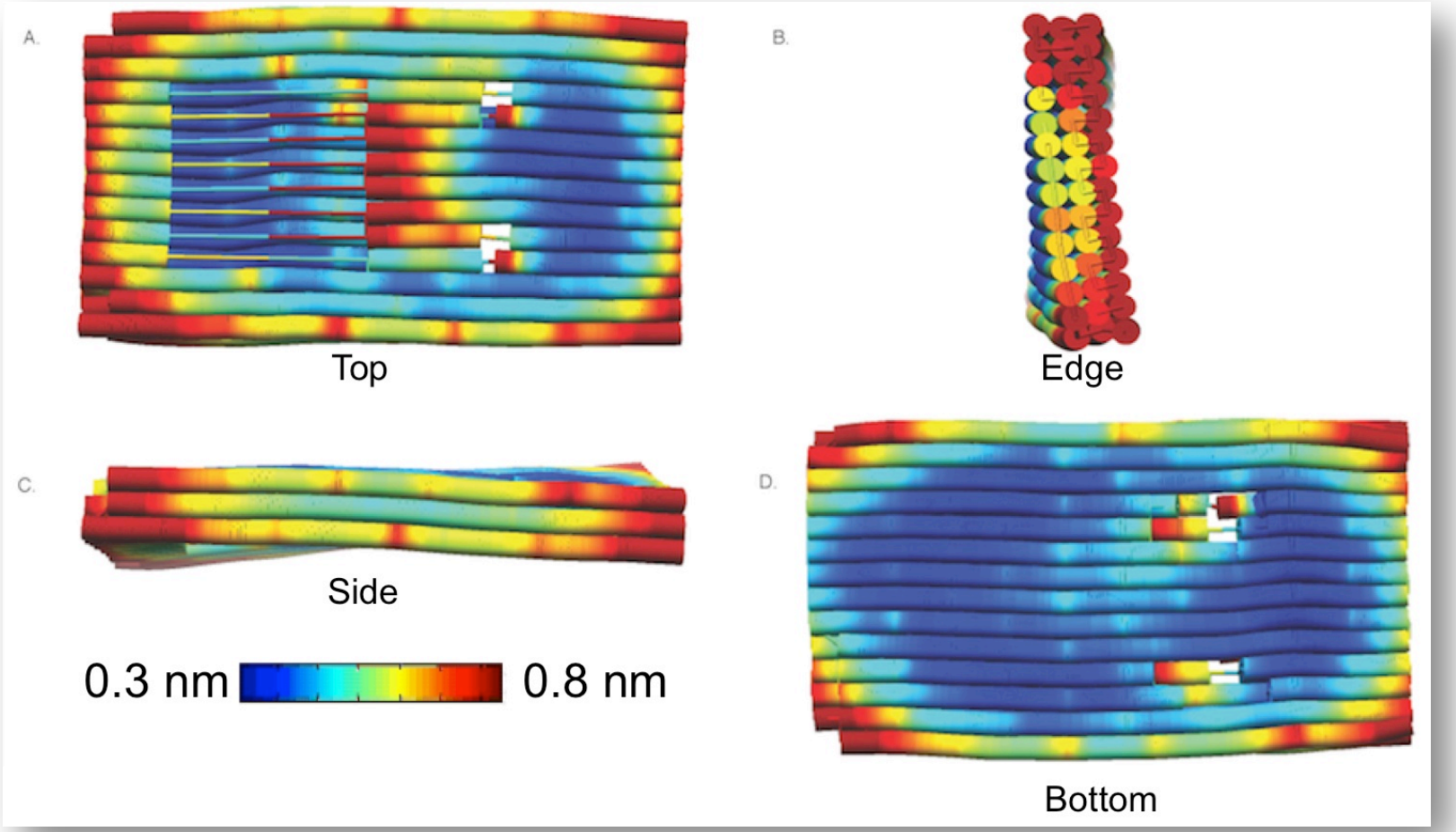
Enter Divita Mathur: Slider v.2.x



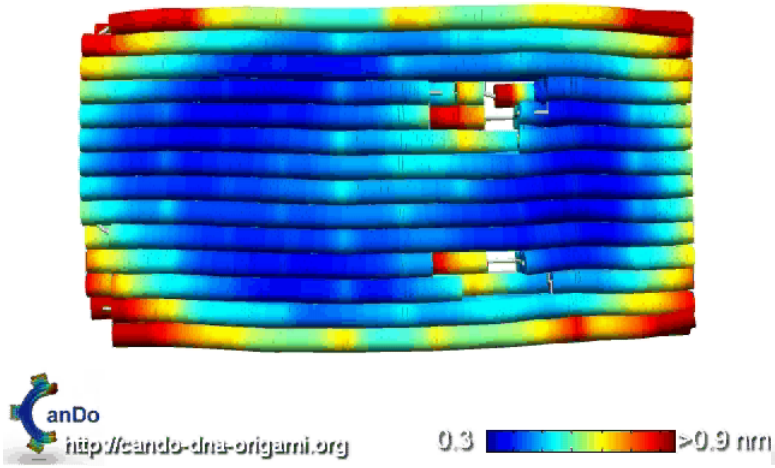
The “Slider” v.2.0



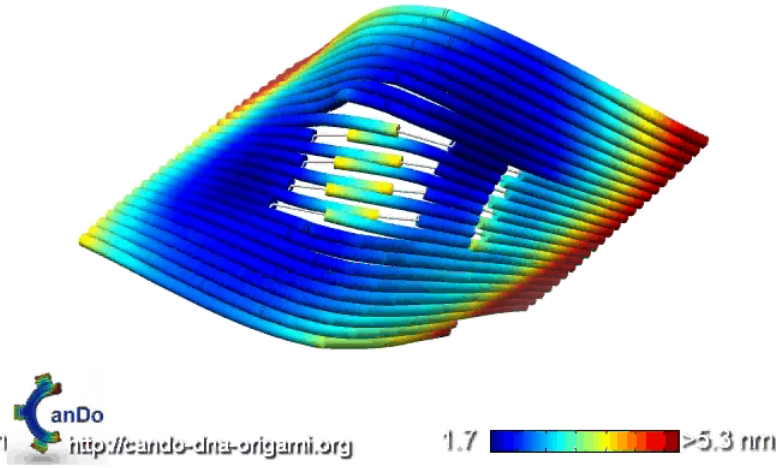
Slider v.2.x Stability



Slider 2.x Top View

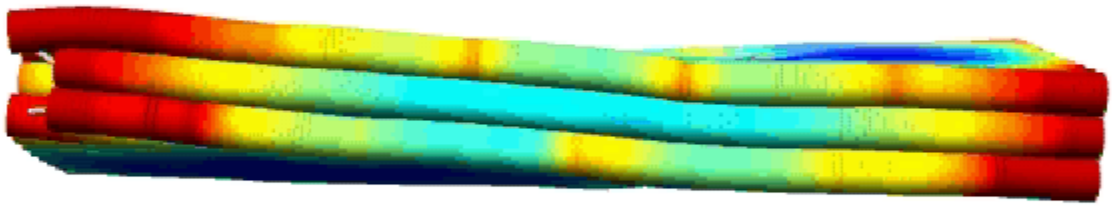


v.2.x

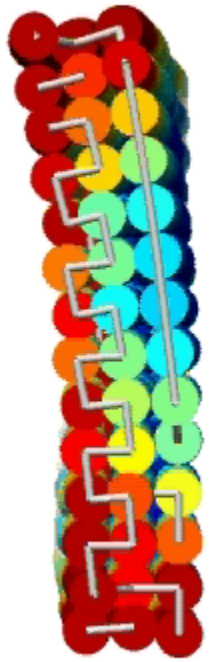


v.1.x

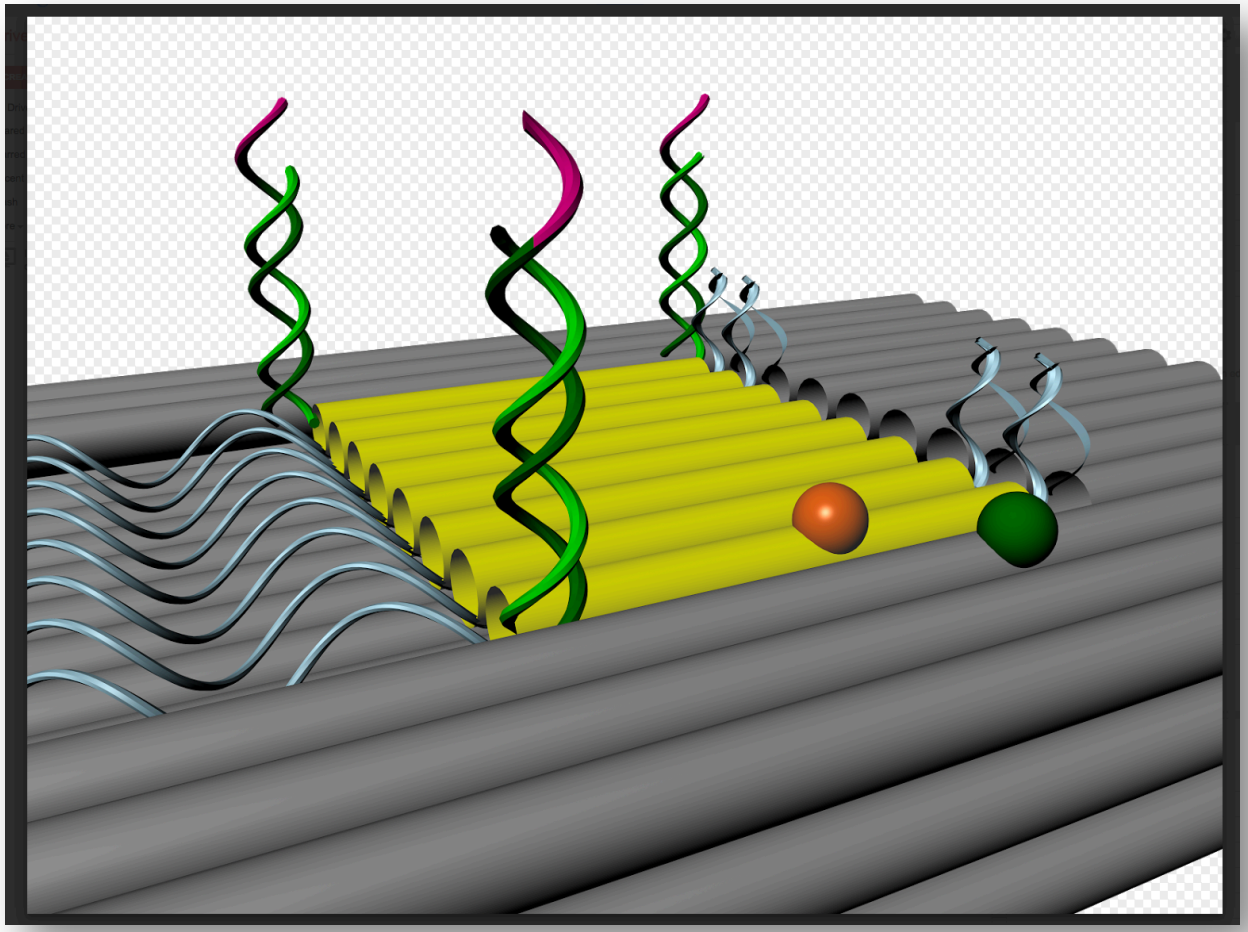
Slider 2.x Side View



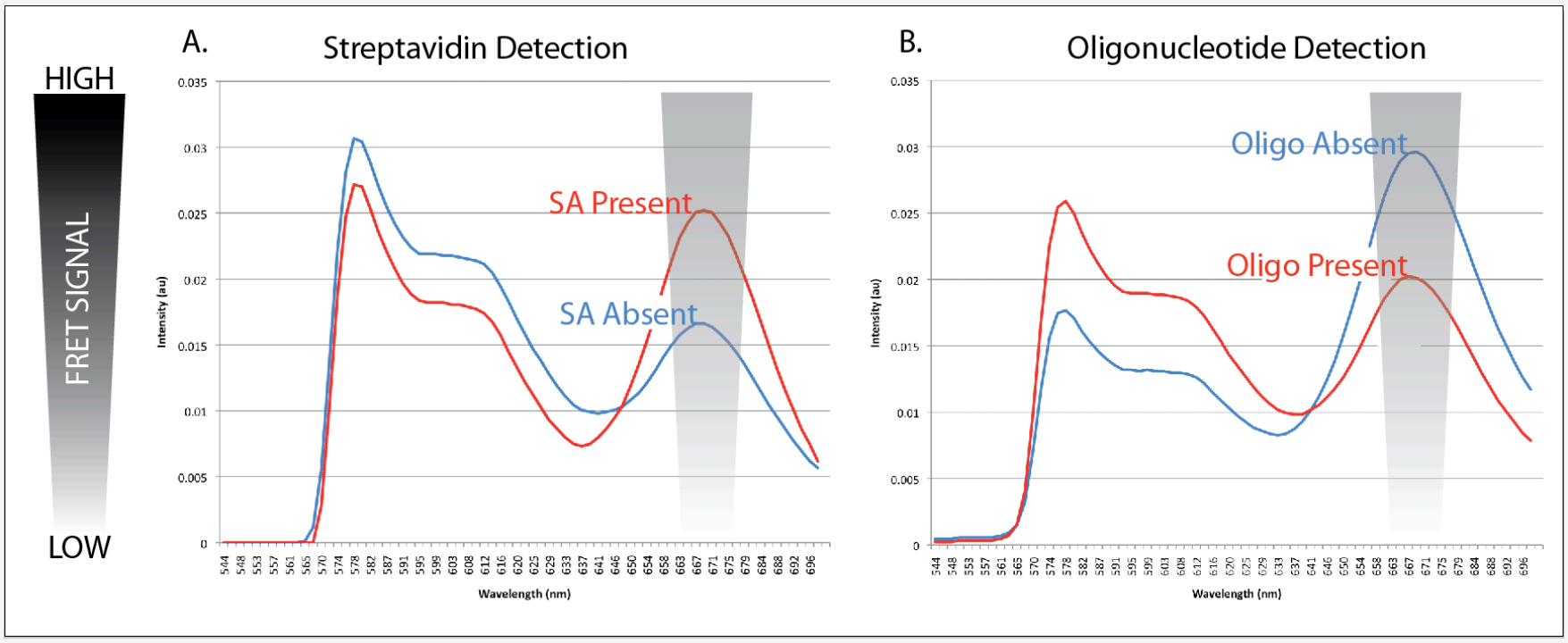
Slider 2.x End View



Slider is a Platform Technology



And The Darn Thing Actually Works!

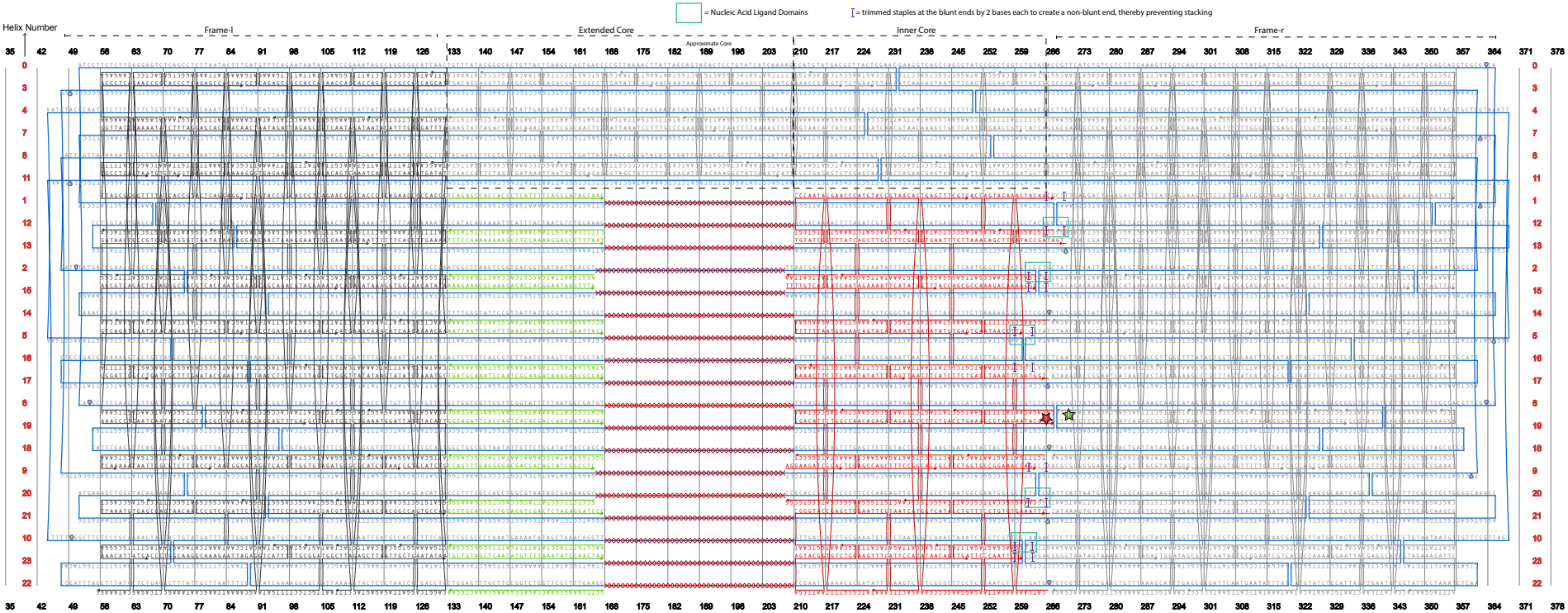


Four Years and One Great Student Later…

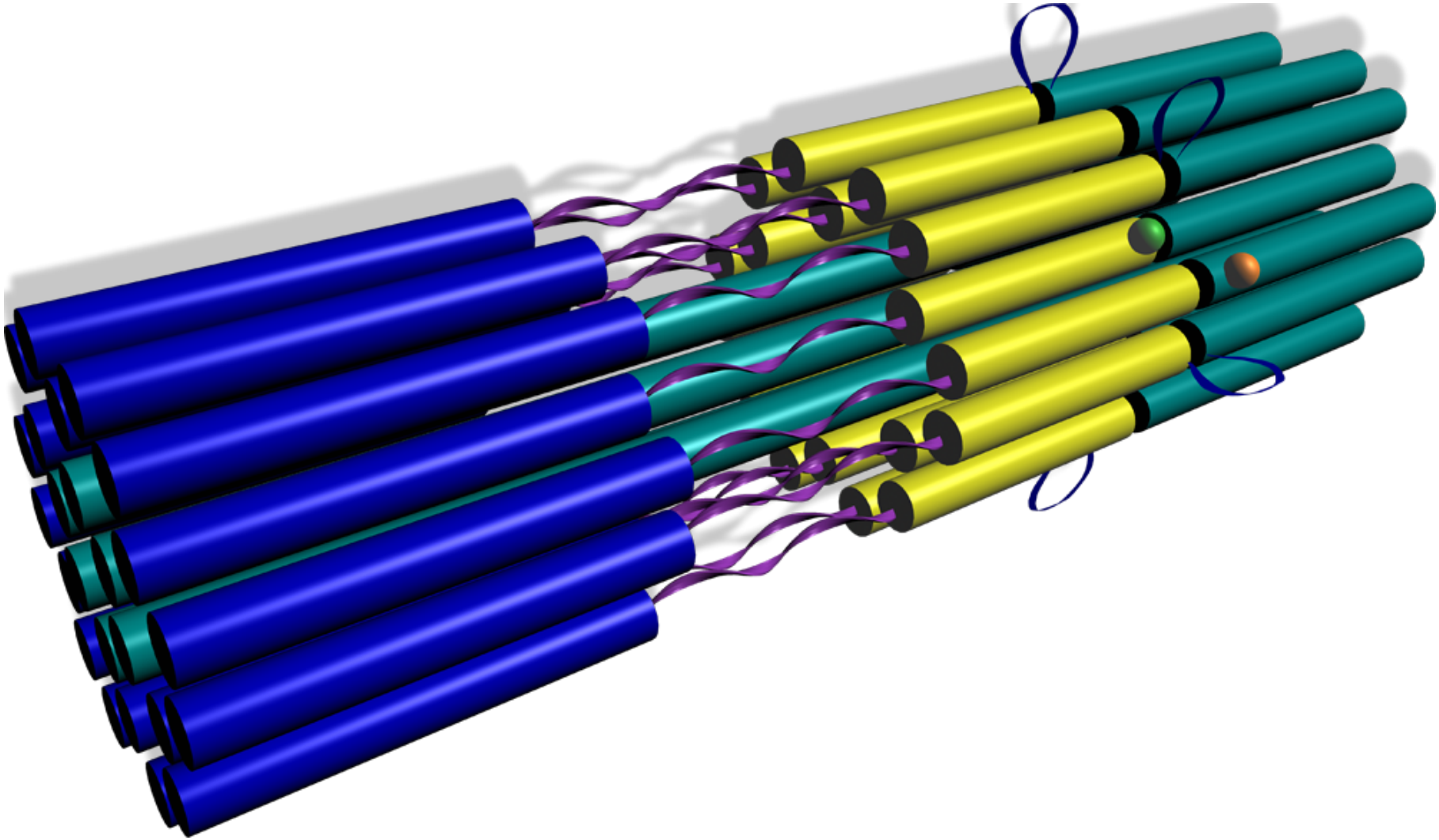
Slider .v.3.x

Dr. Divita Mathur

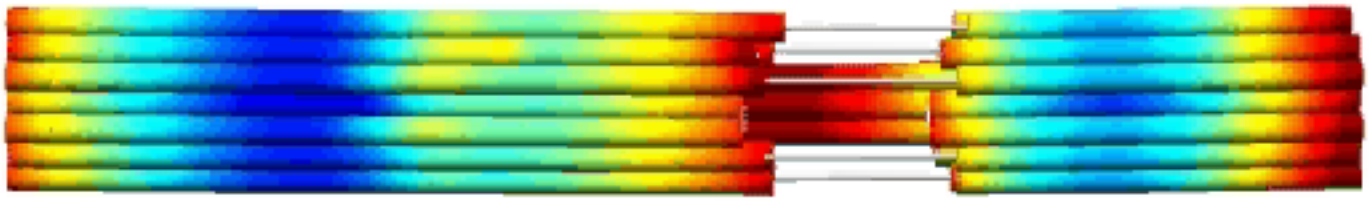
Beautiful!


= Nucleic Acid Ligand Domains
= trimmed staples at the blunt ends by 2 bases each to create a non-blunt end, thereby preventing stacking
Helix Number
Frame-l
Extended Core
Approximate Core
Inner Core
Frame-r

Beautiful!



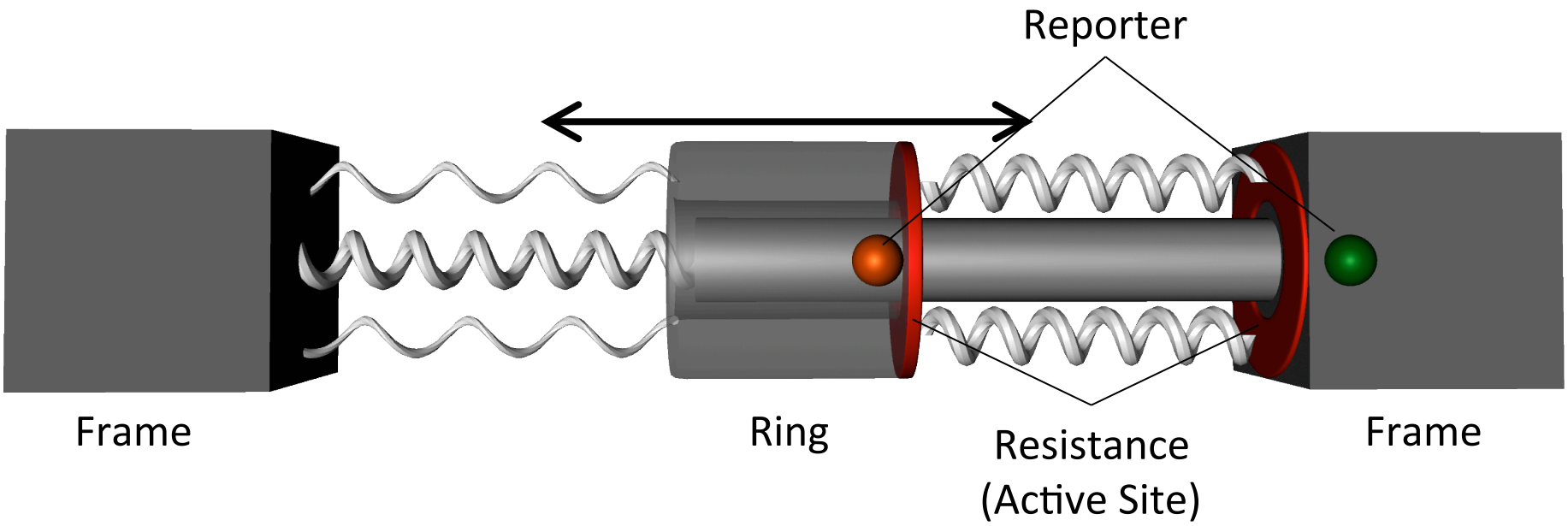
Slider v.3.x Stability



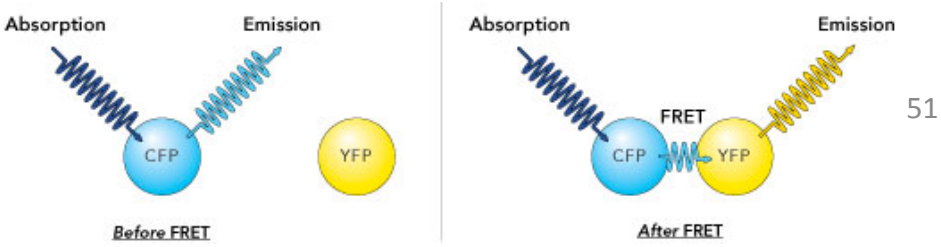
anDo
http://cando-dna-origami.org

Mechanical Architecture

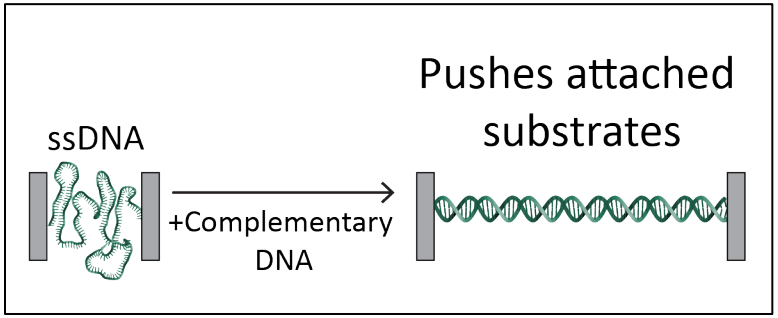
Motion in ring is induced by spring elements

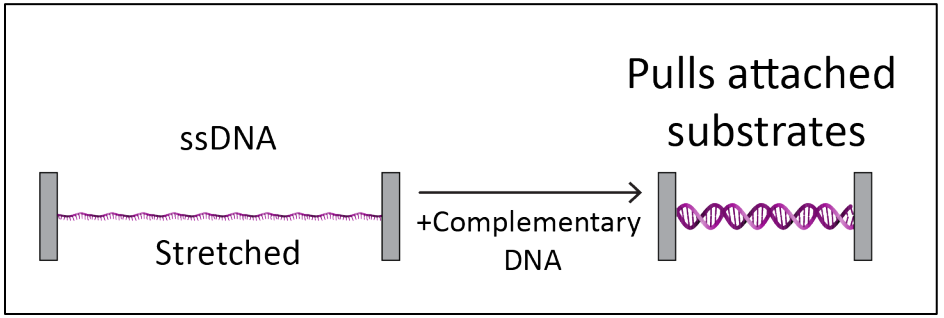


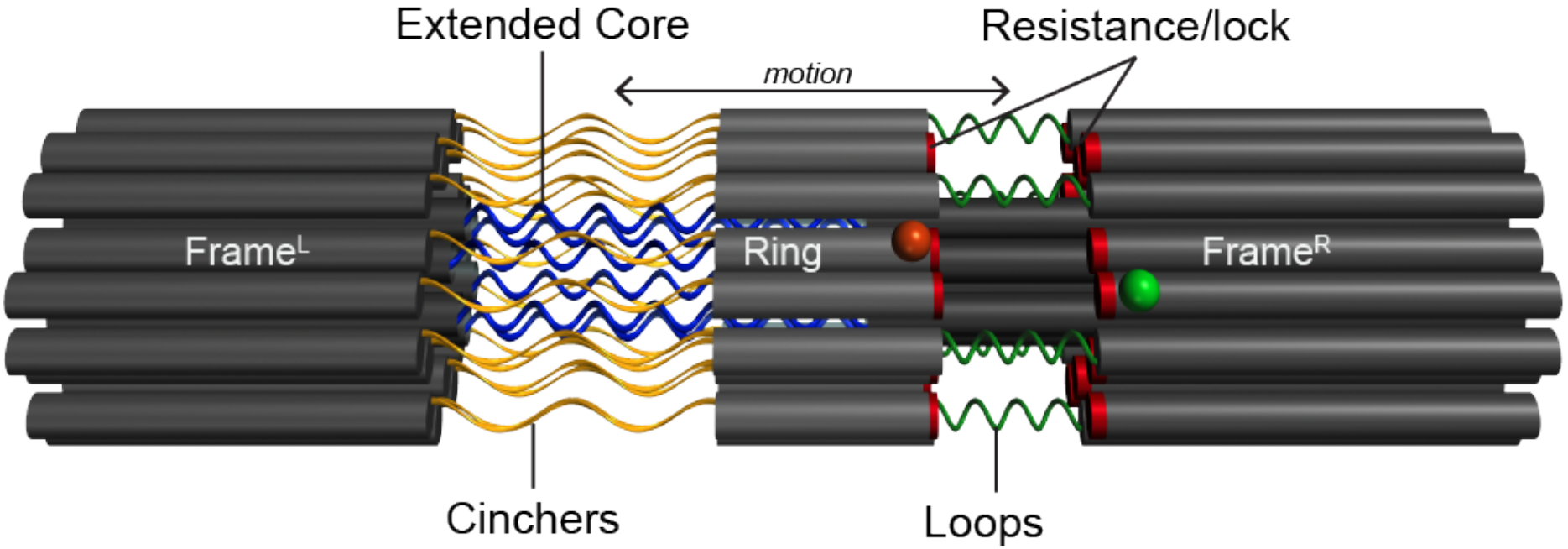
FRET (Fluorescence Resonance Energy Transfer)

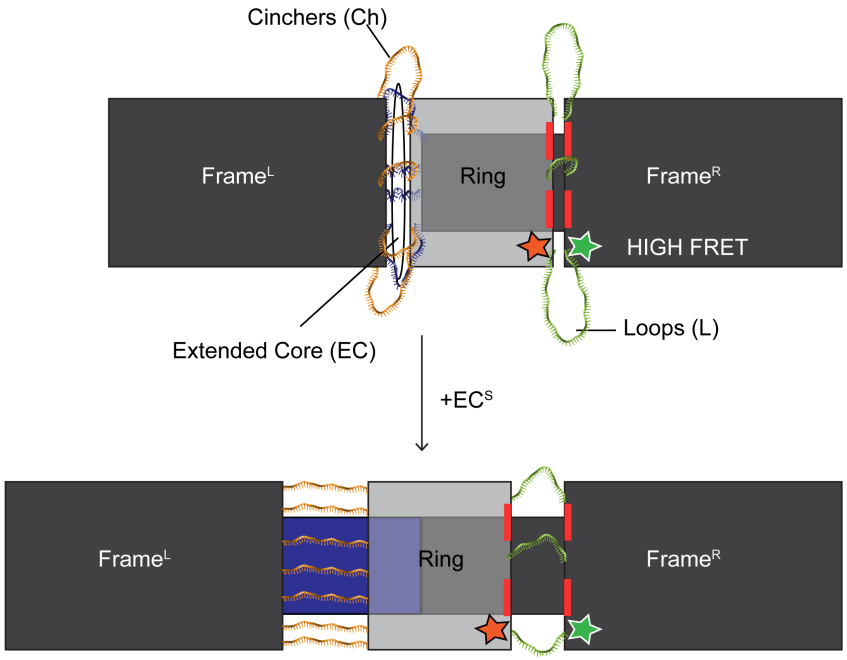


Force induction via DNA hybridization

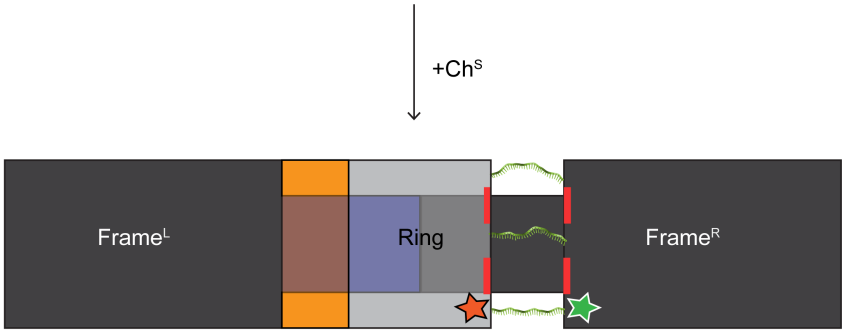




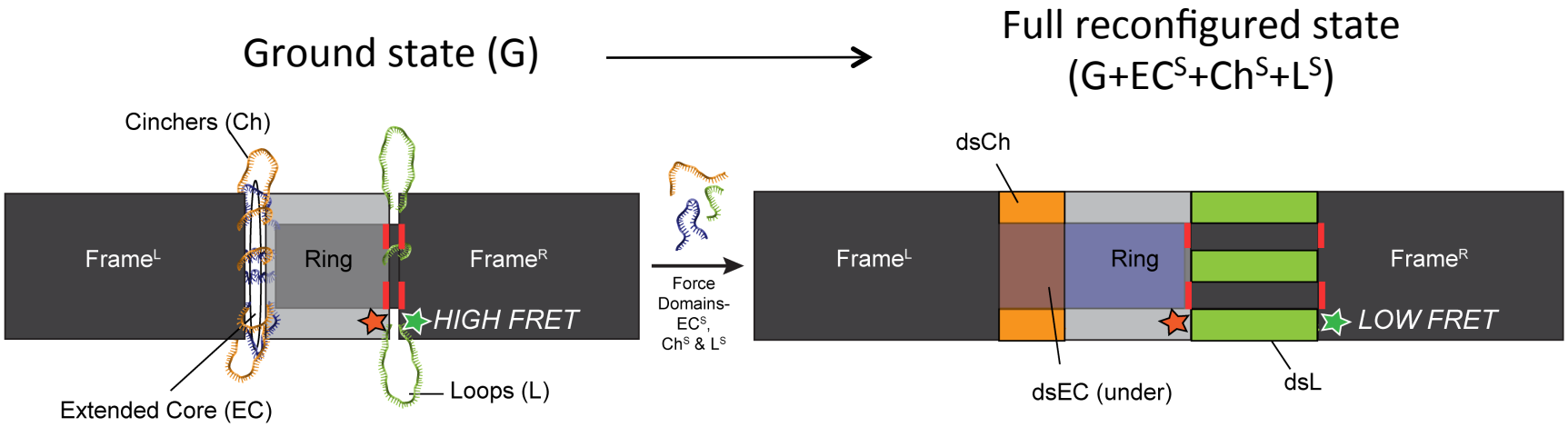


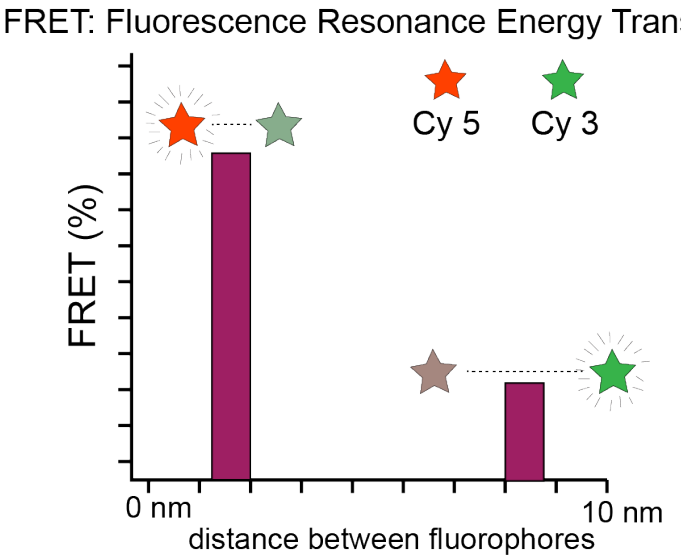


Forms rigid extended inner core, pushes frame^{L} which creates an anchor for Ch domain



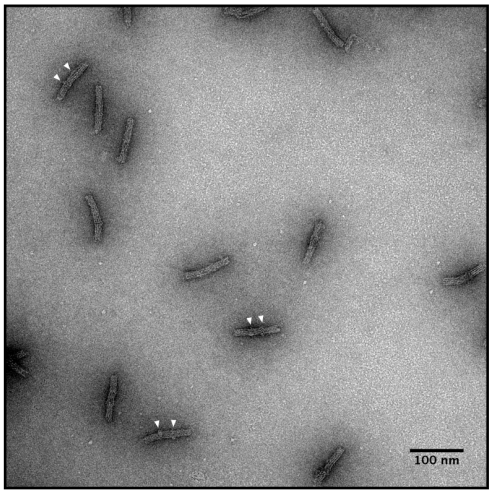
Overall internal reconfiguration



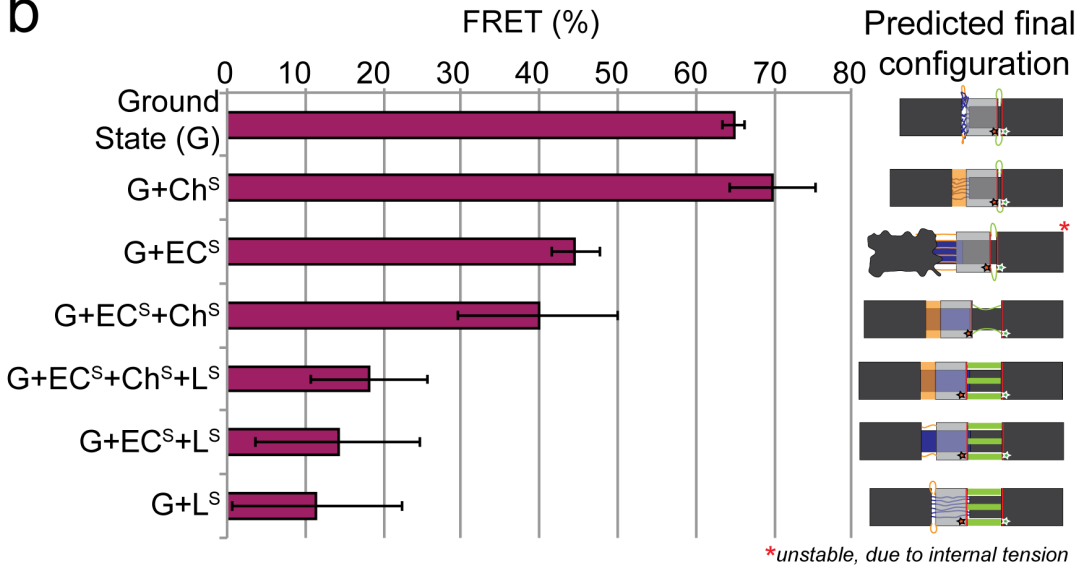


Effect of force domains on ring motion

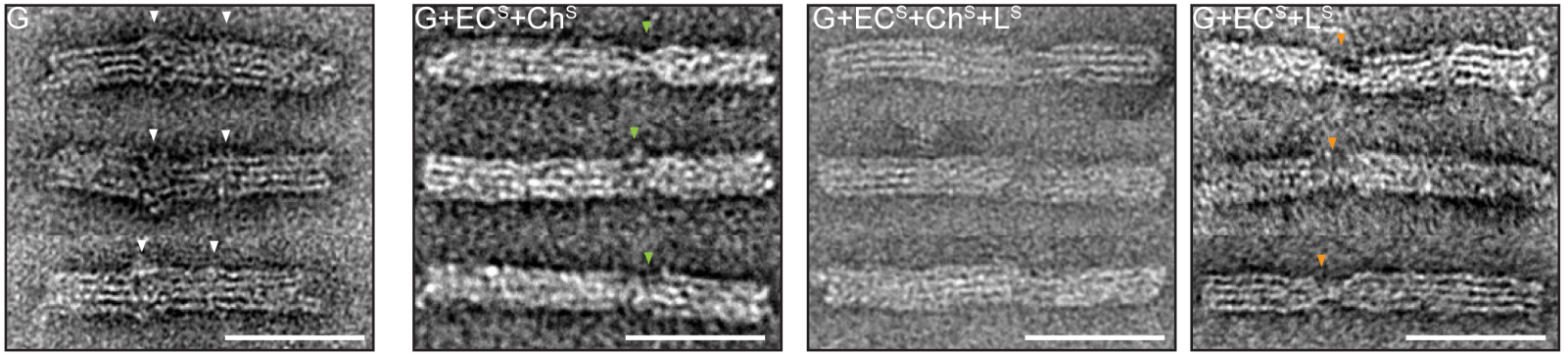
a



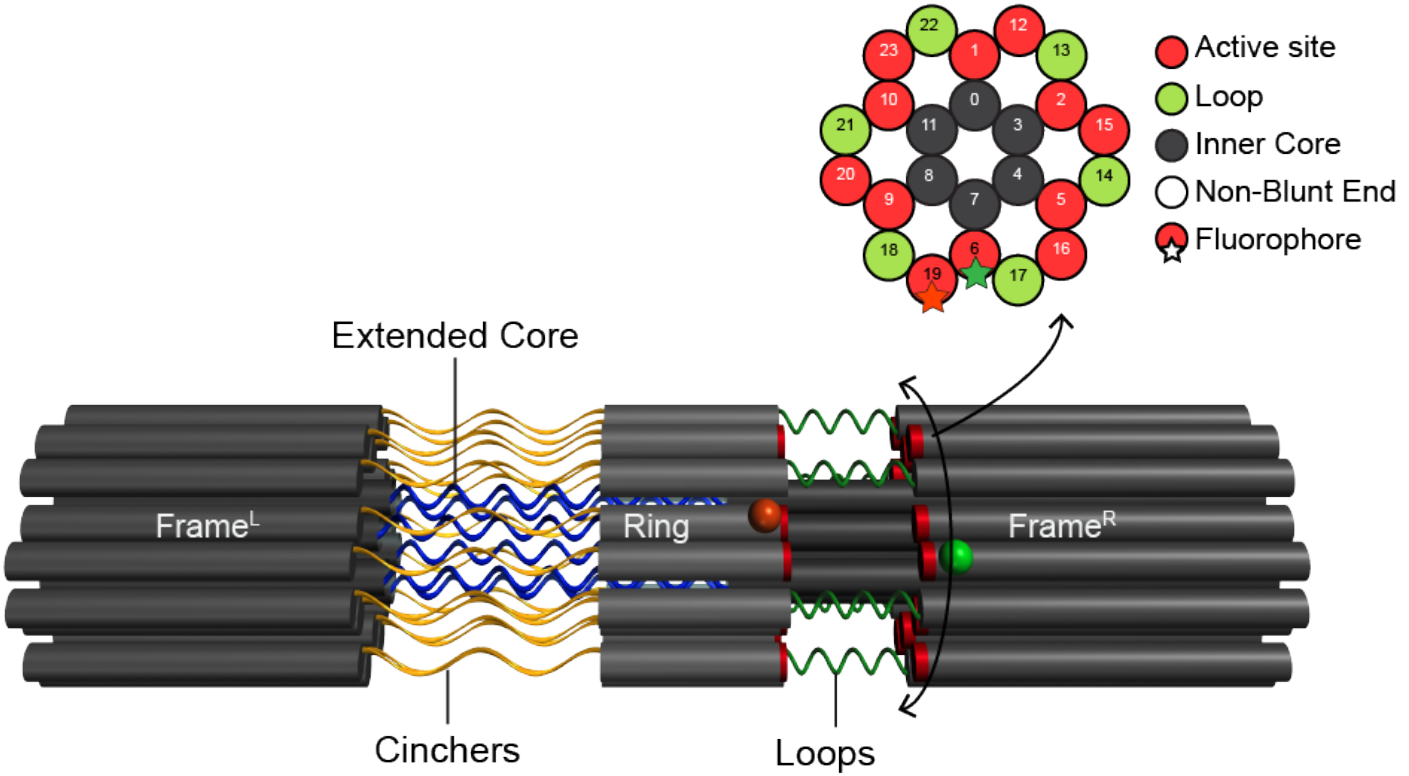
b

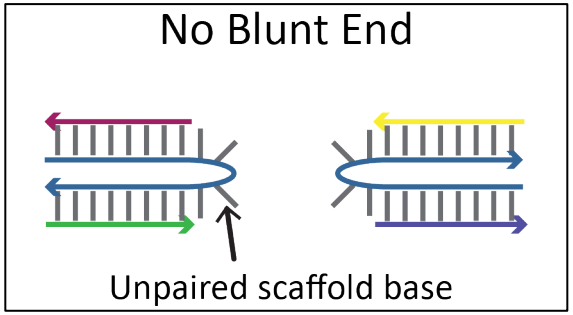


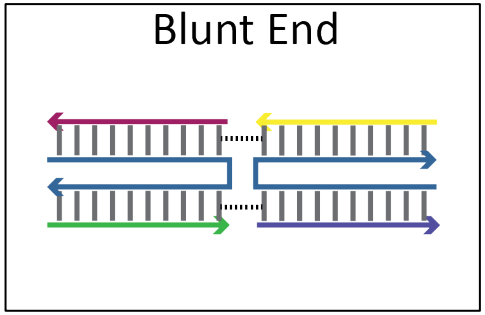
c



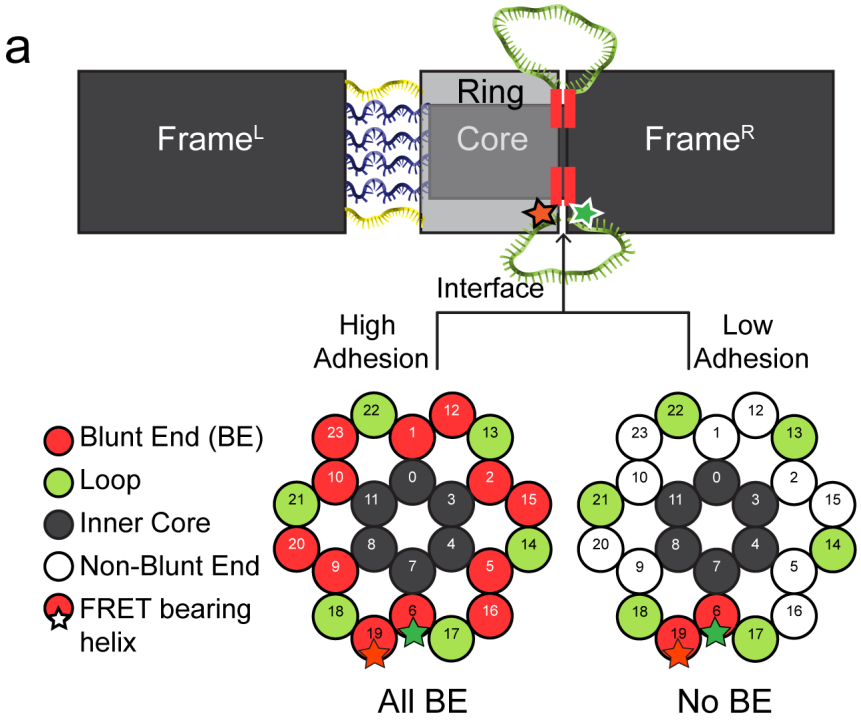
Ring motion is challenged by active site modification

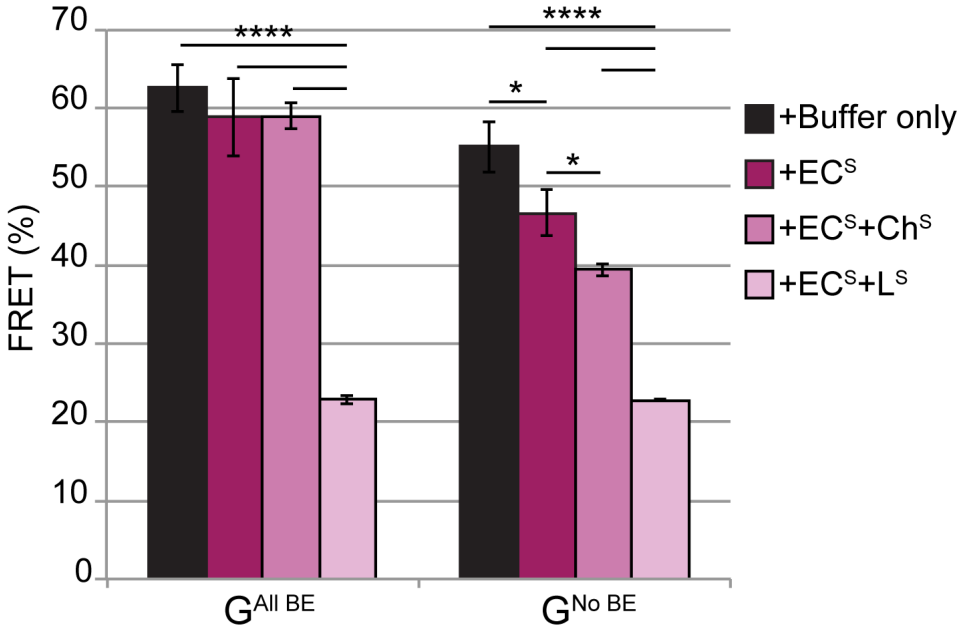






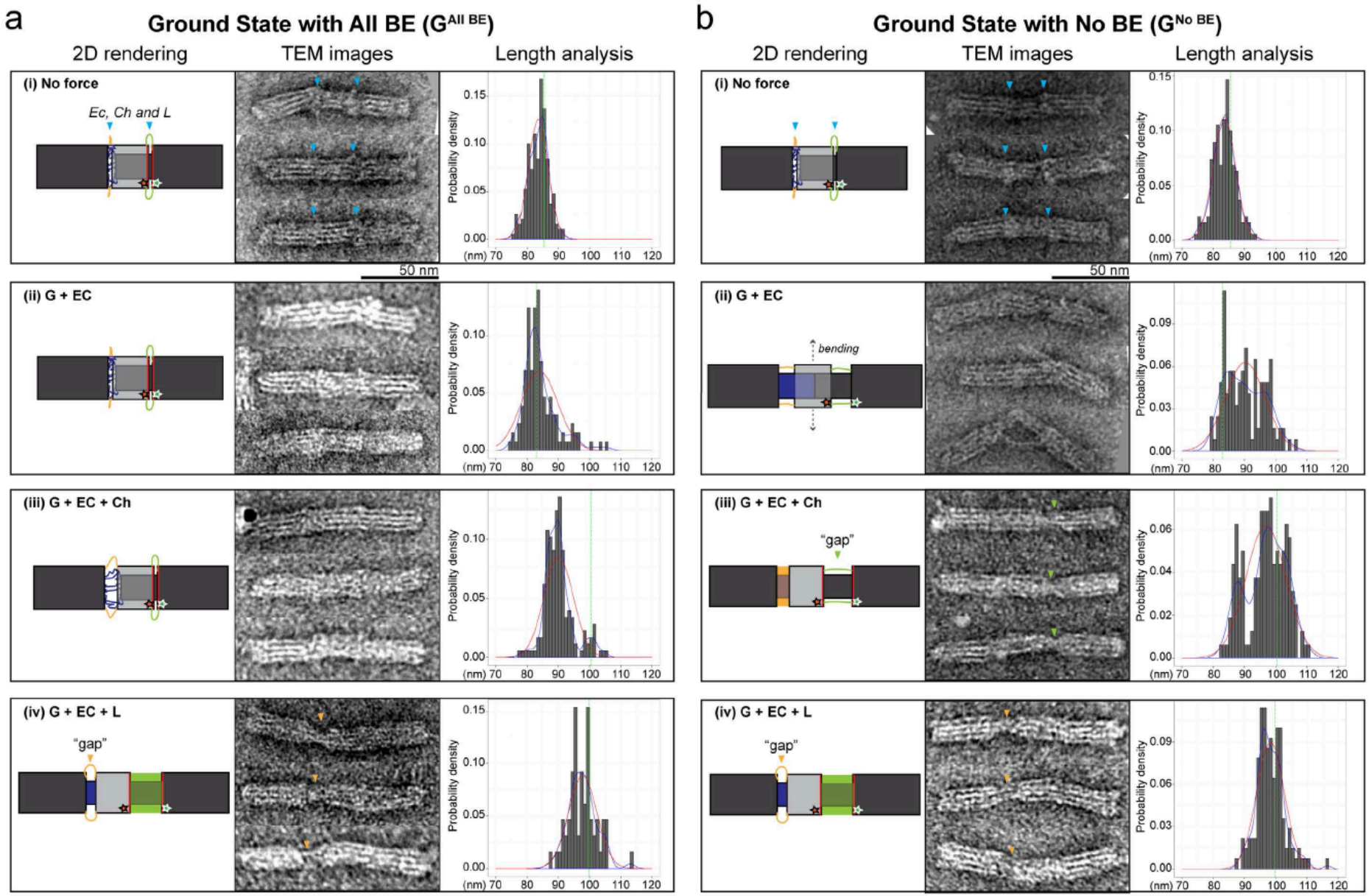
Blunt end stacking versus force domains



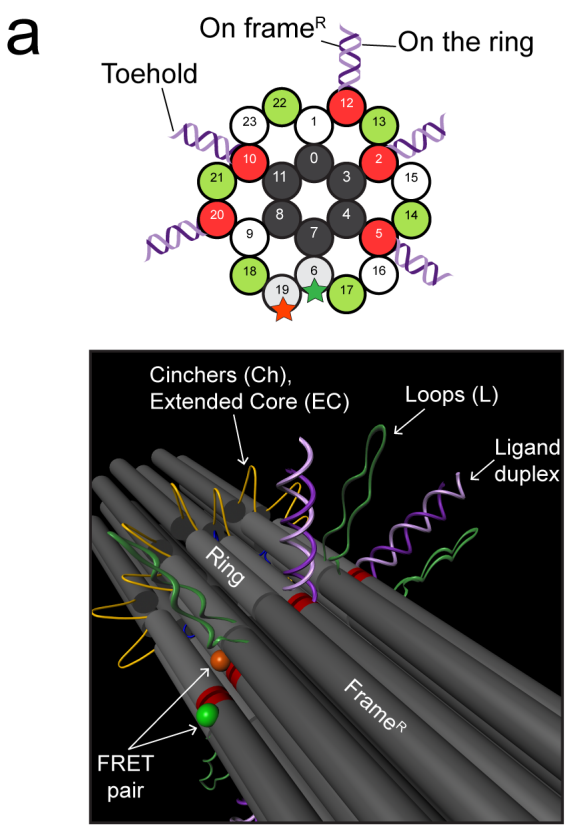


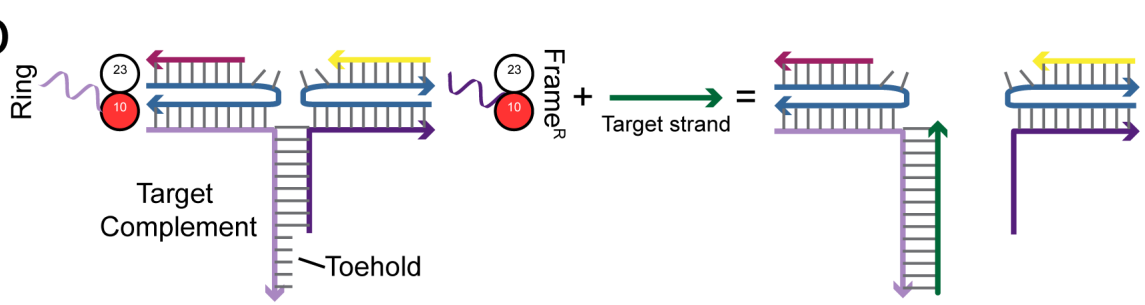
* $p<0.05$; **** $p<0.0001$

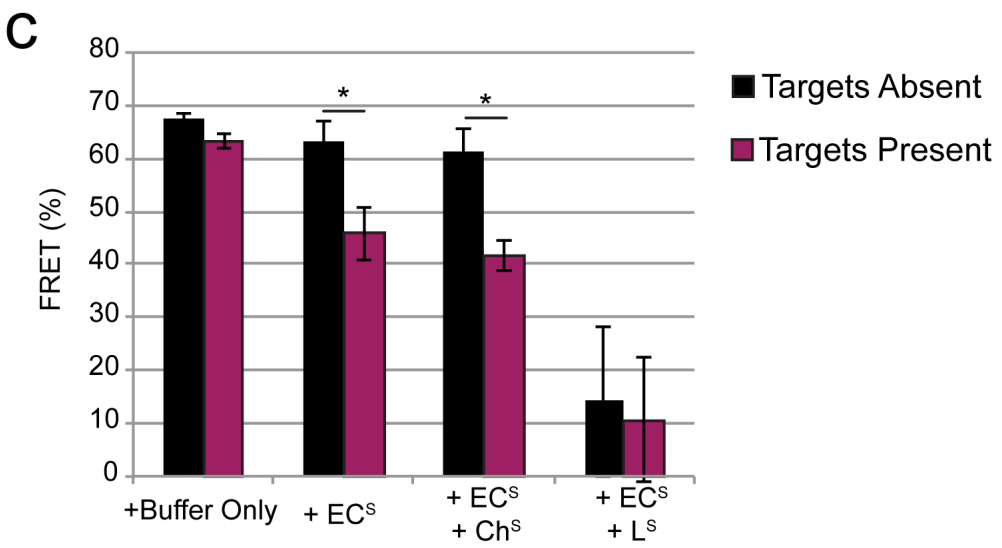
Effect of force domains on OPTIMuS length



DNA duplex versus force domains

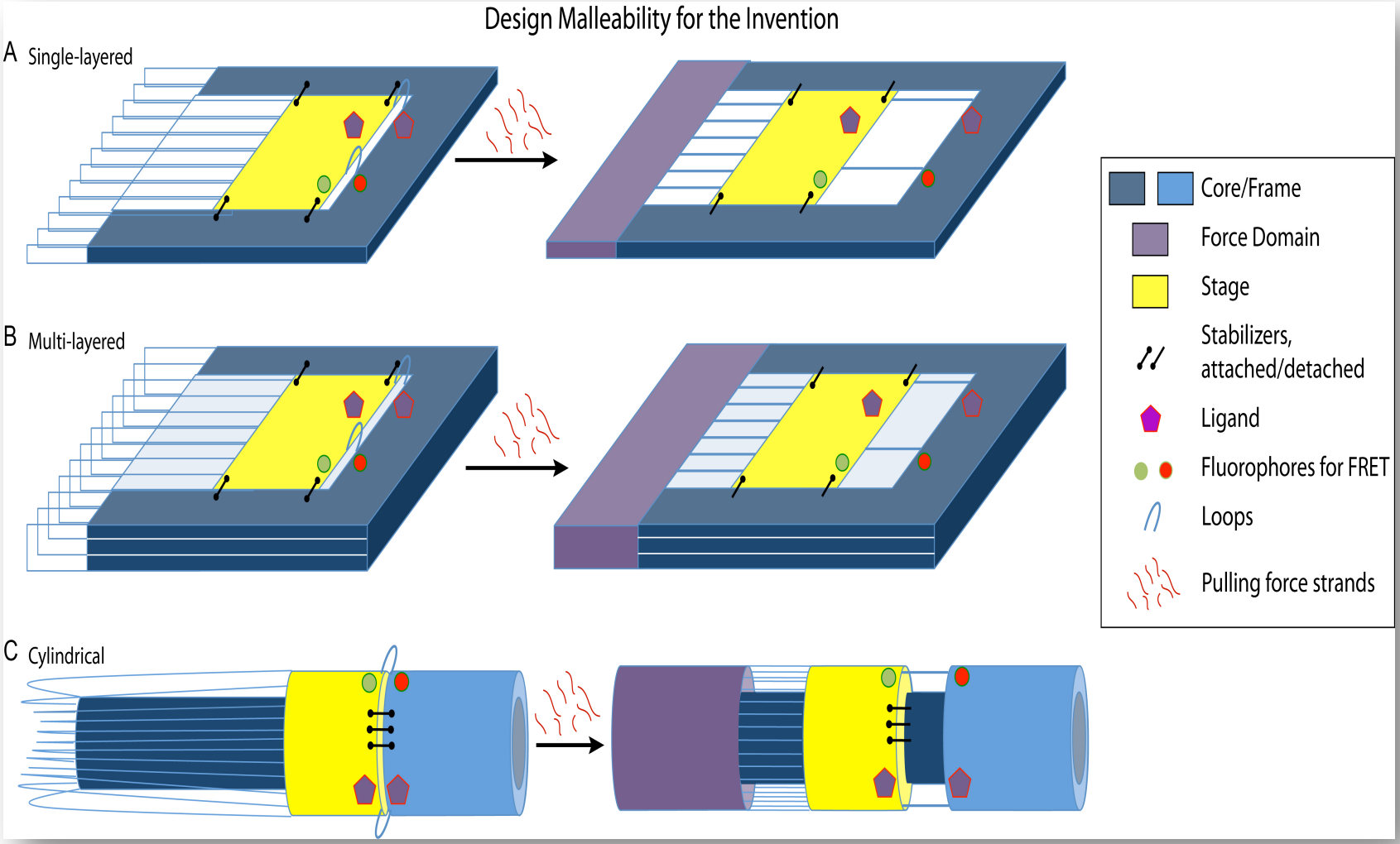




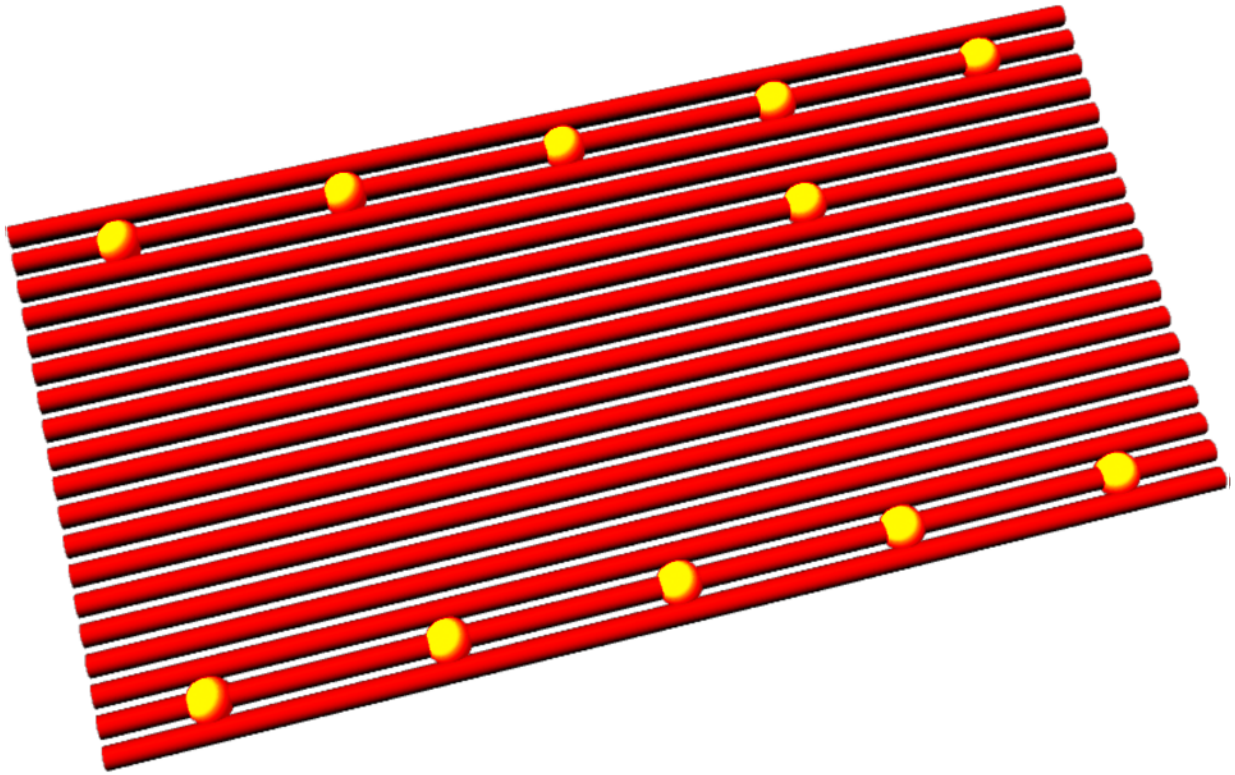


Detection of Ebola genome mimic

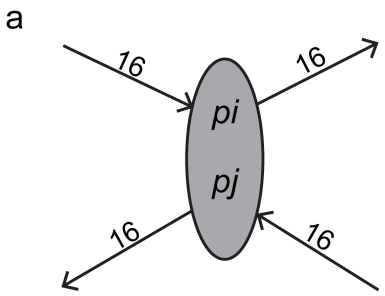
Slider Evolutionary Tree

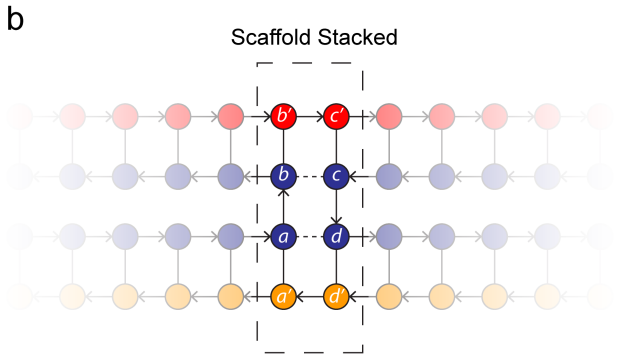


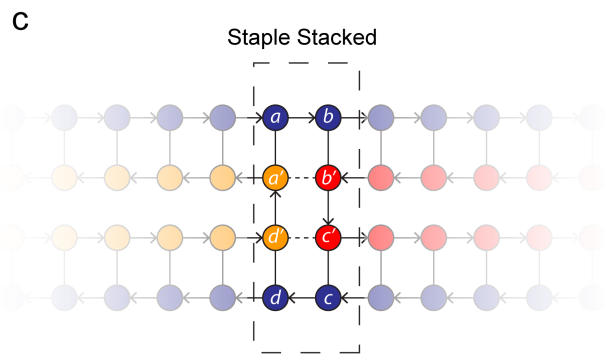
Shapeshifting DNA Nanostructures

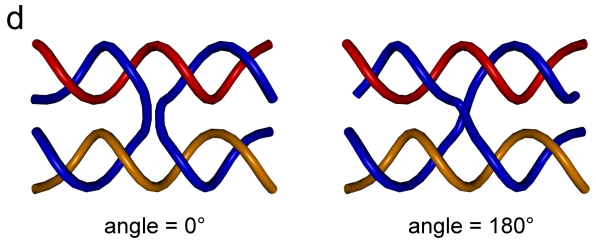


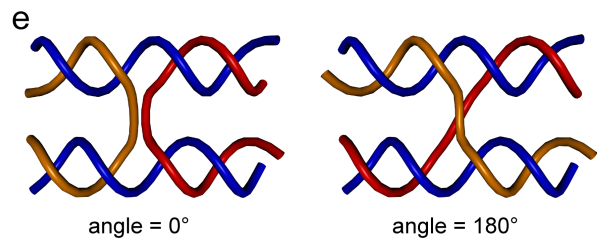
Modeling

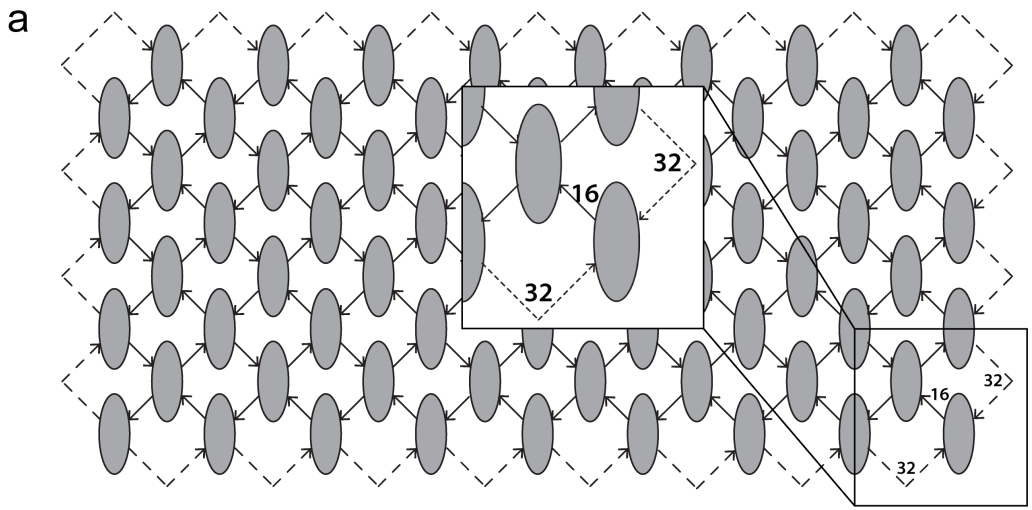

a
16
16
pi
pj
16
16

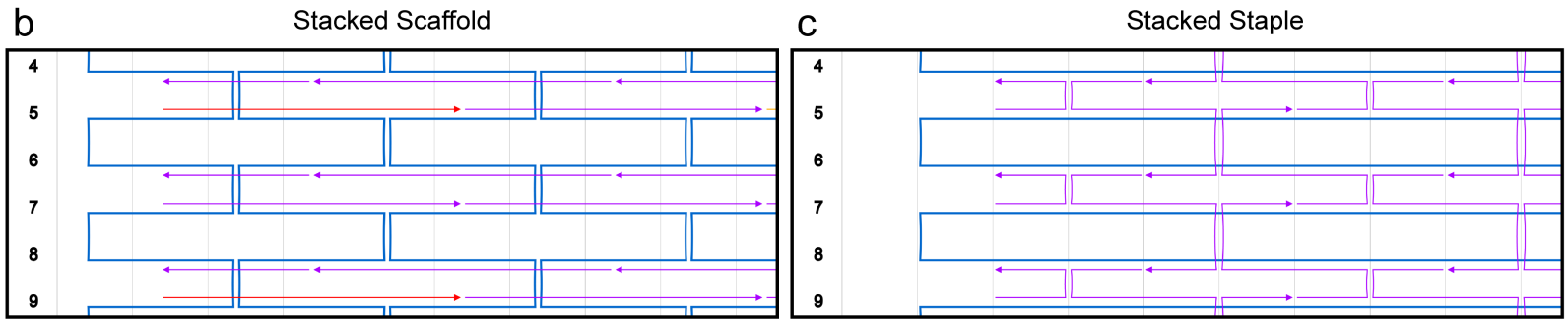

b
Scaffold Stacked
b'
c'
b
c
a
d
a'
d'

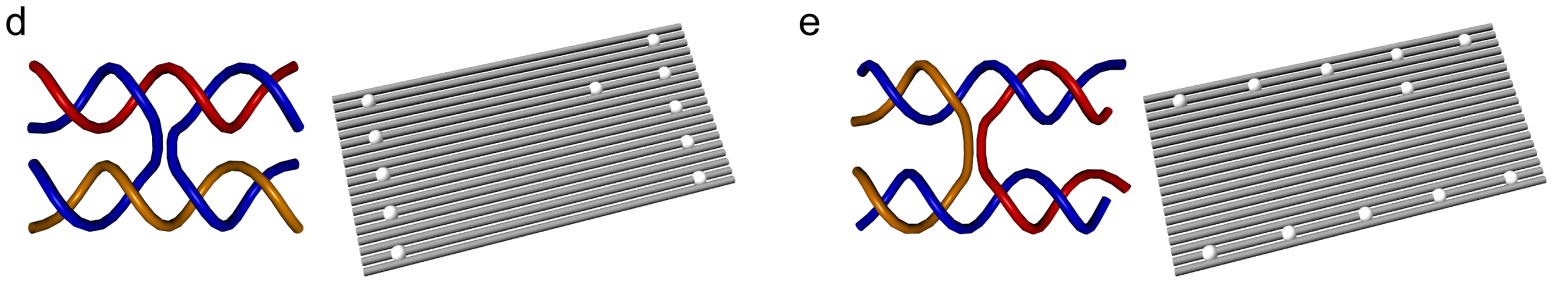

c
Staple Stacked
a
b
a'
b'
d'
c'
d
c


d
angle = 0°
angle = 180°

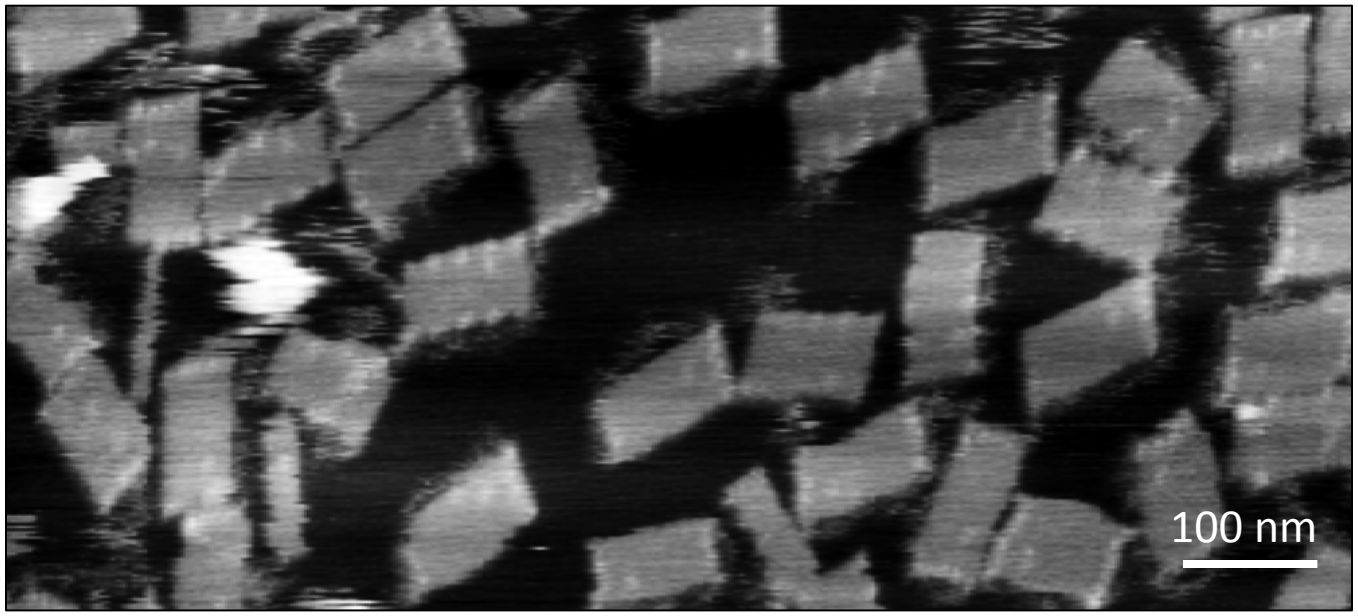

e
angle = 0°
angle = 180°

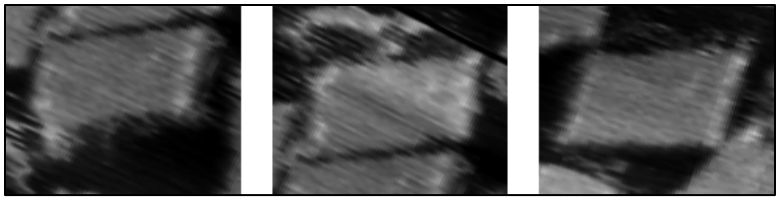

a
32
16
32
16
32
32


b
Stacked Scaffold
4
5
6
7
8
9
c
Stacked Staple
4
5
6
7
8
9


d
e

Results



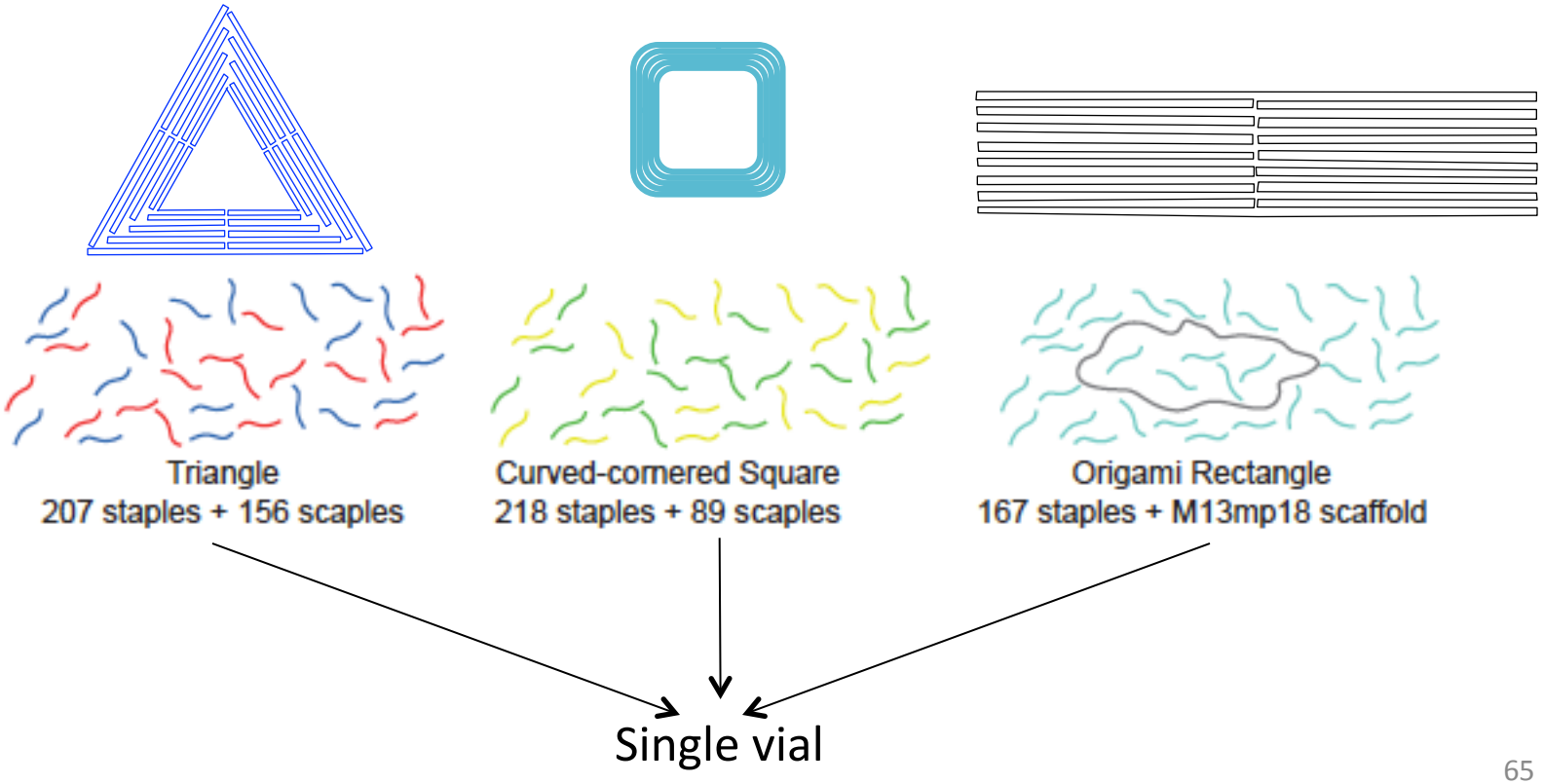


Stacked Scaffold

Stacked Staple

Structures with random sequence + “one pot” assembly

- Create structures with random sequences
- Assemble multiple structures in a single reaction



It Can Be Done!

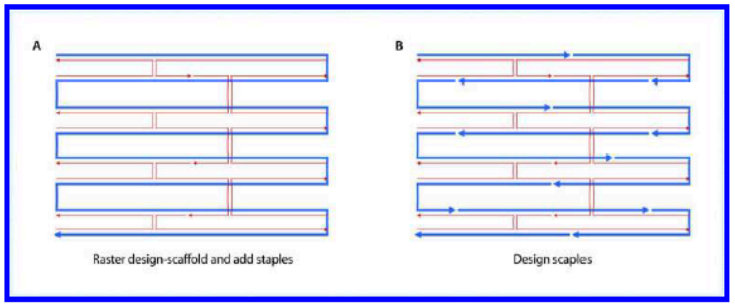
ACS SyntheticBiology

Letter

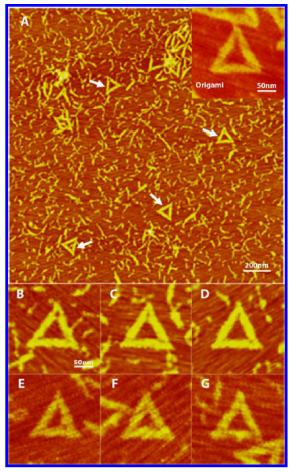
Complex DNA Nanostructures from Oligonucleotide Ensembles

Divita Mathur, and Eric R Henderson

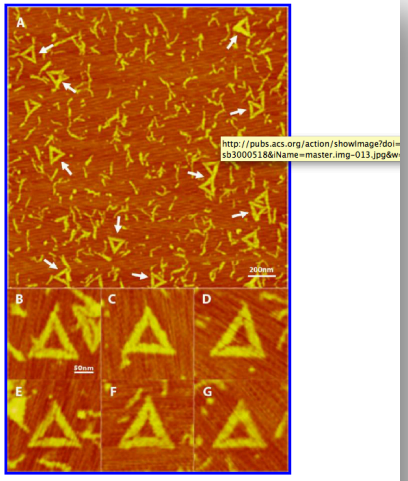
ACS Synth. Biol., **Just Accepted Manuscript** • DOI: 10.1021/sb3000518 • Publication Date (Web): 24 Jul 2012



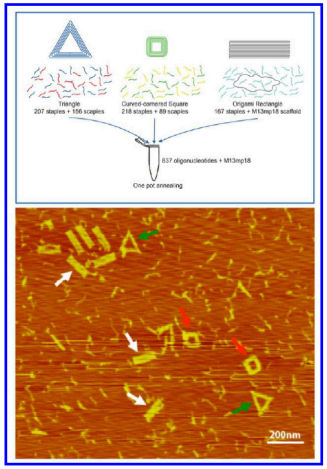
Strategy for the design of scaples-based nanostructures. (A) The first step is to raster a "design-scaffold" through the desired shape. Staples are then introduces using software (e.g., caDNAno (22)) or by hand. (B) In the key step, positions for the insertion of breakpoints on the design-scaffold are determined. The scaples and staples thus generated are synthesized and annealed as described here and in the Supporting Information.
177x65mm (300 x 300 DPI)



Scaples-based triangle using M13mp18 as the design-scaffold. (A) A representative field of the scaples version of the original DNA Origami triangle (shown for comparison in the inset). The 154 scaples created for this triangle were designed using the M13mp18 design-scaffold layout exactly as in the origami triangle (8). (B to G) Higher magnification AFM images of individual scaples-based triangles.
82x136mm (300 x 300 DPI)

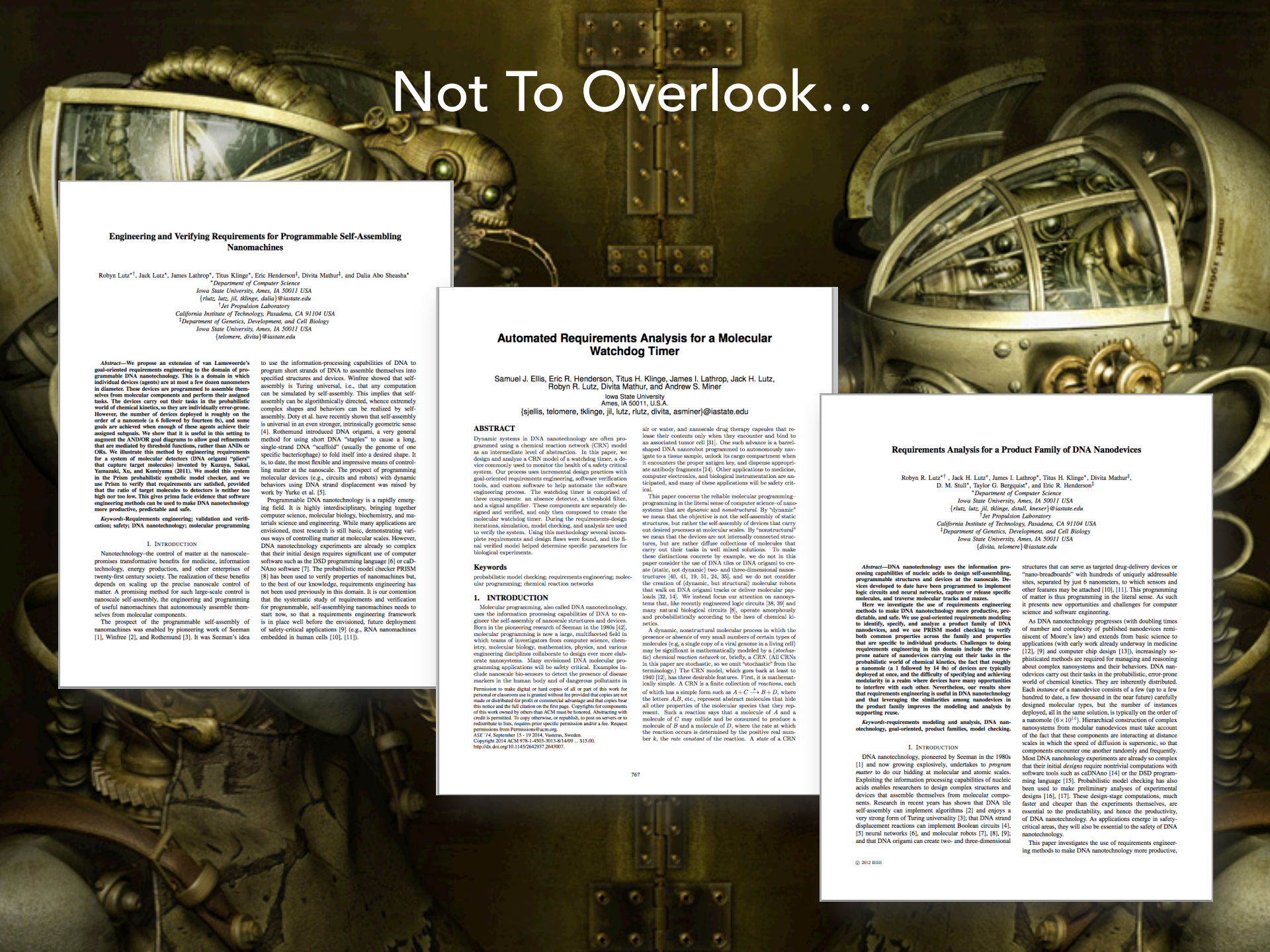


Scaples-based triangles constructed using a random sequence design-scaffold. The nanostructures shown are geometrically identical to those shown in Figure 2 but were created using a non-biological, random sequence design-scaffold. The sequence was processed to remove internal subsequence similarity, undesired internal complementarity and sequences formally capable of forming G-quartets. (A) A representative field AFM image of the triangles. (B to G) Individual examples of the same structure.
82x136mm (300 x 300 DPI)



Simultaneous assembly of three DNA nanostructures in a "one-pot" reaction. In the experiment shown here one DNA origami structure with an M13mp18 scaffold (rectangle, identified by white arrows) and two scaples-based nanostructures, a triangle (green arrows) and a square with curved corners (red arrows), were assembled in a single reaction containing over 800 distinct oligonucleotides. Both the triangle and the round-cornered square were designed with non-biological random sequences.
177x259mm (300 x 300 DPI)

Not To Overlook...



Engineering and Verifying Requirements for Programmable Self-Assembling Nanomachines

Robyn Lutz*†, Jack Lutz*, James Lathrop*, Titus Klinge*, Eric Henderson‡, Divita Mathur‡, and Dalia Abo Sheasha*
*Department of Computer Science
Iowa State University, Ames, IA 50011 USA
{rlutz, lutz, jil, tklinge, dalia}@iastate.edu
†Jet Propulsion Laboratory
California Institute of Technology, Pasadena, CA 91104 USA
‡Department of Genetics, Development, and Cell Biology
Iowa State University, Ames, IA 50011 USA
{telomere, divita}@iastate.edu

***Abstract*—We propose an extension of van Lamsweerde's goal-oriented requirements engineering to the domain of programmable DNA nanotechnology. This is a domain in which individual devices (agents) are at most a few dozen nanometers in diameter. These devices are programmed to assemble themselves from molecular components and perform their assigned tasks. The devices carry out their tasks in the probabilistic world of chemical kinetics, so they are individually error-prone. However, the number of devices deployed is roughly on the order of a nanomole (a 6 followed by fourteen 0s), and some goals are achieved when enough of these agents achieve their assigned subgoals. We show that it is useful in this setting to augment the AND/OR goal diagrams to allow goal refinements that are mediated by threshold functions, rather than ANDs or ORs. We illustrate this method by engineering requirements for a system of molecular detectors (DNA origami "pliers" that capture target molecules) invented by Kuzuya, Sakai, Yamazaki, Xu, and Komiyama (2011). We model this system in the Prism probabilistic symbolic model checker, and we use Prism to verify that requirements are satisfied, provided that the ratio of target molecules to detectors is neither too high nor too low. This gives prima facie evidence that software engineering methods can be used to make DNA nanotechnology more productive, predictable and safe.**

***Keywords*-Requirements engineering; validation and verification; safety; DNA nanotechnology; molecular programming**

I. Introduction

Nanotechnology–the control of matter at the nanoscale–promises transformative benefits for medicine, information technology, energy production, and other enterprises of twenty-first century society. The realization of these benefits depends on scaling up the precise nanoscale control of matter. A promising method for such large-scale control is nanoscale self-assembly, the engineering and programming of useful nanomachines that autonomously assemble themselves from molecular components.

The prospect of the programmable self-assembly of nanomachines was enabled by pioneering work of Seeman [1], Winfree [2], and Rothemund [3]. It was Seeman's idea to use the information-processing capabilities of DNA to program short strands of DNA to assemble themselves into specified structures and devices. Winfree showed that self-assembly is Turing universal, i.e., that any computation can be simulated by self-assembly. This implies that self-assembly can be algorithmically directed, whence extremely complex shapes and behaviors can be realized by self-assembly. Doty et al. have recently shown that self-assembly is universal in an even stronger, intrinsically geometric sense [4]. Rothemund introduced DNA origami, a very general method for using short DNA "staples" to cause a long, single-strand DNA "scaffold" (usually the genome of one specific bacteriophage) to fold itself into a desired shape. It is, to date, the most flexible and impressive means of controlling matter at the nanoscale. The prospect of programming molecular devices (e.g., circuits and robots) with dynamic behaviors using DNA strand displacement was raised by work by Yurke et al. [5].

Programmable DNA nanotechnology is a rapidly emerging field. It is highly interdisciplinary, bringing together computer science, molecular biology, biochemistry, and materials science and engineering. While many applications are envisioned, most research is still basic, demonstrating various ways of controlling matter at molecular scales. However, DNA nanotechnology experiments are already so complex that their initial design requires significant use of computer software such as the DSD programming language [6] or caDNAno software [7]. The probabilistic model checker PRISM [8] has been used to verify properties of nanomachines but, to the best of our knowledge, requirements engineering has not been used previously in this domain. It is our contention that the systematic study of requirements and verification for programmable, self-assemblying nanomachines needs to start now, so that a requirements engineering framework is in place well before the envisioned, future deployment of safety-critical applications [9] (e.g., RNA nanomachines embedded in human cells [10], [11]).

Automated Requirements Analysis for a Molecular Watchdog Timer

Samuel J. Ellis, Eric R. Henderson, Titus H. Klinge, James I. Lathrop, Jack H. Lutz, Robyn R. Lutz, Divita Mathur, and Andrew S. Miner
Iowa State University
Ames, IA 50011, U.S.A.
{sjellis, telomere, tklinge, jil, lutz, rlutz, divita, asminer}@iastate.edu

ABSTRACT

Dynamic systems in DNA nanotechnology are often programmed using a chemical reaction network (CRN) model as an intermediate level of abstraction. In this paper, we design and analyze a CRN model of a watchdog timer, a device commonly used to monitor the health of a safety critical system. Our process uses incremental design practices with goal-oriented requirements engineering, software verification tools, and custom software to help automate the software engineering process. The watchdog timer is comprised of three components: an absence detector, a threshold filter, and a signal amplifier. These components are separately designed and verified, and only then composed to create the molecular watchdog timer. During the requirements-design iterations, simulation, model checking, and analysis are used to verify the system. Using this methodology several incomplete requirements and design flaws were found, and the final verified model helped determine specific parameters for biological experiments.

Keywords

probabilistic model checking; requirements engineering; molecular programming; chemical reaction networks

1. INTRODUCTION

Molecular programming, also called DNA nanotechnology, uses the information processing capabilities of DNA to engineer the self-assembly of nanoscale structures and devices. Born in the pioneering research of Seeman in the 1980s [42], molecular programming is now a large, multifaceted field in which teams of investigators from computer science, chemistry, molecular biology, mathematics, physics, and various engineering disciplines collaborate to design ever more elaborate nanosystems. Many envisioned DNA molecular programming applications will be safety critical. Examples include nanoscale bio-sensors to detect the presence of disease markers in the human body and of dangerous pollutants in air or water, and nanoscale drug therapy capsules that release their contents only when they encounter and bind to an associated tumor cell [31]. One such advance is a barrel-shaped DNA nanorobot programmed to autonomously navigate to a tissue sample, unlock its cargo compartment when it encounters the proper antigen key, and dispense appropriate antibody fragments [14]. Other applications to medicine, computer electronics, and biological instrumentation are anticipated, and many of these applications will be safety critical.

This paper concerns the reliable molecular programming–programming in the literal sense of computer science–of nanosystems that are *dynamic* and *nonstructural*. By "dynamic" we mean that the objective is not the self-assembly of static structures, but rather the self-assembly of devices that carry out desired *processes* at molecular scales. By "nonstructural" we mean that the devices are not internally connected structures, but are rather diffuse collections of molecules that carry out their tasks in well mixed solutions. To make these distinctions concrete by example, we do not in this paper consider the use of DNA tiles or DNA origami to create (static, not dynamic) two- and three-dimensional nanostructures [40, 41, 19, 51, 24, 35], and we do not consider the creation of (dynamic, but structural) molecular robots that walk on DNA origami tracks or deliver molecular payloads [32, 14]. We instead focus our attention on nanosystems that, like recently engineered logic circuits [38, 39] and many natural biological circuits [8], operate amorphously and probabilistically according to the laws of chemical kinetics.

A dynamic, nonstructural molecular process in which the presence or absence of very small numbers of certain types of molecules (e.g, a single copy of a viral genome in a living cell) may be significant is mathematically modeled by a *(stochastic) chemical reaction network* or, briefly, a *CRN*. (All CRNs in this paper are stochastic, so we omit "stochastic" from the terminology.) The CRN model, which goes back at least to 1940 [12], has three desirable features. First, it is mathematically simple. A CRN is a finite collection of *reactions*, each of which has a simple form such as $A + C \xrightarrow{k} B + D$, where the letters A, B, etc., represent abstract molecules that hide all other properties of the molecular species that they represent. Such a reaction says that a molecule of A and a molecule of C may collide and be consumed to produce a molecule of B and a molecule of D, where the rate at which the reaction occurs is determined by the positive real number k, the *rate constant* of the reaction. A *state* of a CRN

ASE '14, September 15 - 19 2014, Vasteras, Sweden.

http://dx.doi.org/10.1145/2642937.2643007.

Requirements Analysis for a Product Family of DNA Nanodevices

Robyn R. Lutz*† , Jack H. Lutz*, James I. Lathrop*, Titus H. Klinge*, Divita Mathur‡, D. M. Stull*, Taylor G. Bergquist*, and Eric R. Henderson‡
*Department of Computer Science
Iowa State University, Ames, IA 50011 USA
{rlutz, lutz, jil, tklinge, dstull, bnexer}@iastate.edu
†Jet Propulsion Laboratory
California Institute of Technology, Pasadena, CA 91104 USA
‡Department of Genetics, Development, and Cell Biology
Iowa State University, Ames, IA 50011 USA
{divita, telomere}@iastate.edu

***Abstract*—DNA nanotechnology uses the information processing capabilities of nucleic acids to design self-assembling, programmable structures and devices at the nanoscale. Devices developed to date have been programmed to implement logic circuits and neural networks, capture or release specific molecules, and traverse molecular tracks and mazes.**

Here we investigate the use of requirements engineering methods to make DNA nanotechnology more productive, predictable, and safe. We use goal-oriented requirements modeling to identify, specify, and analyze a product family of DNA nanodevices, and we use PRISM model checking to verify both common properties across the family and properties that are specific to individual products. Challenges to doing requirements engineering in this domain include the error-prone nature of nanodevices carrying out their tasks in the probabilistic world of chemical kinetics, the fact that roughly a nanomole (a 1 followed by 14 0s) of devices are typically deployed at once, and the difficulty of specifying and achieving modularity in a realm where devices have many opportunities to interfere with each other. Nevertheless, our results show that requirements engineering is useful in DNA nanotechnology and that leveraging the similarities among nanodevices in the product family improves the modeling and analysis by supporting reuse.

***Keywords*-requirements modeling and analysis, DNA nanotechnology, goal-oriented, product families, model checking.**

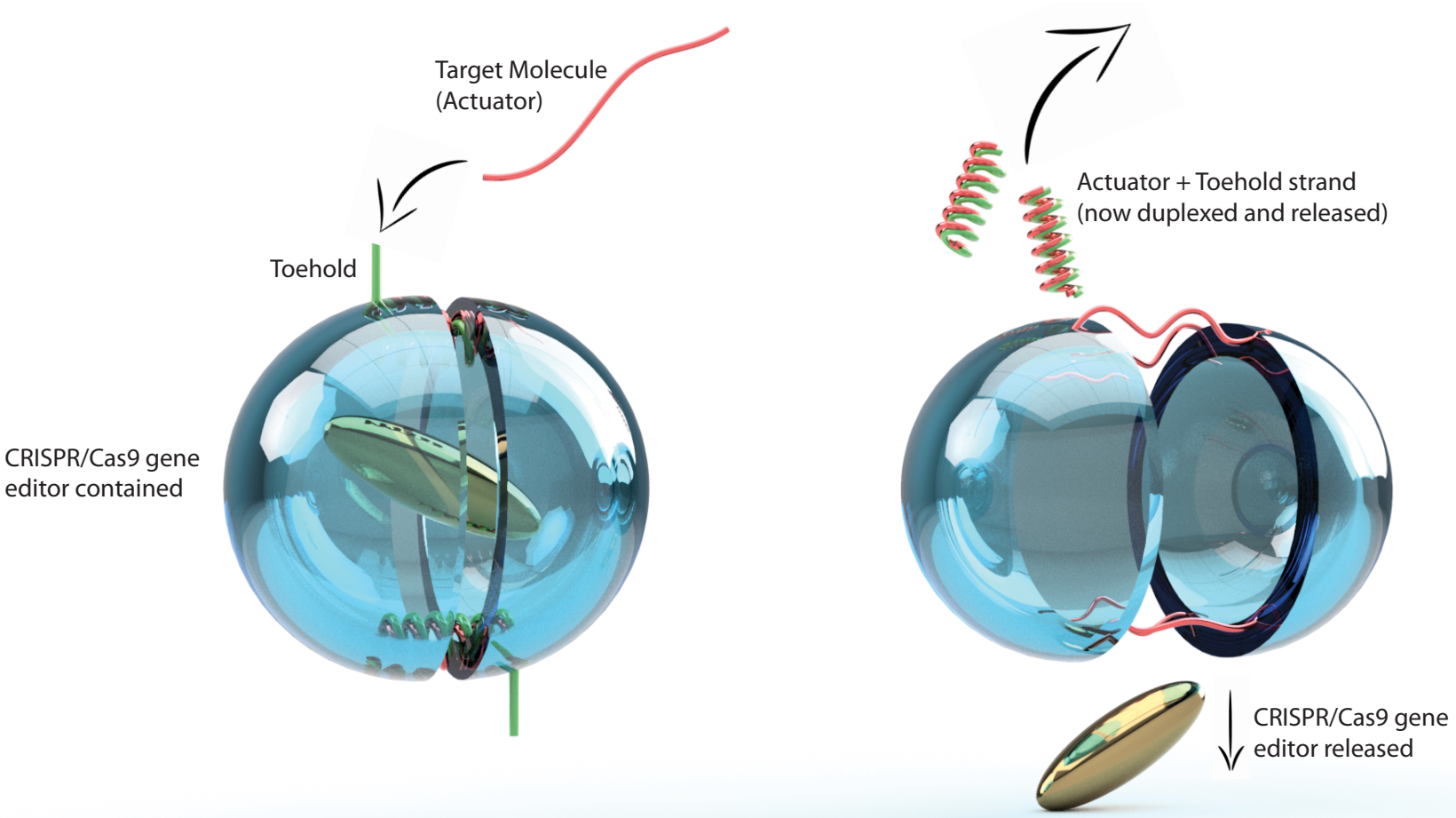
I. Introduction

DNA nanotechnology, pioneered by Seeman in the 1980s [1] and now growing explosively, undertakes to *program matter* to do our bidding at molecular and atomic scales. Exploiting the information processing capabilities of nucleic acids enables researchers to design complex structures and devices that assemble themselves from molecular components. Research in recent years has shown that DNA tile self-assembly can implement algorithms [2] and enjoys a very strong form of Turing universality [3]; that DNA strand displacement reactions can implement Boolean circuits [4], [5] neural networks [6], and molecular robots [7], [8], [9]; and that DNA origami can create two- and three-dimensional structures that can serve as targeted drug-delivery devices or "nano-breadboards" with hundreds of uniquely addressable sites, separated by just 6 nanometers, to which sensors and other features may be attached [10], [11]. This programming of matter is thus programming in the literal sense. As such it presents new opportunities and challenges for computer science and software engineering.

As DNA nanotechnology progresses (with doubling times of number and complexity of published nanodevices reminiscent of Moore's law) and extends from basic science to applications (with early work already underway in medicine [12], [9] and computer chip design [13]), increasingly sophisticated methods are required for managing and reasoning about complex nanosystems and their behaviors. DNA nanodevices carry out their tasks in the probabilistic, error-prone world of chemical kinetics. They are inherently distributed. Each *instance* of a nanodevice consists of a few (up to a few hundred to date, a few thousand in the near future) carefully designed molecular types, but the number of instances deployed, all in the same solution, is typically on the order of a nanomole (6×10^{13}). Hierarchical construction of complex nanosystems from modular nanodevices must take account of the fact that these components are interacting at distance scales in which the speed of diffusion is supersonic, so that components encounter one another randomly and frequently. Most DNA nanotechnology experiments are already so complex that their initial *designs* require nontrivial computations with software tools such as caDNAno [14] or the DSD programming language [15]. Probabilistic model checking has also been used to make preliminary analyses of experimental designs [16], [17]. These design-stage computations, much faster and cheaper than the experiments themselves, are essential to the predictability, and hence the productivity, of DNA nanotechnology. As applications emerge in safety-critical areas, they will also be essential to the safety of DNA nanotechnology.

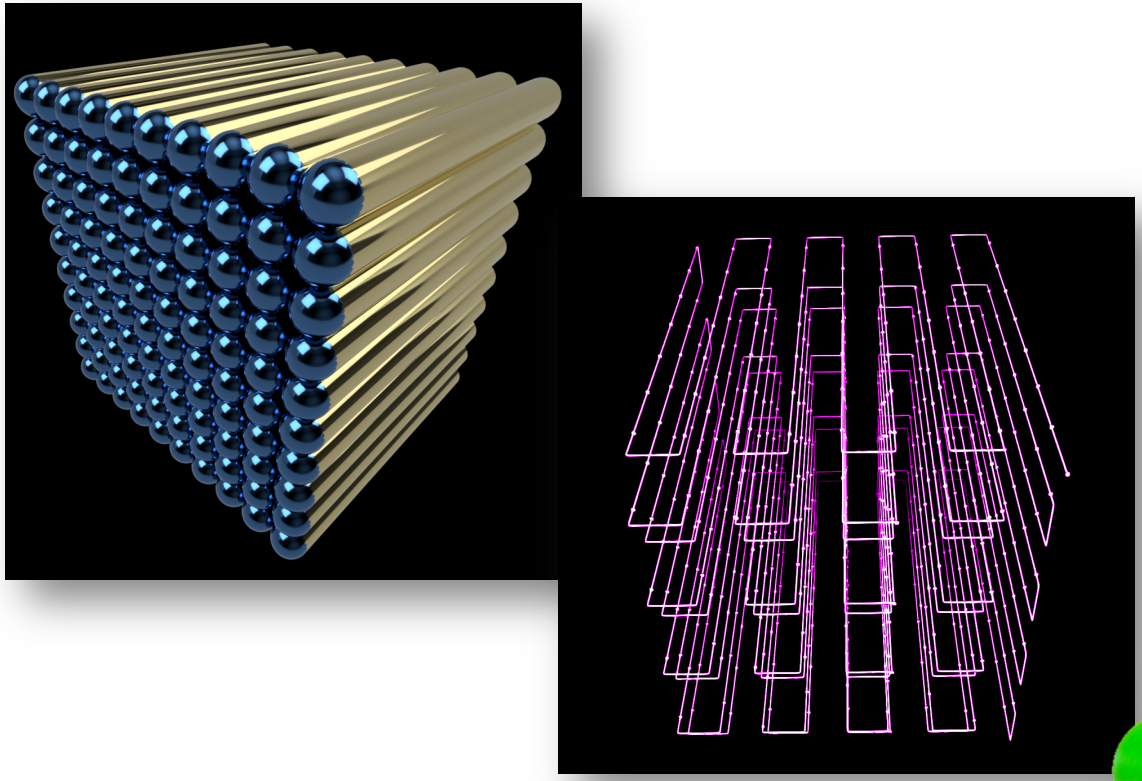
This paper investigates the use of requirements engineering methods to make DNA nanotechnology more productive,

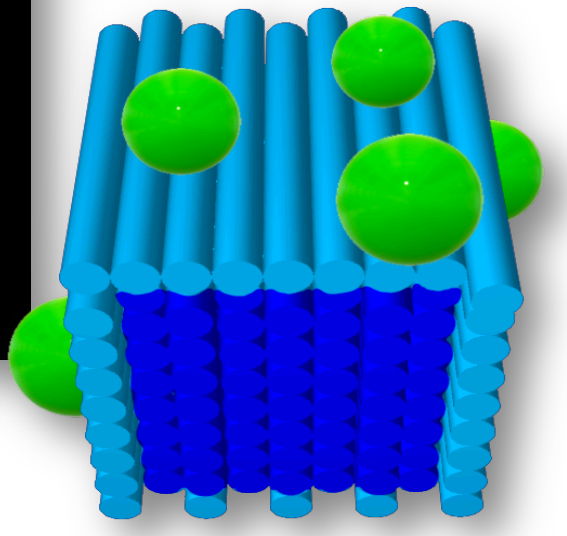
Next Goal: Eradicate HIV (HSV, HPV)



Wait, it's all DNA so why not…

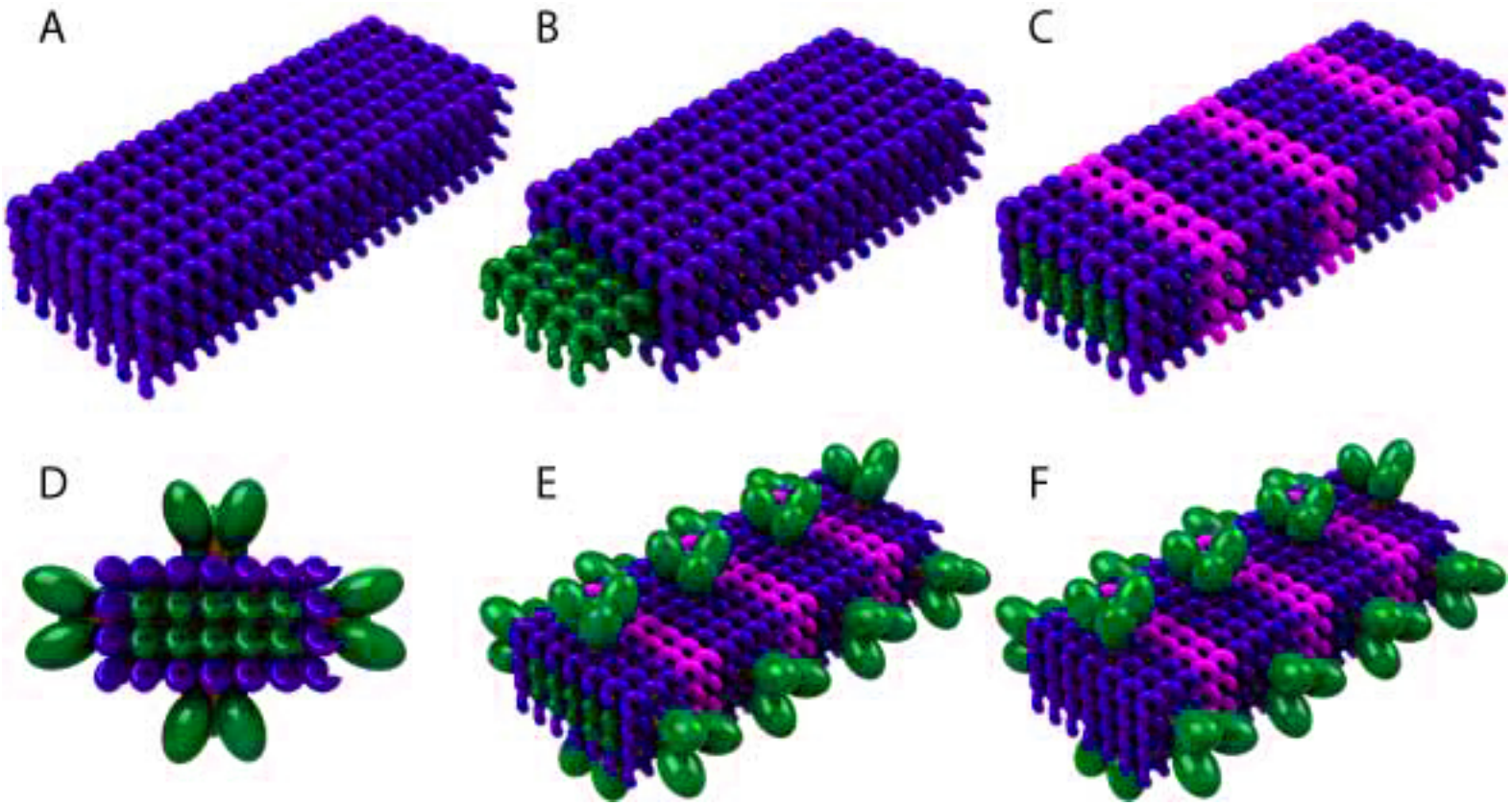
CamoNano™



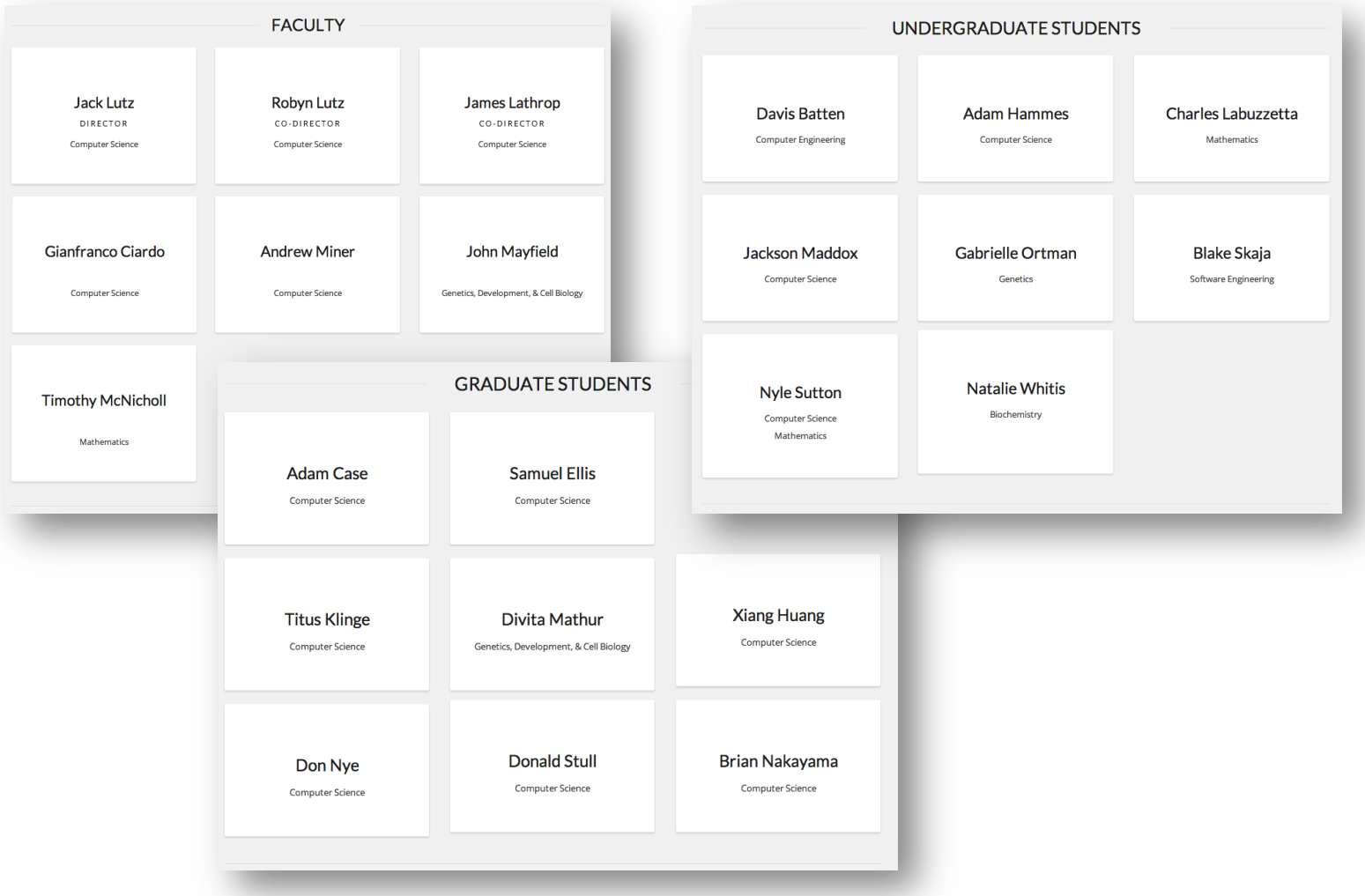


The nanobot IS the treatment!

CamoNano™



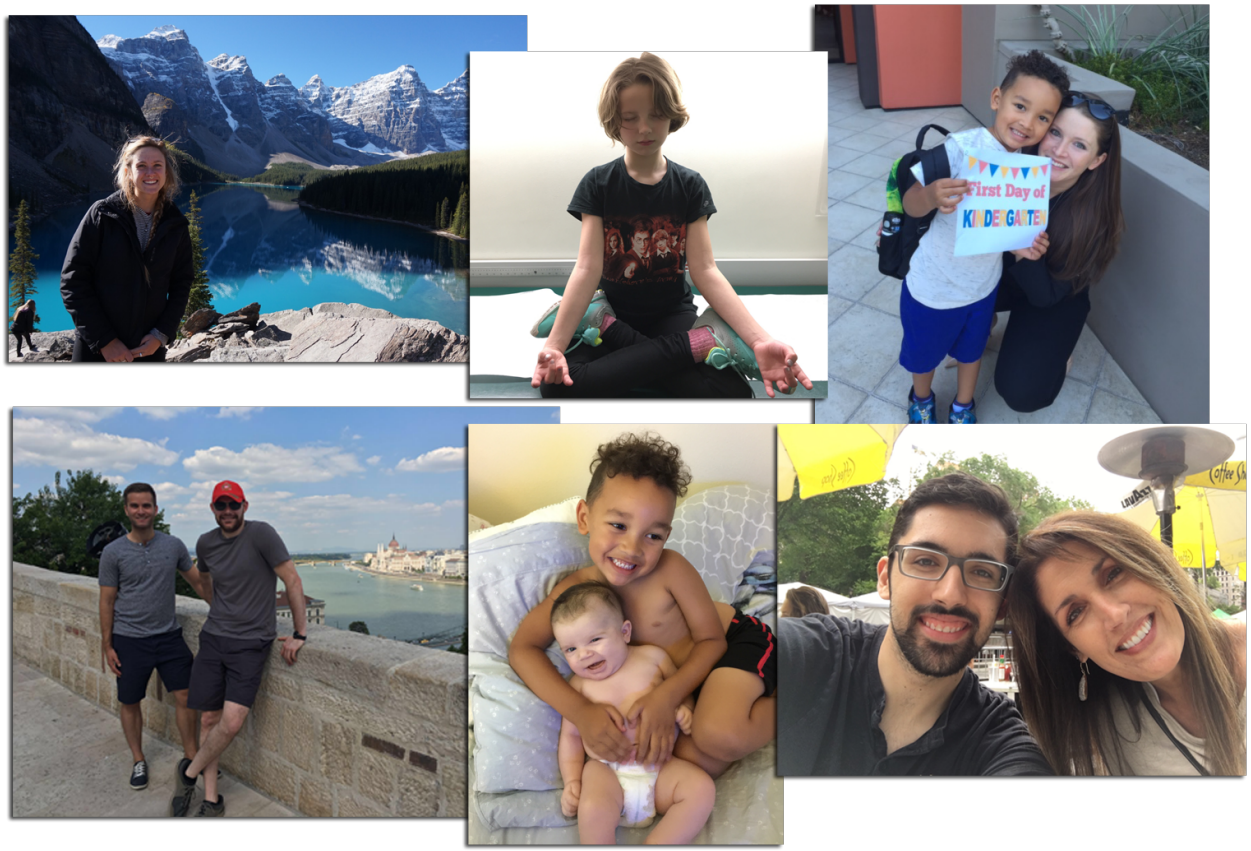
Heroes Include: Dr. Divita Mathur and…

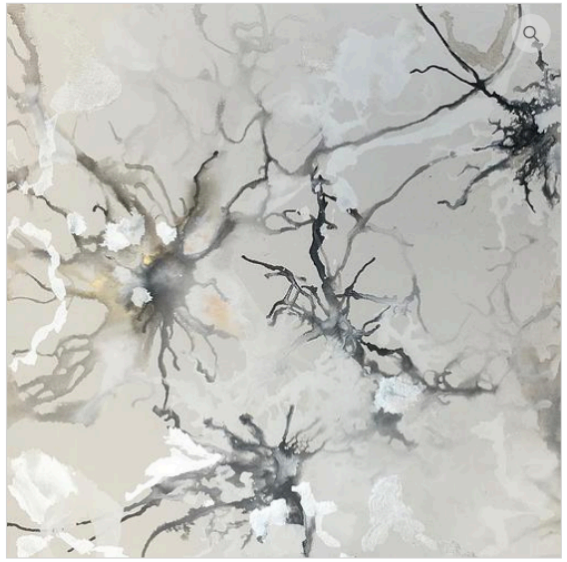


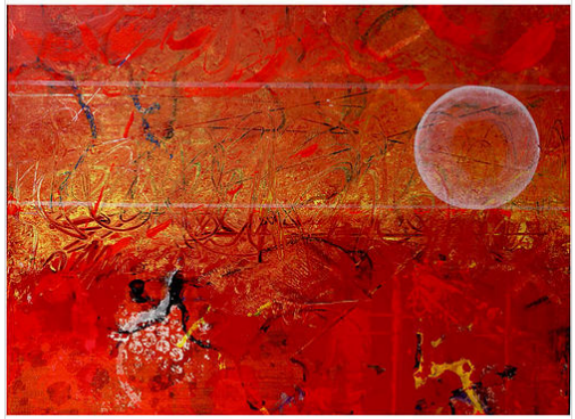
LAMP (http://www.cs.iastate.edu/~lamp/)
Henderson Lab (http://www.public.iastate.edu/~telomere/)

NSF
Chase Bank (VISA)/Gates Foundation

Be Fearless
Pursue Your Passion
Leave the World Better Than You Found It







Studiomjd.com