Computational Analysis & Design of Biopolymers

by

Jonah Procyk

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

Approved November 2022 by the
Graduate Supervisory Committee:

Petr Šulc, Co-Chair
Nicholas Stephanopoulos, Co-Chair
Rizal Hariadi
Matthias Heyden

ARIZONA STATE UNIVERSITY

December 2022

ABSTRACT

Biopolymers perform the majority of essential functions necessary for life. From a small amount of components emerges considerable complexity in both structure and function. The separated timescales of dynamic processes and intricate intra- and inter-molecular interactions of these molecules necessitate the development and utilization of computational approaches for biopolymer study and nanotechnology applications. Biopolymer nanotechnology exploits the natural chemistry of biopolymers to perform novel functions at the nanoscale. Molecular dynamics is the numerical simulation of chemical entities according to the physical laws of motion and statistical mechanics. The number of atoms in biopolymers require coarse-grained methods to fully sample the dynamics of the system with reasonable resources. Accordingly, a coarse-grained molecular dynamics model for the characterization of hybrid nucleic acid-protein nanotechnology was developed. Proteins are represented as an anisotropic network model (ANM) which show good agreement with experimentally derived protein dynamics for a small computational cost. The model was subsequently applied to hybrid DNA-protein cages systems and exhibited excellent agreement with experimental results. Ongoing development efforts look to apply network models to oxDNA origami to create multiscale models for DNA origami. The network approximation will allow for detailed simulation of DNA origami association, of concern to DNA crystal and lattice formation.

Identification and design of target-specific binders (aptamers) has received considerable attention on account of their diagnostic and therapeutic potential. Generated in selection cycles from extensive random libraries, biopolymer aptamers are of particular interest due to their potential non-immunogenic properties. Machine learning leverages the use of powerful statistical principles to train a model to transform an input into a desired output. Parameters of the model are iteratively adjusted according to the gradient of the cost function. An unsupervised and generative machine learning model was applied to Thrombin aptamer sequence data. From the model, sequence characteristics necessary for binding were identified and new aptamers capable of binding Thrombin were sampled and verified experimentally.

Future work on the development and utilization of an unsupervised and interpretable machine learning model for unaligned sequence data is also discussed.

DEDICATION

Dedicated to Ally and Amber. My absolute favorites.

ACKNOWLEDGMENTS

Earning my PhD has been an experience. Due to the nature of my research, the learning curve was quite steep at first and while it never plateaued, it provided an ideal learning environment for me: one where there was always more to learn.

This work represents my long journey through the challenges of research, academia, and software development. Thankfully through it all I had the support of my wife, my family, and my friends. They will all be addressing me as Dr. Jonah for the next six months at a minimum.

My advisors Petr Sulc and Nicholas Stephanopoulos have helped me considerably through this ordeal. Who else would bravely put their name on something I had a hand in writing? In all seriousness, I appreciate everything Petr and Nick did for me. I don't believe I would have succeeded to the degree that I have without advisors that supported my ideas and allowed me to pursue my own interests as much as Petr and Nick have. I first met Nick when I was an undergraduate research volunteer in his lab who in turn introduced me to Petr and his computational work with DNA nanotechnology. Nick's excitement for nanotechnology and desire for us to succeed was my favorite part about being a member of his lab. One of my favorite memories of Petr was his attempt to debug my first iteration of the anm-oxDNA model. I can't repeat what he said here, but to my keyboard's horror I adopted a similar attitude when debugging software. One thing I'll always remember about both is the excruciatingly long group meetings.

My labmates in the Sulc lab were a constant source of reliable advice and inspiration. In particular I want to thank Micha, Hao, Erik, Matthew, Joshua, Lanshen, Thong, and Subhajit for being great labmates. I also want to thank the current and past Stephanopoulos lab members, in particular Tara, Alex, Raghu, Yang, Minghui, Julio, Al-Amin, Skylar, Alexandra, Paul, Abhay, Rong, Xinyi, and Charitha for their motivating research and feigned interest in my work.

I'd like to thank my generous benefactor who made existing in Arizona with below poverty wages possible. Your support can not be understated, and I appreciate everything you have done for me. Thanks Mom!

Finally, a special thank you to Amber for her unwavering support in all stages of my education. Though no longer with us, she should be considered a ghost author on all of my publications, including this work.

As it is in bad taste to have an unacknowlegement section, I will refrain from mentioning ASU Parking and Transit Services.

TABLE OF CONTENTS

Page

LIST OF TABLES

Table Page

Table Page

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

Chapter 1

INTRODUCTION

1.1 The Nanoscale

The universe as we currently know it has at its fundamental building blocks bosons and fermions. The full expanse of the universe, estimated to be 93 billion light years, $8.79 * 10^{26}$ m wide, consists of matter, which is built from fermion components approximately 10^{-15}m in size, and energy. Much as size exists across this impressive scale so does time; the age of the universe is approximately 13.7 billion years ($4.35 * 10^{17}$ s) while electron transfer occurs at the attosecond ($1 * 10^{-18}$ s) scale. Across these extremes of time and spatial dimensions are all processes of the universe: planet formation, magnetic reversals, and star life cycles. More pertinent to our daily lives are the processes of the natural world: organism life spans and cellular life cycles.

The first discoveries by humankind centered around time and length scales immediately accessible and observable to us. Postulations of why objects fell, i.e. gravity, were recorded as early as 380 BC by Aristotle [1]. Observations of how the objects in the sky moved each year can be seen in ancient architecture such as Stonehenge [2]. Less obvious to humans are objects we cannot see or interact with. It was not until the creation of the microscope in the late 1600s, that the discovery of micro-organisms and cells first occurred [3]. Going even smaller, the first experimental evidence for the atom was not until the 1800s, with the development of quantum mechanics not beginning until the early 1900s [4, 5].

Continuing to today, much progress has been made at the nanoscale (or smaller) in a diverse array of fields: particle physics with the standard model, electronics with nanometer precision of semiconductor placement, and immunology with the development of vaccines. Despite these advances there is still much we do not understand, particularly in the field of biology. There are fundamental challenges that arise from study of nanoscale biological systems and their components. These include:

1. **We cannot directly see nanoscale systems**, so a vast array of experimental assays have been developed to retrieve information on these systems. Popular methods to visualize nanoscale systems directly include methods such as cryogenic electron microscopy (cryo-EM) [6], DNA-PAINT [7], Atomic Force Microscopy (AFM) [8], super-resolution Microscopy [9], transmission electron Microscopy (TEM) [10], scanning electron Microscopy (SEM) [11], nuclear magnetic resonance (NMR) [12], small-angle X-ray scattering (SAXS) [13], and X-ray crystal diffraction (XRD) [14]. Each assay only gives a piece of information about the system usually averaged over a sample made up of many copies of the system of interest. The assay's preparation or procedure is often highly involved, and may also be destructive to the sample itself.
2. **The timescales of nanoscale system dynamics are typically extremely fast (**fs - μs**)**, with different processes within the same system being separated by orders of magnitude in time. Investigations of electron and atomistic dynamics require sophisticated techniques that can reach femtosecond timescales such as pump-probe spectroscopy [15] or X-ray free-electron lasers (XFEL) [16]. However, pump probe techniques require simple samples for feasible analysis and XFEL techniques require immense data processing capabilities and are destructive to the sample.
3. **Tracking individual species in a sample is not easy**. State of the art methods require labeling of the target species with fluorophore or other molecular trackers [17], but these methods can achieve less than micrometer resolution over second timescales.
4. **Probing the interactions (and their strengths) between different molecules cannot be done in cellular or other complex media**. To determine these interactions in biologically relevant milieus requires individual samples that are labeled (as in flow cytometry [18]) or label-free methods such as a native gel assay for relative binding strength

information. Exact determination of a ligand's binding affinity can be performed using expensive surface plasmon resonance (SPR) techniques [19].

5. **Our inability to predict the behavior of complex systems with no prior information makes research at the nanoscale difficult**. Known as the many-body problem, interactions of a system of particles at the quantum level scales exponentially with system size, resulting in a computationally intractable problem for large systems. The system can be approximated to the classical level of dynamics, an approach typically used by fully atomistic molecular dynamic methods. However, even fully atomistic methods still scale poorly with the system sizes of biopolymers [20].

Most cellular processes are heavily dependent on three key biopolymeric molecules: DNA, RNA, and proteins. Despite being almost entirely made up of a small set of monomers, these biopolymers are responsible for incredibly complex functions across all domains of life[21]. In *E.Coli*, estimates of cellular composition place protein as the most abundant of these three molecules at 55 % of the cell's dry weight, followed by RNA at 20.5 % and DNA at 3 % [22].Taken together, DNA, RNA, and proteins account for 80 % of the cell's dry weight and perform the majority of functions responsible for life. Understanding how the key components of cells interact is central to our understanding of the cell's mechanisms themselves.

By focusing on individual biopolymers in a test tube environment we can study and learn how these fundamental molecules function and interact. Further, we can use these biopolymers to perform novel functions. The collective fields within biomolecular nanotechnology aim to create functional materials, therapeutics, diagnostics, and assays from polypeptides, oligonucleotides, lipids, polysaccharides, cofactors, and small molecules [23].

1.1.1 Summary

In this work, we will focus on the development and application of computational approaches to biopolymer design and characterization using two main techniques, molecular dynamics and machine learning. Molecular dynamics is the numerical simulation of chemical entities according to the physical laws of motion and statistical mechanics. The large sizes of biopolymers require coarse-grained methods to fully sample the dynamics of the system of interest. Accordingly, we develop a coarse-grained molecular dynamics model for the characterization of hybrid nucleic acid-protein nanotechnology. We then apply the model to hybrid DNA-protein cages systems and compare the results with experimental data.

Machine learning leverages the use of powerful statistic principles to train a model to transform an input into a desired output. Parameters of the model are iteratively adjusted according to the gradient of the cost function. Here, we apply an unsupervised and generative machine learning model to Thrombin aptamer sequence data. From the model, we obtain sequence characteristics necessary for binding and generate new aptamers that are able to bind Thrombin.

Finally, I discuss ongoing research projects for coarse-grained molecular dynamics of large DNA origami and machine learning for aptamer datasets.

1.1.2 Biomolecular Nanotechnology

1.1.2.1 DNA Nanotechnology

DNA consists of four bases on a phosphate-deoxyribose sugar backbone–adenine (A), guanine (G), cytosine (C), and thymine (T)–which the molecule uses to encode the entire genome of all known living organisms [24]. Depictions of the chemical structure of DNA are shown in Figure 1b. In bacteria, almost all DNA codes for functional proteins; however, in large

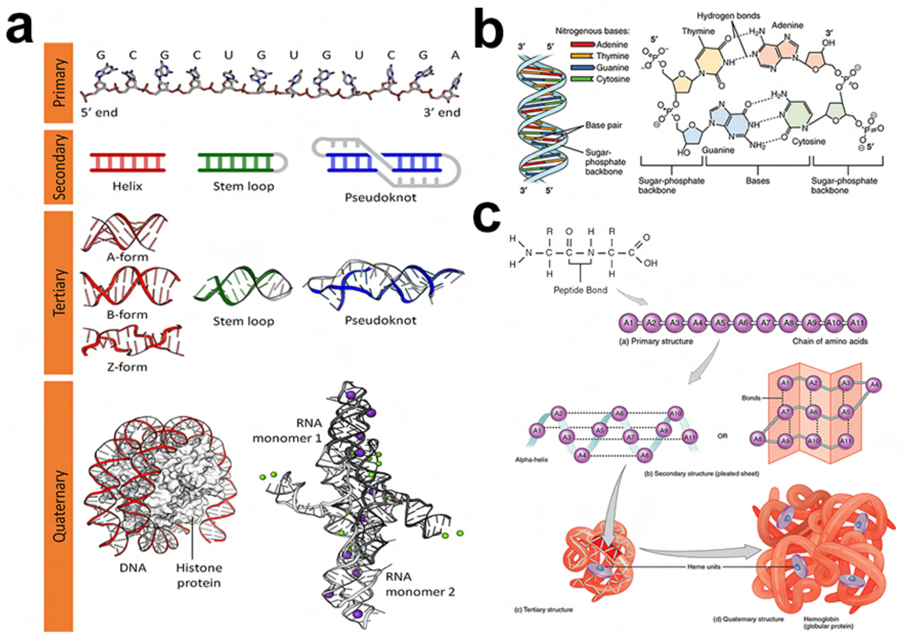


Figure 1. Chemical and higher order structure of (ab) DNA and RNA, and (c) Proteins. Images adapted under Creative Commons licenses, (a) by Thomas Shafee, CC BY 4.0, via Wikimedia Commons, (b) by OpenStax, CC BY 4.0, via Wikimedia Commons, and (c) consists of two images from OpenStax College, CC BY 3.0, via Wikimedia Commons.

multicellular organisms, a small percent of DNA actually codes for proteins while the rest is noncoding. While protein-coding DNA (the exome) makes up only 1% of the human genome, the true proportion of functional DNA is estimated to be around 15 % [25]. The noncoding 14 % of functional DNA has regulatory functions and produces non-coding RNA [26], which in turn can have a variety of different regulatory functions[27, 28].

DNA serves as the carrier for genetic information which is constantly translated, repaired, and replicated to maintain a cell's function. Bonds between complementary bases known as Watson-Crick base pairs (A-T, C-G) and the deoxyribose backbone give DNA the stability necessary to be a storage medium for cells.

Pioneered by Nadrian Seeman [29], DNA Nanotechnology is a field of research that uses the predictability of Watson-Crick base pairing in DNA to form user-defined nanoscale objects. While originally envisioned as a method for crystallizing protein structures, DNA Nanotechnology has evolved to a broad range of uses, including the development of therapeutics, diagnostics, and molecular computing [30]. Early demonstrations included folding a long scaffold strand with short staple strands into arbitrary 2D shapes including letters of the alphabet [31]. Since then notable contributions to the field have included design of dynamic strand displacement circuits to implement basic logic gates as well as more complicated functions[32, 33, 34], a DNA nanorobot capable of killing tumorous cells [35], and low-cost diagnostics for virus detection [36]. DNA has also seen interest for its ability to encode and store data [37]. Even mechanical nanoscale motors have been recently developed using entirely DNA [38].

Central to DNA nanotechnology is the canonical DNA helix (termed "B-form" DNA), which is made up of watson-crick pairs governed by hydrogen bond formation between complementary bases. Base pairs are spaced approximately 3.4 Åapart. One full turn of the right-handed molecule corresponds to 10.5 base pairs. Neighboring base pairs also have sequence dependent stacking and coaxial stacking interactions which help to stabilize the double helix structure [39]. B-form DNA is by far the most dominant DNA tertiary structure in organisms as well as DNA nanotechnology applications.

Other forms of DNA tertiary structure also exist such as Z-form DNA, which occurs in G- and C-rich tracts of DNA. Other structural motifs such as G-motifs and i-motifs can also emerge in the right ionic conditions. These forms can be useful for select applications such as aptamers where the oligonucleotide binds a target, but are largely avoided in DNA origami formation. In nature, G-motifs are enriched in telomeres and gene promoter sequences [40]. Illustrations of A-form, Z-form, and B-form tertiary DNA structures can be seen in Figure 1a.

1.1.2.2 RNA Nanotechnology

Similar to DNA, RNA consists of four bases on a phosphate-ribose sugar backbone: adenine (A), guanine (G), cytosine(C), and uracil(U), which are used to encode proteins as well as perform a host of other functions including catalysis. The presence of RNA viruses does suggest that RNA-only organisms may have existed (or may even still exist today)[24]. Types of RNA within the cell are numerous with diverse functions. Basic examples include messenger RNA (mRNA), which is translated into a polypeptide sequence, transfer RNA (tRNA) is responsible for transferring amino acids to the polypeptide chain during protein synthesis, and ribosomal RNA (rRNA) a major contributor to the ribosomal complex [41].

RNA Nanotechnology has not come quite as far as DNA Nanotechnology despite it's promise of both programmability and expanded chemical function. Overall, RNA is a much less stable molecule than DNA, mostly due to the ribose sugar's additional hydroxyl group. The unpredictability of RNA tertiary structure–due to interactions outside the canonical A-U, C-G pairing e.g. Hoogsteen and wobble base pairs, as well as sugar-edge interactions–makes design of these structures significantly more difficult [42]. Further, experimental determination of RNA structure is challenging [43]. The trade-off to these difficulties, however, is the increased functionality that RNA is capable of relative to DNA. Therapeutic examples of RNA nanotechnology include small interfering RNA's (siRNA) that stop the expression of a target protein, and RNA aptamers which tightly bind a specific target [44, 45].

1.1.2.3 Protein Nanotechnology

Proteins primarily consist of the twenty naturally occurring amino acids. Synthesized from the genetic information carried by mRNA, proteins perform most of the important functions of the cell, including catalysis [46], movement [47], and energy production [48, 49]. Protein nanotechnology has seen considerable development and use because of the diverse chemical functionality proteins are capable of. Uniting different protein components to create self-assembled protein complexes has seen considerable interest as supramolecular protein complexes perform the most important and essential functions of the cell [50]. Creation of artificial protein complexes contributes to the exploration of how key features of the protein components and their interfaces (size, charge, shape, etc.) affect the resulting complex. While nature's protein complexes are exceedingly asymmetric and heterogeneous, designed complexes are most often symmetric and homogeneous. To date, nanomaterials including nanotubes, nanofibers, and nanoparticles have been constructed from the self-assembly of both proteins and peptides [51].

Often the modification of an existing protein-protein interaction allows for introduction of added functionality or other novel behavior into a naturally occurring protein. Azuma *et al.* review how the modification of the bacterial enzyme lumazine synthase has resulted in novel protein cages with potential uses in drug delivery and virus mimics [52]. Another important direction for the field is the use of existing protein motifs to design scaffolds. Lapenta *et al.* demonstrated that linked coiled coil protein motifs allow for the creation of user defined protein cages [53]. Rigid α-helical linkers have also been designed for fusion proteins [54].

Of particular importance is the design of specific protein-protein interactions. A recent comprehensive review of many protein-protein interaction algorithms found most to perform poorly on new data, having features reflecting spurious features of the training dataset, rather than features applicable to new data [55]. Clearly this is a complicated problem, albeit one that is getting more tractable as the amount of experimental data increases and the computational approaches to the problem get better. Novel proteins have been designed using deep learning methods [56, 57], though challenges still remain in designing proteins with new functions [58]. Steady progress is being made in these directions– *de novo* design of bioactive protein switches capable of large induced conformations with demonstrated regulatory applications[59] and the design of enzymes using a combination of experimental and computational methods [60, 61].

One of the most common applications is the design of antibodies due to their ability to recognize specific antigens. Monoclonal antibodies have seen considerable usage as therapeutics and diagnostics [62] including for SARS-CoV-2 treatments. Combinatorial approaches for monoclonal antibodies discovery include phage display which enriches antibody fragments with specific interactions in their complementary-determining region (CDR) to a given target from an initial random library. Optimization of the most enriched antibody fragments is essential as an antibody's efficacy is based not just on it's ability to specifically bind a target but also the antibody's folding stability and solubility[63]. Computational approaches primarily focus on improving binding in the CDR region by mutating unfavorable contacts and improving folding stability with predictions from known antibody structures [63].

1.1.2.4 Hybrid DNA-Protein Nanotechnology

Another approach to the design of functional nanoscale objects is the combination of one or more biopolymers. In particular hybrid DNA-protein nanotechnology has received a lot of attention due to its ability to combine the programmability of DNA with the functionality of proteins. Leveraging the benefits of both biopolymers has various potential and reported applications include multivalent binders, biomimetics, biocatalysis, and biomaterials [64]. In common applications, the protein or peptide is covalently linked to the DNA scaffold using chemistry specific small molecule linkers[64]. Despite this additional difficulty in their construction, hybrid materials have demonstrated their potential in various applications including the synthesis of size-tunable DNA-protein cages [65] and cancer targeting hybrid nanorobots [35].

1.2 Computational Tools for Biomacromolecular Design

1.2.1 Design Tools

Designing nanostructures is not a trivial task. Nucleic acid origami designs require intricate routing and design of short staple strands to fold the longer scaffold sequence into the desired structure. As the design complexity and size grow, the problem becomes unmanageable to solve by hand. Further, changes to an existing structure would require re-routing the entire structure, an unenviable task to perform manually. To remedy this, numerous programs for designing DNA and RNA origami have been developed including Adenita [66]

MagicDNA[67], CaDNAno[68], Tiamat[69], sCaDNAno [70], oxView [71], and the upcoming ENSnano for curved DNA origami [72]. Collectively these tools have aided the design of the vast majority of published DNA and RNA nanostructures. Each program has their niche use, with MagicDNA, Tiamat, and CaDNAno being used for larger origami designs. Both oxView and Adenita allow more flexibility as they are free form editors. This means that elements such as loops and complicated junctions can be added easily; however, larger edits are more difficult.

However, designing a DNA nanostructure in one of the aforementioned programs does not mean it will function as intended. Simple single-layer DNA sheets, which are commonly represented as being planar, are actually highly dynamic structures that constantly bend and twist. If this flexibility is not accounted for, an experiment requiring exact positioning of other moieties on the sheet–such as fluorophore/quencher pairs or specific protein ligands for signaling cells–will be doomed from the beginning. This limitation applies to a majority of DNA nanostructures, as they are commonly designed as a single static structure when they instead exist as an ensemble of structures, dictated by the underlying principles of statistical mechanics.

Protein nanostructures require yet more specialized tools as the interactions between protein chains is significantly more difficult to predict, due to their large amount of weak intramolecular interactions like hydrogen bonds, disulfide bridges, van der Waals forces, and multi-body electrostatics. The most prevalent protein design tool is Rosetta and it's add-ons which enabled the design of protein icosahedral structures [73] and tightly binding mini-binders of the SARS-CoV-2 spike protein based off the ACE2 helix interaction with the spike receptor binder domain (RBD) [74].

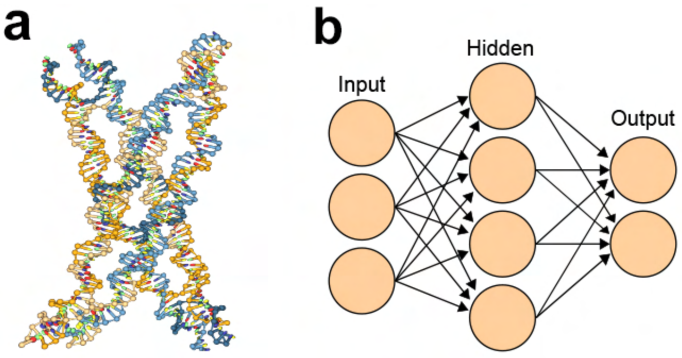


Figure 2. (a) oxDNA representation of a 45 degree angle layered crossover from the Nanobase Repository [75], originally published in [76]. (b) Basic neural network topology. Image used under a Creative Commons Licence: Cburnett, CC BY-SA 3.0, via Wikimedia Commons

1.2.2 Molecular Dynamics

Vital to numerous industries, molecular dynamics (MD) has granted key insight into the processes of the nanoscale. Molecular dynamics is founded on the rigorous theory of statistical mechanics, which relates variables at the microscopic scale (typically described by the position and momenta of individual particles) to properties at the macroscale: volume, pressure, temperature, etc. Unlike Netwonian or quantum mechanics, statistical mechanics considers multiple copies of the system, and includes uncertainty as to which state the system is in. Probabilities are assigned to each possible state the system can visit, where all states with equal energies are equally probable.

Fixing certain extrinsic variables allows for an ensemble of the system to be well defined. Most important to classical molecular dynamics are the canonical ensemble (NVT) and the isothermal-isobaric ensemble (NPT). In both cases, the system is in isolation aside from contact with a much a larger heat bath that exchanges energy but not particles with the system. The canonical ensemble keeps the number of particles (N), the volume (V), and the temperature (T) constant in the system.

MD models the individual particles of the system of interest. Particles interact through "force fields" which consist of potential energy equations. Forces acting on the system are derived from these equations. Many different forms of potentials exist for MD programs. Typically for fully atomistic force fields the bond lengths and angles are modeled as harmonic oscillators, atomic charges with Coloumb's law, and van der Waals interactions with the Lennard-Jones potential. Parameters used in the potentials are tediously fit to reproduce experimental and computationally derived data.

At each step of the simulation, the forces on each particle are calculated according to the potential functions and the particles moved according to the numerical integration of Newton's equations of motion. To stay consistent with the statistical mechanical ensemble of interest, special computational methods are employed. Termed the "thermostat" this component maintains the temperature of the system usually by modifying the velocities of individual particles. This operation is equivalent to the heat bath formalism of statistical mechanics. Constant volume, on the other hand, is enforced by confining the system to a box. Isolation of the system is difficult without introducing boundary effects. To avoid these issues periodic boundary conditions are used, whereby a particle colliding with the box wall instead is moved to the opposite side of the box. Constant pressure can be similarly enforced by a barostat. A common implementation scales the size of the box to maintain the correct pressure.

Molecular Dynamic simulations span a range of timescales and system sizes. The larger the system size the more computationally intensive it is to simulate the system. As a direct result, to simulate large systems fewer details of the system are included to make simulation tractable.

When using fully atomistic methods, the solvent is typically represented at the atomistic level. Including the solvent is a significant computational cost but is crucial for accurately representing many system properties such as the water-mediated interactions of proteins. Due to the huge computational cost, atomistic modeling of large systems for a significant time period (microseconds to milliseconds) requires enormous amounts of computing power. Disregarding some of the finer details of the system, by contrast, allows for both larger

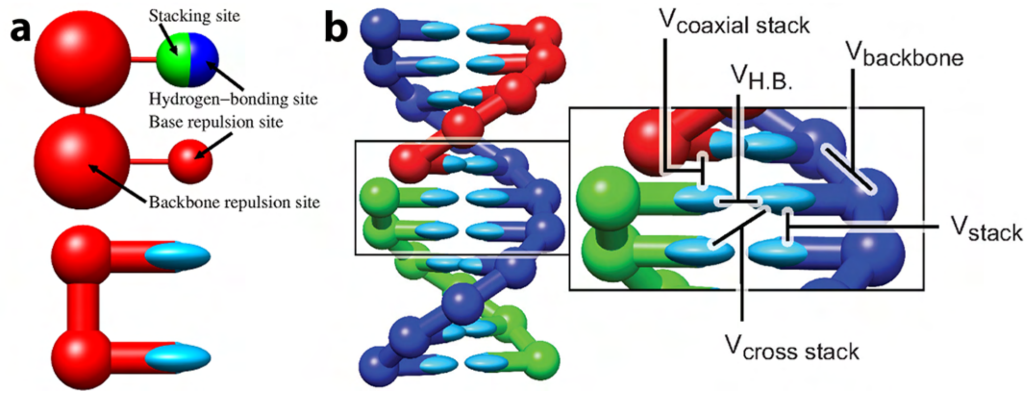


Figure 3. (a) Topology of oxDNA nucleotides. (b) Double helix and interactions of oxDNA model.

system sizes and longer simulation times. Coarse-graining is a technique that represents groups of atoms as a single particle. In conjunction with implicit solvents, the computational demand scales much more reasonably with system size. Removing some details of the system also has the effect of smoothing the potential energy surface and speeding up the sampling and dynamics of the system.

1.2.2.1 The oxDNA Model

One of the most prevalent simulation models for simulation of DNA and RNA nanotechnology is oxDNA. oxDNA is a coarse-grained model that represents each nucleotide as a single particle and has been parameterized to reproduce the thermodynamic and structural properties of DNA [77]. Notable examples of oxDNA's usefulness to the DNA nanotech field include simulations and insights into meta-DNA structures [78], toehold-mediated strand displacement reactions [79], jointed DNA nanostructures [80], and self-assembly of DNA nanostructures [81].

Each nucleotide in the model is represented as a single rigid body with two interactions sites: the base and the backbone. Interactions between nucleotides capture the base pair stacking, hydrogen bonding, backbone covalent bonds, and salt-screened electrostatics of

DNA. An illustration of the oxDNA model topology and interactions is shown in Figure 3. Through careful parameterization, the model correctly reproduces single-stranded and B-form DNA with realistic hybridization kinetics [82]. A layered crossover tile from Hong et. al.[76] is shown in the oxDNA format in Figure 2a.

There are some smaller issues with the model itself. It is unable to reproduce G-motif, I-motif, Z-form, or A-form DNA structures. As these are seldom used in nanotechnology or experimental applications, most systems are unaffected. Cations are not directly represented in the model, and are instead accounted for by parameters in the Debye-Hückel potential that accounts for the screened electrostatics of the system.

Though able to accurately represent large DNA with accurate dynamics, the model is unable to interface directly with other molecules such as proteins. Hybrid DNA- protein nanotechnology in particular necessitates the ability to predict the dynamics and characteristics of large hybrid nanostructures. Design of such structures is not trivial as the dynamics of the protein can have major effects on the DNA component of the system. If designed as a single static structure, assembly yields and dynamic behavior can be quite different than intended (or designed). It can be expensive to synthesize a single hybrid structure, so coming up with a predictable design is critical. To remedy these issues, I introduce in this dissertation an extension of oxDNA as a simulation model for hybrid DNA-protein conjugates and further show its application to large scale DNA-protein hybrids.

1.2.3 Machine Learning

One of the most promising fields and tools to emerge in the past twenty years is that of machine learning. “Machine learning” is an umbrella term for a variety of statistical algorithms that, at their most basic level, create a model from some training data in order to perform tasks on other new, unobserved data. Heavily based in statistics and optimization theory, machine learning is being increasingly used in standard commercial settings as well as in scientific research. Figure 2b is the topology of a very basic neural network.

Applications of machine learning for biological datasets include Alphafold2 [83], which was trained on protein sequences and their experimentally resolved structures (e.g. from crystal structures). Alphafold2 can predict the folded structure of proteins and has already been used to predict almost the entire human proteome [84]. Though questions still remain over the algorithm's accuracy, particularly for single position mutations and sequences with little homology, it still represents a significant advance in the field of protein folding prediction.

Pharmaceutical companies are increasingly leveraging machine learning to aid in the design process of novel therapeutics. Using recent technologies such as high-throughput sequencing, the amount of data is growing at a rate much higher than any human could hope to sort through it all. Direct utilization of machine learning for genomic datasets is complicated by the lack of inherent labels for supervised methods, and the amount of noise present in the experimentally derived data.

One exciting use of machine learning is that molecular dynamics force fields can now be fit automatically utilizing differentiable operations through supported frameworks such as PyTorch or JAX-MD[85]. Rather than tuning parameters by hand, the software finds the best parameters via gradient descent of specified metrics. While established force fields such as CHARMM and AMBER will continue to be used, machine learning methods may eventually lead to more accurate or less computationally-demanding force fields.

Neural networks have also been directly applied to calculate the forces on each particle for molecular dynamics. The ANI-1 force field claims to have near-DFT accuracy for the computational cost of a traditional fully atomistic force field [86]. Force fields for water have been similarly developed [87], as have reactive force fields [88]. Like any novel method, there are challenges when using machine learned force fields. They have similar problems of scalability to large system sizes from the ever-present many body problem. Simply the interactions between a large collection of particles poses a high dimensional problem that gets exponentially more difficult to solve with increasing system size. This drawback plagues ab-initio predictions and fundamentally limits the system sizes of both the training data and predictions using neural networks. Like any machine learning algorithm, machine learned force fields perform poorly at states far (in some parameter) from the training data. Introduction of unphysical effects may occur during a simulation if presented with a corresponding configuration [89].

In the following sections, I discuss the background of a simple generative model the Restricted Boltzmann Machine (RBM) and its application to sequence data.

1.2.3.1 Restricted Boltzmann Machine

A shallow neural network composed of two layers, the Restricted Boltzmann Machine (RBM), was first developed in 1986 by Paul Smolensky [90]. However, they were popularized by Geoffrey Hinton and company in the early-to-mid-2000s [91]. RBM's restrict all connections in the network to be between opposite layers, i.e. there are no connections between two visible units or two hidden units. This "restriction" enables a simplified learning procedure as opposed to the Boltzmann machine which contains intra-layer connections. The topology of the model is displayed in Figure 4.

Energy-based models aim to associate a scalar, namely the energy, with each configuration of the input. The energy determines the compatibility of the model parameters with the input, with low energy values being in agreement with model parameters [92]. The RBM is an energy-based model that aims to decompose the dataset (represented as the visible layer) into independent latent factors (the hidden layer) from messages passed between the layers. This is a probabilistic graphical model that can be represented as a bipartite factor graph with conditionally independent probability distributions of the visible and latent factors [93]. RBM's make use of Markov Chain Monte Carlo methods to sample from the conditional probability distributions, most commonly using Gibbs sampling.

The probability of a given configuration of visible and hidden units can be expressed as a Gibbs distribution as in equation 1.1

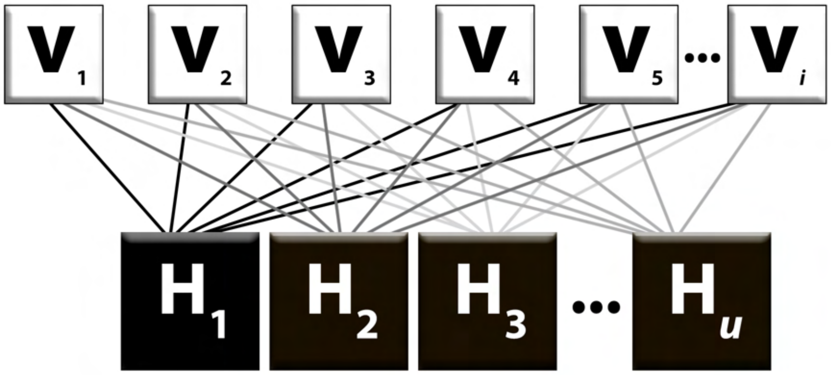


Figure 4. a) Model Topology of Restricted Boltzmann Machine (RBM). V represents the visible layer. H represents the hidden layer. The connections between the visible and hidden layers are the weight matrix.

$$P(v,h) = e^{-E(v,h)}/Z \tag{1.1}$$

where Z is the partition function given by equation 1.2.

$$Z = \sum_{v,h} e^{-E(v,h)} \tag{1.2}$$

The sum in equation 1.2 is replaced by a integral for continuous visible and hidden variables.

The energy function is defined by the visible layer, hidden layer, and the connections between them. For a binary RBM the energy function is given as:

$$E(v,h) = -b^T v - c^T h - b^T W h \tag{1.3}$$

The visible layer is represented as v, the hidden layer as h, the visible biases as b, the hidden biases as h, and the weights between the hidden and visible layers as W.

Conditional probabilities for both the hidden and visible units can be derived using Bayes theorem and used for sampling new visible and hidden layers. The probability of the training dataset can be described by the marginal distribution over all possible states of the hidden layer. Evaluation of P(v), however, requires knowledge of the partition function as shown in 5.7.

$$P(v) = \frac{\sum_h P(v,h)}{Z} \tag{1.4}$$

Taking the negative log of P(v) is equivalent to the free energy of the model and can be expressed as Equation 1.5

$$F(v) = -\log\left(P(v)\right) = -\log(\sum_h P(v,h)) + \log(\sum_{v,h} P(v,h)) \tag{1.5}$$

Maximizing the probability of the training data P(v) is equivalent to minimizing the free energy F(v). Performing gradient descent on Equation 1.5 in turn maximizes the probability (lowering the energy) of our data in the model while minimizing the probability (raising the energy) of the rest of the energy landscape. However, the second term in Equation 1.5 is the partition function Z which is impossible or very expensive to compute, especially in high dimensional spaces.

To train the RBM, an approximation of the partition function is needed. The most common learning algorithm for training a RBM is known as contrastive divergence which generates samples from the model's current parameterization and uses the free energy of the generated samples to approximate the partition function. The net effect of this learning procedure is the energy of samples from our dataset are made more likely (by lowering their energy) while the generated samples are made less likely (by raising their energy). Typically the generated samples are generated by iteratively sampling a hidden layer from the visible layer and then sampling a new visible layer given the hidden layer. While not calculating the exact gradient of the free energy equation, it does perform well in practice [91].

1.2.3.2 RBMs for Sequence Datasets

Many methods have sought to perform analysis on biological sequence datasets. Direct Coupling Analysis has been used in protein structure prediction [94, 95] and RNA secondary structure prediction [42]. Variational AutoEncoders have been used for aptamer design and prediction [96].

A unique approach for aligned protein sequence data, Tubiana *et al.* implemented an RBM with multinomial visible units and a dReLU activation function on the hidden units [97]. As the potential used on the hidden units influences the statistics of hidden units, the dReLU activation influences the statistics less than other activations such as bernoulli or gaussian as it can model asymmetric, symmetric, gaussian, and super-gaussian distributions [98]. The dReLU activation has four parameters and is shown in Equation 1.6.

$$\mathcal{U}(h) = \frac{1}{2}\gamma^{+}h^{+2} + \frac{1}{2}\gamma^{-}h^{-2} + \theta^{+}h^{+} + \theta^{-}h^{-} \tag{1.6}$$

$$h^{+} = max(x, 0), h^{-} = min(h, 0)$$

This model has been shown to find biologically relevant features in aligned protein sequences and be able to generate sequences far from the training dataset. It has also been used for aptamer design and generation as seen previously in 4 as well as major histocomptability complex antigen Prediction [99]. While powerful, this model does have some limitations. Due to the structure of the model, the features that the model finds are locked into specific positions of the visible layer. Therefore the model cannot be applied to unaligned data, or data of different lengths. RNA and peptide aptamers often have common motifs that vary in their sequence location. To account for these, a different model is needed which does not encode the positional dependence of each feature. Focus on the development of a model fitting the criteria is discussed in Chapter 5.

1.2.3.3 Aptamer Prediction

An aptamer is any molecule that tightly binds a specific target. Targets can be but are not limited to single proteins, small molecules, or specific tissues. Using naturally occurring biopolymers such as DNA, RNA or polypeptides (depending on the scaffold protein) has the distinct advantage of avoiding an immune response [45]. Further, specific binders have huge potential as therapeutics or diagnostics depending on their particular target.

Generating these aptamers is typically accomplished using a combinatorial approach, where a large random library of sequences is used initially and eventually tight specific binders are generated. In a cyclical fashion, the library is exposed to the target, the unbound sequences washed away, and the bound sequences then eluted, sequenced and used as the starting library for the next cycle. For DNA and RNA aptamers, this cycle is known as the Selective Evolution of Ligands by Exponential Enrichment (SELEX). For antibody development, the method is more complicated and involves the use of antibody fragments on phage coat proteins in a technique known as phage display.

Analysis of the sequence data from these cycles typically yields a few strong binders and a few enriched motifs in the dataset. Depending on the target and experimental details, this process could leave a significant number of good or better binders unsampled. To sample these unobserved binders, we can apply machine learning methods to generate novel sequences from our data. In Chapter 4, I describe the use of unsupervised machine learning methods to generate novel thrombin aptamers. Novel approaches are discussed in Chapter 5.

1.3 List of Publications

1. **Procyk, J.**, Poppleton, E., & Šulc, P. (2021). Coarse-grained nucleic acid-protein model for hybrid nanotechnology. *Soft Matter*, 17(13), 3586–3593.
2. Narayanan, R. P.$^{+}$, **Procyk, J.**$^{+}$, Nandi, P.*, Prasad, A.*, Xu, Y.*, Poppleton, E.,

Williams, D., Zhang, F., Yan, H., Chiu, P. L., Stephanopoulos, N., & Šulc, P. (2022). Coarse-Grained Simulations for the Characterization and Optimization of Hybrid Protein-DNA Nanostructures. *ACS Nano*, 16, 14086–14096.

3. Bohlin, J.$^{+}$, Matthies, M.$^{+}$, Poppleton, E.$^{+}$, **Procyk, J.**$^{+}$, Mallya, A., Yan, H., & Šulc, P. (2022). Design and simulation of DNA, RNA and hybrid protein–nucleic acid nanostructures with oxView. *Nature Protocols*, 17(8), 1762–1788.
4. Di Gioacchino A.$^{+}$, **Procyk J.**$^{+}$, Molari M, Schreck, J. S., Zhou, Y., Liu, Y., Monasson, R., Cocco, S., & Šulc, P. (2022) Generative and interpretable machine learning for aptamer design and analysis of in vitro sequence selection. *Plos Computational Biology*, 18(9), e1010561.

+ *Co-first Authors*

* *Co-second Authors*

Chapter 2

ANISOTROPIC NETWORK MODEL

This chapter was published in **Procyk, J.**, Poppleton, E., & Šulc, P. (2021). Coarse-grained nucleic acid-protein model for hybrid nanotechnology. *Soft Matter*, 17(13), 3586–3593.

2.1 Abstract

The emerging field of hybrid DNA - protein nanotechnology brings with it the potential for many novel materials which combine the addressability of DNA nanotechnology with the versatility of protein interactions. However, the design and computational study of these hybrid structures is difficult due to the system sizes involved. To aid in the design and *in silico* analysis process, we introduce here a coarse-grained DNA/RNA-protein model that extends the oxDNA/oxRNA models of DNA/RNA with a coarse-grained model of proteins based on an anisotropic network model representation. Fully equipped with analysis scripts and visualization, our model aims to facilitate hybrid nanomaterial design towards eventual experimental realization, as well as enabling study of biological complexes. We further demonstrate its usage by simulating DNA-protein nanocage, DNA wrapped around histones, and a nascent RNA in polymerase.

2.2 Introduction

Molecular nanotechnology designs biomolecular interactions to assemble nanoscale devices and structures. DNA nanotechnology, in particular, has attracted lots of attention and experienced rapid growth over the past three decades. While originally envisioned as a method of developing a DNA lattice for crystallizing proteins for structure determination Seeman [29], DNA nanotechnology is seeing promising applications in e.g. biomaterial assembly [100], biocatalysis [101], therapeutics [102], and diagnostics [103]. The programmability of DNA allows for the rapid design and experimental realization of complex shapes, yielding an unprecedented level of control and functionality at the nanoscale. As DNA nanotechology has developed, so have parallel technologies with other familiar biomolecules such as RNA [44], and, to some extent, proteins [104, 105]. While DNA nanostructures and devices have been unequivocally successful in realizing more complex and larger constructs, they are inherently limited in function by their available chemistry, with a possible solution using functionalized DNA nanostructures [106].

Of particular interest is hybrid DNA-protein nanotechnology, which can combine the already well developed design strategies of DNA nanotechnology and cross-link them with functional proteins. The combination of the two molecules in nanotechnology will open new applications, such as diganostics, therapeutics, molecular "factories" and new biomimetic materials [64]. Examples of successfully realized hybrid nanostructures include DNA-protein cages [65], a DNA nanorobot with nucleolin aptamer for cancer therapy [102] and peptide-directed assembly of large nanostructures [107].

At the same time, computational tools for the study and design of DNA and RNA nanostructures have become increasingly relevant as size and complexity of nanostructures grow. Design tools such as Adenita [66] MagicDNA[108], CaDNAno[68], and Tiamat[69] are essential for the structural design of DNA origamis. New coarse-grained models have been introduced to study DNA nanostructures, as the sizes (thousands or more) as well as rare events (formation or breaking of large sections of base pairs) involved in study of these systems make atomistic-resolution modeling more difficult. Several coarse-grained models have been developed to match thermodynamic and energetic properties of nucleic acids [109, 110, 111, 112]. Among the available tools, the oxDNA and oxRNA models [113, 114, 115, 116] have been quite popular over the past few years, being used by dozens of research groups in over one hundred articles to study various aspects of DNA and RNA nanosystems including the biophysical properties of DNA and RNA [80, 117, 118, 76, 119, 120]. Each nucleotide is represented as a rigid body in the simulation, with interactions between different sites parameterized to reproduce mechanical, structural and thermodynamic properties of single-stranded and double-stranded DNA and RNA respectively.

However, the oxDNA/oxRNA models only allow for representation of nucleic acids alone, limiting their scope of usability. While there have been coarse-grained simulation models developed for protein-DNA interactions [121, 122, 123, 124, 125, 126, 127], none are able to be directly used with the oxDNA model. The development of an efficient tool compatible with oxDNA would allow for efficient study of arbitrary protein-DNA complexes.

Here, we introduce such a coarse-grained model that uses an Anisotropic Network Model (ANM) to represent proteins alongside the oxDNA or oxRNA model. The ANM is a form of elastic network model used to probe the dynamics of biomolecules fluctuating around their native state. Originally formulated by Atilgan et. al.[128], the ANM has become fundamental tool in probing protein dynamics, often closely matching residue-residue fluctuations and normal modes of fully atomistic simulations [129, 130, 131]. Here we use the ANM to approximately capture native state protein dynamics. The ANM representation of proteins interact with just an excluded volume interaction with the oxDNA / oxRNA representation, but specific attractive or repulsive interactions can be added as well. The mass of each residue is set as equal to that of a nucleotide. The less than one order of magnitude difference between the average masses of nucleotides and amino acids makes the equal mass approximation acceptable within the high level of coarse-graining employed by ANM and oxDNA/oxRNA models. We further provide parameterization of common linkers that are used to conjugate proteins to DNA in typical hybrid nanotechnology applications.

The ANM-oxDNA/oxRNA hybrid models are intended to help design and probe function of large nucleic-acid protein hybrid nanostructures, but also aim to be used to study biological complexes and processes which can be captured within the approximations employed by the models. As an example of the model's use, we show simulations of DNA-protein hybrid nanocage, DNA wrapped around a histone, and a nascent RNA strand inside a polymerase exit channel.

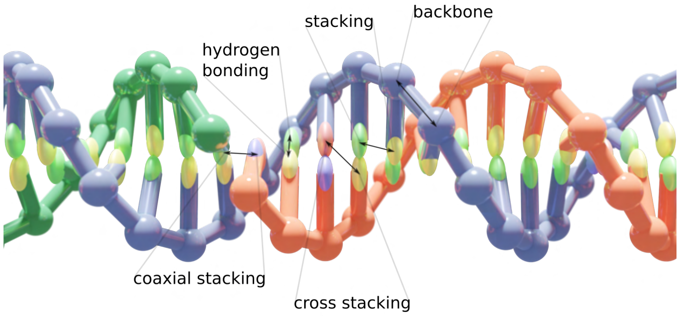


Figure 5. A schematic overview of the oxDNA2 model and its interactions. Each nucleotide is represented as a single rigid body with backbone and base interaction sites (shown here schematically as a sphere and an ellipsoid) with their effective interactions designed to reproduce basic properties of DNA.

2.3 Model Description

Implemented in the oxDNA simulation package [132], our model allows for a coarse-grained simulation of large hybrid nanostructures. It consists of two coarse-grained particle representations, the already existing oxDNA2 or oxRNA model for their respective nucleic acids and an Anistropic Network Model (ANM) for proteins [128]. The detailed description of the oxDNA2/oxRNA models is available in Refs. [114, 115]. A DNA duplex with a nicked strand is schematically illustrated in Fig. 5. The ANM allows us to represent a protein with a known structure as beads connected by springs. We chose to use the ANM to represent proteins for its efficiency and relative simplicity, while still providing reasonably accurate representations of proteins crosslinked to DNA nanostructures. Furthermore, it can be implemented using only pairwise interaction potentials, the same as oxDNA/oxRNA models.

Table 1. Excluded volume parameters used in Eq. 2.2 for (a) protein-protein, (b) protein-nucleic base and (c) protein-nucleic backbone non-bonded interactions in simulation units.

Parameter	(a)	(b)	(c)
σ	0.350	0.360	0.570
r_c	0.353	0.363	0.573
r^*	0.349	0.359	0.569
b	30.7×10^7	29.6×10^7	17.9×10^7

2.3.1 Protein Model

In the ANM representation, each protein residue is represented solely by its α-carbon position. All residues within a specified cutoff distance r_{max} from one another are considered 'bonded'. Please see Ref. [128] for a more detailed introduction. Each bond between residues i and j in the ANM is represented as a harmonic potential that fluctuates around the equilibrium length r_0^{ij}:

$$V_{ij}\left(r^{ij}\right) = \frac{1}{2}\gamma\left(r^{ij} - r_0^{ij}\right)^2 \tag{2.1}$$

The total bonded interaction potential $V_{bonded-anm}$ is the sum of terms Eq. (2.1) for all pairs i, j of aminoacids at a distance smaller than r_{max} in the resolved protein structure, as schematically illustrated in Fig. 6. We set r_0^{ij} to the the distance between α-carbons of the residues i and j in the PDB file. Free parameter γ is set uniformly on each bond in the ANM and and is chosen to best fit the Debye-Waller factors of the original PDB structure. Debye-Waller factors (or B-factors when applied specifically to proteins) describe the thermal motions of each resolved atom in a protein given by their respective X-ray scattering assay. As previously done[128], we use the B-factor of the α-carbon to approximately capture the fluctuations of the protein backbone.

Since an ANM is typically an analytical technique, it has no excluded volume effects. Hence we here extend the model to use a repulsive part Lennard-Jones potential between both bonded and non-bonded particles (Eq. 2.2) to model the excluded volume at a per particle excluded volume diameter of 2.5 Å.

For any two particles (either protein/protein or protein-DNA/RNA) that are at distance

r, we define the excluded volume interaction in Eq. 2.2:

$$V_{exc}(r) = \begin{cases} 4\epsilon(-\frac{\sigma^6}{r^6} + \frac{\sigma^{12}}{r^{12}}) & r < r^* \\ b\epsilon(r - r_c)^4 & r^* < r < r_c \\ 0 & r \geq r_c. \end{cases} \tag{2.2}$$

The excluded volume diameter r_c between protein particles was set by simulating both large and small proteins at various values to tune to a value allowing excluded volume interactions between nearest neighbors with little deviation between simulated and analytical B-factors. Protein-DNA/RNA r_c values were set as the sum of the excluded volume radii of both particle types. Parameters b and r^* were calculated so that V_{exc} is a differentiable function. The constant ϵ sets the strength of the potential and we use $\epsilon = 82\, pN\, nm^{-1}$.

2.3.1.1 Parameterization

In parameterizing our model for simulation, the goal is to mimic the dynamics of the protein in the native state. Though not without their drawbacks [133, 134], we selected B-factors for their widespread availability in PDB structures and history of being used to fit elastic network models of proteins [133]. Our model contains two free parameters, the cutoff distance r_{max} and the spring constant γ. The r_{max} value alone determines which connections will be present in the ANM network. As noted in the original formulation of the ANM [128], the best choice of r_{max} should reproduce the distribution found for globular proteins' densities of vibrational states [135, 136]. A value of 13 Å was found to approximately capture the shape of the target distribution for a large set of proteins with r_{max} values much lower (7 Å) or higher (20 Å) tending to shift the eigenfrequencies towards lower and higher frequencies respectively. In practice, the best r_{max} varies from protein to protein but can usually be varied in a narrow range (12-18Å) with little effect on the distribution of normal mode frequencies.

For each protein (consisting of N aminoacids) represented by ANM, we linearly fit the

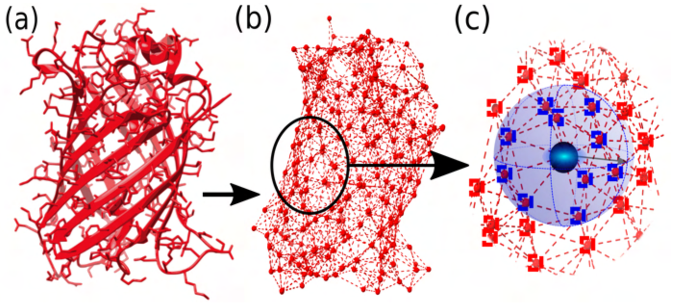


Figure 6. Illustration of ANM using GFP protein (PDB code: 1W7S) from (a) starting PDB structure to (b) ANM representation at r_{max} of 8 Å, (c) bonding criteria per residue: all particles within distance r_{max} (bounds depicted by blue sphere) of center particle (black circle) are considered 'bonded' (blue squares) while those further (outside of sphere) are considered 'nonbonded' (red squares).

analytically computed B-factors to their experimental counterpart with γ as a free parameter. To solve for the B-factors analytically, we first calculate the $3N \times 3N$ Hessian matrix of the spring potential V_{spring}, a task made simple by the harmonic potential energy function [128]. After constructing the Hessian H for the system at a specified cutoff r_{max}, the mean squared deviation from the mean position for each residue i can be calculated from the ensemble average:

$$\langle \Delta R_i^2 \rangle = \frac{k_b T}{\gamma} \left(Tr \left(H_{i,i}^{-1} \right) \right) \tag{2.3}$$

The B-factor B of the residue i can be directly computed from our previous result as [128]:

$$B_i = \frac{8\pi^2}{3} \langle \Delta R_i \rangle^2 . \tag{2.4}$$

The experimental B-factors are provided along with resolved crystal structures of proteins, and we can hence use Eqs. (2.3) and (2.4) to obtain N equations. We then fit γ parameter to minimize

$$f(\gamma) = \sum_{i=1}^{N} \left(B_i^{exp.} - \frac{8\pi^2}{3} \langle \Delta R_i \rangle^2 \right)^2 \tag{2.5}$$

for a selected r_{max}.

We can further measure the mean square deviation of residue positions in a simulation of our model and compare to the analytical calculation. We show the comparison in Fig. 7

for ribonuclease T1 and green fluoresecent proteins simulated with the ANM model and our ANMT model, to be introduced later. While the simulation and analytical prediction of the classic ANM agree well with each other, as expected, we note that the model still does not fully reproduce the measured B-factors as reported in the experimental structures. ANM models are not able to fully reproduce the measured B-factors [128], and are known to have peaks in the mean square displacement profiles that have not been observed in the measured B-factors [133]. The model nevertheless provides semi-quantitative agreement with the measured data, and hence represents an accurate enough representation of a protein to model its mechanical properties under small perturbations, as required for DNA-hybrid nanotechnology systems.

2.4 Expansion of the ANM model

In addition to the classic ANM model, our model can also optionally use unique γ_{ij} for each bonded pair of residues, which allows for implementation of other analytical models, such as the heterogeneous ANM (HANM)[137] and multiscale ANM (mANM) [138] that can generate better fits to experimental B-factors using the γ_{ij} values. The HANM iteratively fits a normal ANM network to given experimental B-factors with variable realistic force parameters γ_{ij}. While unquestionably useful, the inaccuracy of B-factor data particularly in large or low resolution structures limits its application. In the mANM model, our conversion from the PDB structure to ANM representation also allows the fitting of multiple networks with varying γ_{ij} values tuned by scale parameters[138] (similar to r_{max}). A linear combination of the networks is then solved to minimize the difference between the ANM network's predicted and experimental B-factors. The original formulation of the mANM[138] is limited in computational application as it has no cutoff value (r_{max}); a protein of size N residues would have $N(N-1)/2$ connections, significantly more than the average ANM. For the proteins studied in this work, neither HANM nor mANM provided a significant advantage,

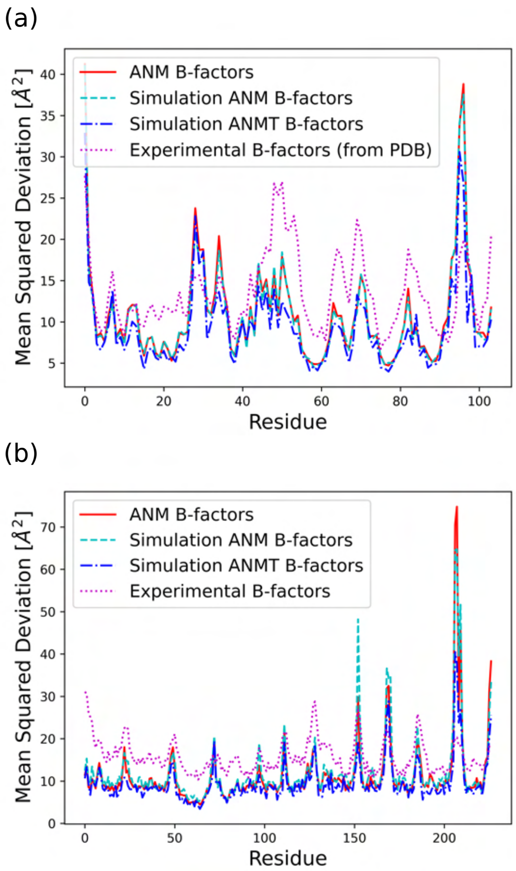


Figure 7. Analytical, classic ANM simulation, ANMT simulation, and experimentally determined B-factors calculated in Å^2 per residue for *(a)* ribonuclease T1 (PDB code 1BU4) at $25^{\circ}C$ ($r_{max} = 15\text{Å}, k_s = 42.2pN/\text{Å}, k_b = k_t = 171.3pN/\text{Å}$) and *(b)* green fluorescent protein (PDB code 1W7S) at $25^{\circ}C$ ($r_{max} = 13\text{Å}, k_s = 33.2pN/\text{Å}, k_b = k_t = 171.3pN/\text{Å}$)

so we decide to use the simple ANM with fixed r_{max} and the same γ for all spring interactions. A Cα coarse-grained HANM and a mANM with an additional cutoff value parameter are, however, implemented in our conversion scripts and can be optionally used to represent proteins in our model.

One major obstacle in using an ANM is known as the tip effect [139]. The result is an extremely large spike in the B-factors due to a residue being under-constrained. Often this can be solved by raising the cutoff value in ANM construction; however, doing so raises the computational requirements of simulations. Furthermore, we found the ANM model did not accurately represent short peptides, as the spring network does not provide enough constraints to reproduce their end-to-end distance as seen when simulated with more detailed models like AWSEM-MD[140]. To overcome this obstacle, we implemented harmonic pairwise bending and torsional modulation forces into the existing simulation model. These new constraints allow for reduced r_{max} values, and also can more accurately represent shorter peptides, which are often used in DNA-hybrid nanostructures. We introduce these optional modulation forces below.

2.4.1 Bending and Torsional Modulation

We introduce the torsional and bending potential as optional interaction potentials in our protein representation on top of the ANM model with bonded and excluded volume potentials. Each protein residue corresponds to a spherical particle, with associated orientation given by its orthonormal axes $\hat{i}_1$, $\hat{i}_2$, $\hat{i}_3$ (Fig. 8a). Harmonic terms control the angle between the normalized interparticle distance vector $\hat{r}_{ij}$ and the normal vector of each particle $\hat{i}_1, \hat{j}_1$ to control bond bending. The angles between two sets of orientation vectors, $\hat{i}_1, \hat{j}_1$ and $\hat{i}_3, \hat{j}_3$, are controlled as well allowing for modulation of the torsion based on the particles relative orientations. The full pairwise potential is given by Eq. 2.6:

$$V_{ij}^{B\&T} = \frac{k_b}{2}\left(\left(\hat{\mathbf{r}}_{ij}\cdot\hat{\mathbf{i}}_1 - a_0^{ij}\right)^2 + \left(-\hat{\mathbf{r}}_{ij}\cdot\hat{\mathbf{j}}_1 - b_0^{ij}\right)^2\right) + \frac{k_t}{2}\left(\left(\hat{\mathbf{i}}_1\cdot\hat{\mathbf{j}}_1 - c_0^{ij}\right)^2 + \left(\hat{\mathbf{i}}_3\cdot\hat{\mathbf{j}}_3 - d_0^{ij}\right)^2\right) \tag{2.6}$$

The function $V_{ij}^{B\&T}$ is defined for all pairs of residues that are neighbors along the protein backbone. We set the energy minimum values $a_0^{ij}, b_0^{ij}, c_0^{ij}, d_0^{ij}$ to correspond to the cosines of respective angles in between residues in the PDB file for the protein structure. The terms k_b and k_t are two new global parameters that control the strength of the bending and torsion potential respectively. Currently, we set their values empirically, though pair specific terms could lead to further agreement with experimental data. Fig. 7 shows the effect of the torsional and bonding modulation on the same set of proteins used prior. As intended, a noticeable decrease in high peak B-factors is observed using a modest k_b and k_t value. Fig. 8

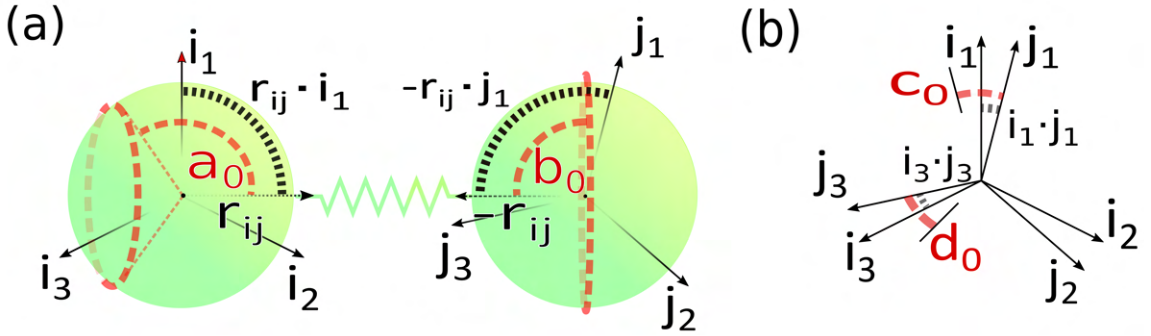


Figure 8. Depiction of (a) bending and (b) torsional potential terms on a pair of particles i and j. The angles depicted as dot products correspond to the cosine of that angle. Equilibrium values (in red) correspond to (the cosine of) initial angle displacements derived from coordinates in the PDB file.

illustrates the potential in a two particle system. Hereafter, we will refer to the ANM model with torsional and bending modulation as the ANMT model.

2.4.2 Protein-Nucleic Acid Interactions

In our current implementation of the model, protein residues and nucleotides have no interaction except for excluded volume and optional explicitly specified spring potentials between user-designated protein residues and nucleotides:

$$V_{spring}(r) = \frac{k}{2}\left(r - r_0\right)^2 \tag{2.7}$$

where r is the distance between the centers of mass of the respective particles and k and r_0 and external parameters.

The excluded volume interaction potential between protein and DNA/RNA residues has the same form as defined in Eq. (2.2), with the respective interaction parameters given in Table 1. In the oxDNA/oxRNA models, each nucleotide has two distinct interaction sites (backbone and base), each of which is interacting with the protein residue using separate excluded volume parameters.

Future expansion of the model will include an approximate treatment of electrostatic interaction between protein and nucleic acids based on Debye-Hückel theory as implemented in oxDNA [114], as well as coarse-grained protein model AWSEM[140].

Many non-specific DNA-protein interactions make use of the electrostatic interactions between the DNA backbone and positively charged portions of the protein [141]. Sensitive to salt concentration, these electrostatic contributions have been previously modeled using Debye-Hückel theory[142] to investigate the role of protein frustration in regulating DNA binding kinetics. Similarly an extension of our model with an appropriate Debye-Hückel potential can capture and enable study of non-specific DNA-binding protein systems.

Since we are interested in exploring conjugated hybrid systems, it is necessary to have an approximation for the covalent linkers bridging the nucleic acid base and protein residue. We model the two bioconjugate linkers, LC-SPDP and DBCO-triazole, (Fig. 9) that are typically used in protein-DNA hybrid nanotechnology [143, 65] using a spring potential as defined in Eq. (2.7) with parameters k and r_0 parameterized to mimic the end-to-end average distance and standard deviation of each linker at temperature 300K. LC-SPDP links the thiol group of a modified cysteine residue to an amine-modified nucleotide. DBCO-trizaole is the product of a copper-free click reaction involving a DBCO-modified residue to link to an azide-modified nucleotide. Each of the linkers (Fig. 9) was first drawn in MolView and then converted into OPLS-AA 1.14*CM1A forcefield format via LibParGen [144, 145, 146]. In GROMACS [147], each linker was first equilibrated and then simulated in both SPCE and TIP3P water molecules at 300K for three trials of 10 nanoseconds each. The obtained averaged end-to-end distance and standard deviation for each trial are shown in Table 2.

Table 2. Average and standard deviation of end-to-end distance of linkers in fully atomistic Gromacs simulation and fit spring constant k

SPCE Solvent	$\langle r \rangle$ (Å)	$\langle r^2 \rangle$ (Å)	k (pN/Å)
LC-SPDP	9.18	2.68	5.75×10^{-2}
DBCO-triazole	10.97	3.43	3.51×10^{-2}
TIP3P Solvent	$\langle r \rangle$ (Å)	$\langle r^2 \rangle$ (Å)	k (pN/Å)
LC-SPDP	9.05	2.8	5.28×10^{-2}
DBCO-triazole	10.95	3.56	3.25×10^{-2}

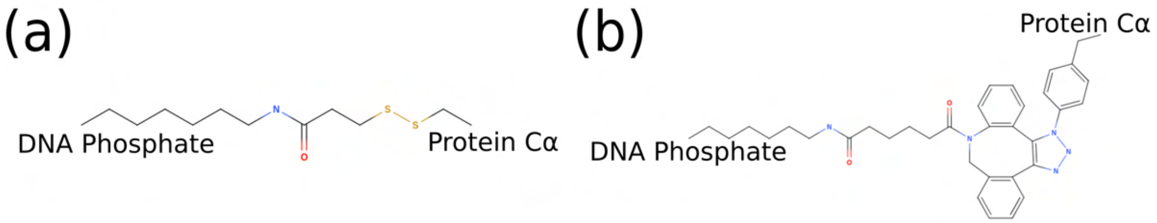


Figure 9. 2D molecular structures of common bioconjugate linkers dubbed (a) LC-SPDP and (b) DBCO-triazole; both can be used to conjugate proteins to DNA phosphate groups

2.5 Examples

Our model is fully functional with the latest version of the visualization tool oxView [148] for both the design of hybrid nanomaterials as well as the viewing of simulation trajectories. The one caveat is that protein topologies are non-editable. Instead each protein starts from their PDB crystal structure and is converted into oxDNA format while the ANM spring constant is set to best match the experimental B-factors via our provided scripts. The output files can then be loaded into oxView as well as used for simulation in our model.

The model is theoretically able to represent any protein or protein complex that the ANM model can represent. Not beyond the scope of our model, biologically relevant multi-chain proteins such as nucleosomes, RNA polymerases, and viral assemblies can be also simulated, allowing for the nucleic acid behavior present in each of these systems to be modeled, studied, and compared to experimental data. While the detailed study of these systems is beyond the scope of this article, we show examples of both biological and designed nanosystems as represented by our ANM-oxDNA or ANM-oxRNA model.

2.5.1 Biological Constructs

Two prominent cases of nucleic acid - protein interactions, RNA polymerases and nucleosomes, were constructed and simulated using the ANMT model for future study. As many PDB files are missing residues, we first reconstruct each individual chain using the best

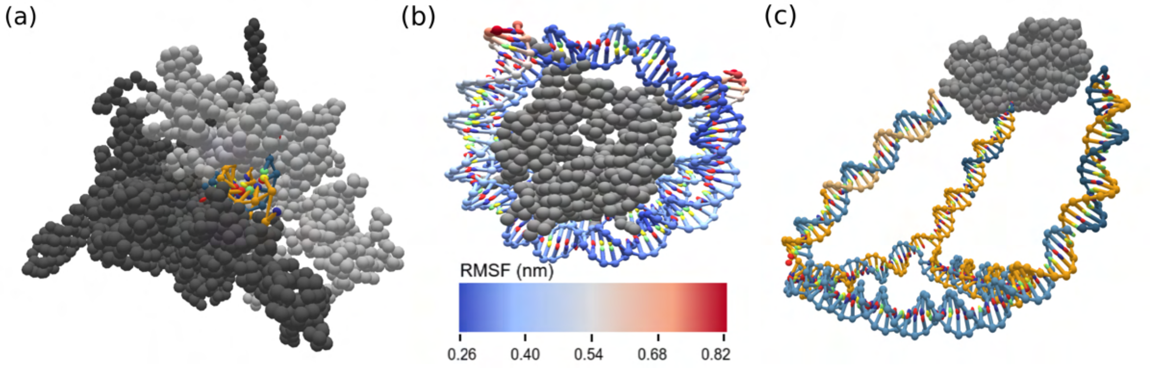


Figure 10. OxView visualization of simulated biological assemblies (a) RNA in exit channel of paused RNA polymerase (PDB code: 6ASX) and (b) Root mean squared fluctuation (nm) of human nucleosome made up of histone octamer and DNA (PDB code: 3LEL), (c) mean structure from MD simulation of KDPG aldolase (PDB code: 1WA3) conjugated to a DNA cage

scoring of ten models generated by the Modeller tool[149]. The reconstructed RNA polymerase was converted into oxDNA format from its PDB entry (6ASX) using an r_{max} of 15 Å. A fragment of the RNA was reinserted into the exit channel and the subsequent MD simulation was allowed to sample the RNA's escape from the exit channel. The reconstructed nucleosome was converted into oxDNA simulation format from its PDB entry (3LEL) using an r_{max} of 12 Å. Spring potentials were added to observed contacts between the DNA and protein residues present in the PDB structure. A snapshot of the RNA polymerase system and fluctuation analysis of the nucleosome are shown in Fig. 10a,b.

While no process was explicitly modeled, our new model can be used to explore behavior of large scale systems of nucleosomes, as at the latest version of GPU cards, the oxDNA model has been shown to be able to equilibrate systems consisting of over 1 million nucleotides.

More pertinent to our goal of aiding in the design of hybrid nanostructures, our model supports conversion of CadNano, Tiamat, and other popular DNA origami design tools into the oxDNA format [118] where they can easily be edited in oxView to include linked proteins of interest. Since an ANM is a highly simplified model of protein dynamics, the predictive power of our model lies not in prediction of protein structure but rather the collection of statistical data of the protein's effect on the nucleic acid component of the system. Available and compatible with this model is also the suite of oxDNA analysis scripts[148] allowing for a detailed exploration of system-specific effects.

2.5.2 Peptides

Synthetic peptides are used in many chemistry applications. Since these peptides are often very small and lack long-distance contacts that enforce specific 3D conformations, we wanted to explore how our models perform on these small structures. We compared the end-to-end distance of 3 hemagglutinin binding peptides[150] simulated in our ANM model, the ANMT model, and another popular coarse-grained protein model, AWSEM-MD[127]. For AWSEM-MD simulations, initial structure predictions were generated from sequence using I-TASSER[151]. A secondary structure weight (ssweight) file was generated using jpred[152], and the structure and weight files were converted to the appropriate formats for AWSEM-MD simulation in LAMMPS[153] using tools provided with AWSEM-MD. Simulations were run for 10^9 steps with end-to-end distance sampled every 10^5 steps.

Using the classic ANM, each peptide was built using strong backbone connections and significantly weaker long-range connections to empirically match the AWSEM mean and standard deviation of the end-to-end distance. The resulting simulation of each peptide; however, showed the trajectory to include a large amount of stretched, nonphysical conformations. The subsequent inclusion of the bending and torsion modulation using the ANMT model allowed for the same level of accuracy using only strong short-range connections. The ANMT model showed much higher rigidity with no stretched conformations when compared to the ANM model alone. Final end-to-end distances and standard deviation are shown in Table 3.

Table 3. Average and standard deviation of end-to-end distance of hemagglutinin peptides between coarse-grained models

Peptide 125 - CSGHNIYAQYGYPYDHMYEG
Peptide 149 - CSGKSQEIGDPDDIWNQMKW
Peptide 227 - CSGSGNQEYFPYPMIDYLKK

Model	AWSEM	ANM	ANMT
Peptide 125			
$\langle r \rangle$ (Å)	12.02	12.9	12.09
$\langle r^2 \rangle$ (Å)	4.9	4.51	4.34
Peptide 149			
$\langle r \rangle$ (Å)	12.9	12.9	12.9
$\langle r^2 \rangle$ (Å)	6.6	4.6	4.6
Peptide 227			
$\langle r \rangle$ (Å)	14.5	16.2	14.7
$\langle r^2 \rangle$ (Å)	7.4	5.4	5.1

2.5.3 KDPG Aldolase-DNA Cage

Hybrid DNA-protein nanostructure constructs such as those developed by the Stepahanopoulos Lab are of particular interest. The Stephanopoulos group has experimentally realized their size-tunable DNA cage attached to homotrimeric protein KDPG aldolase making use of a LC-SPDP linker (Fig. 9) to join the DNA and protein components[65]. The DNA cage was converted from Tiamat format into oxDNA format and the protein was converted from it's PDB structure. The linker between the components was modeled as a spring potential (Eq. (2.7)) using the parameters from Table 2. We conducted a short MD simulation of the full system corresponding to time of about 30 ns. The mean structure from simulation of the experimental cage was calculated using our analysis scripts[148] and is displayed in Fig. 10c.

2.6 Conclusions

We present a coarse-grained protein model, based on elastic network representation of proteins, for use in conjunction with existing coarse-grained nucleic acid models capable of simulating large hybrid nanostructures. Implemented on GPU as well as CPU, our model allows for simulations of large systems based on nanotechnology designs as well as large biological complexes.

Looking forward, we plan to study both the paused RNA polymerase and histone biological systems using this model. In addition, experimental systems such as the hybrid cage in Fig. 10 can be simulated and directly compared to available experimental data. While widely available, B-factors are severely limited particularly in terms of accuracy. However, our model can be parameterized to approximate any available fluctuation data including but not limited to fully atomistic simulation and solution NMR data. In addition to the model, we also extended a nanotechnology design and simulation analysis tool, oxView, to include a protein representation to aid computer design of DNA/RNA-protein hybrid nanostructures. The subsequent analysis of the designs can be used to optimize nanostructure parameters, such as placement of the linkers and lengths of duplex segments in order to achieve desired geometry.

The simulation code is freely available on github.com/sulcgroup/anm-oxdna and will also be incorporated in the future release of the oxDNA simulation package. The visualization of protein-hybrid systems has been incorporated into our previously developed oxView tool [148]. The aforementioned analysis scripts and visualizer are available in git repositories github.com/sulcgroup/oxdna_analysis_tools and github.com/sulcgroup/oxdna-viewer respectively. We also provide the description of the file formats used to setup the simulation in the Supplementary Material.

2.7 Conflicts of interest

There are no conflicts to declare.

2.8 Acknowledgements

We thank all members of the Šulc group for their support and helpful discussions, in particular to H. Liu and M. Matthies. We thank Dr. Stephanopoulos for helpful comments and feedback about simulation study of his DNA-protein hybrid system. We acknowledge support from the NSF grant no. 1931487.

References

[29] Nadrian C. Seeman. "Nucleic acid junctions and lattices". In: *Journal of Theoretical Biology* 99.2 (1982), pp. 237–247.

[44] Peixuan Guo. "The emerging field of RNA nanotechnology". In: *Nature Nanotechnology* 5.12 (2010), pp. 833–842.

[64] Nicholas Stephanopoulos. "Hybrid Nanostructures from the Self-Assembly of Proteins and DNA". In: *Chem* 6.2 (2020), pp. 364–405.

[65] Yang Xu et al. "Tunable Nanoscale Cages from Self-Assembling DNA and Protein Building Blocks". In: *ACS Nano* 13.3 (2019), pp. 3545–3554.

[66] Elisa de Llano et al. "Adenita: interactive 3D modelling and visualization of DNA nanostructures". In: *Nucleic Acids Research* 1 (2020).

[68] Shawn M. Douglas et al. "Rapid prototyping of 3D DNA-origami shapes with caDNAno". In: *Nucleic Acids Research* 37.15 (2009), pp. 5001–5006.

[69] Sean Williams et al. "Tiamat: a three-dimensional editing tool for complex DNA structures". In: *International Workshop on DNA-Based Computers*. Springer. 2008, pp. 90–101.

[76] Fan Hong et al. “Layered-crossover tiles with precisely tunable angles for 2D and 3D DNA crystal engineering”. In: *Journal of the American Chemical Society* 140.44 (2018), pp. 14670–14676.

[80] Rahul Sharma et al. “Characterizing the Motion of Jointed DNA Nanostructures Using a Coarse-Grained Model”. In: *ACS Nano* 11.12 (2017), pp. 12426–12435.

[100] Wenyan Liu et al. “Diamond family of nanoparticle superlattices”. In: *Science* 351.6273 (2016), pp. 582–586.

[101] Chun Geng and Paul J. Paukstelis. “DNA crystals as vehicles for biocatalysis”. In: *Journal of the American Chemical Society* 136.22 (2014), pp. 7817–7820.

[102] Suping Li et al. “A DNA nanorobot functions as a cancer therapeutic in response to a molecular trigger in vivo”. In: *Nature biotechnology* 36.3 (2018), p. 258.

[103] Fei Zhang et al. “Structural DNA nanotechnology: State of the art and future perspective”. In: *Journal of the American Chemical Society* 136.32 (2014), pp. 11198–11211.

[104] Rein V. Ulijn and Roman Jerala. “Peptide and protein nanotechnology into the 2020s: Beyond biology”. In: *Chemical Society Reviews* 47.10 (2018), pp. 3391–3394.

[105] Neil P King et al. “Accurate design of co-assembling multi-component protein nanomaterials”. In: *Nature* 510.7503 (2014), pp. 103–108.

[106] Mikael Madsen and Kurt V Gothelf. “Chemistries for DNA nanotechnology”. In: *Chemical reviews* 119.10 (2019), pp. 6384–6458.

[107] Juan Jin et al. “Peptide assembly directed and quantified using megadalton DNA nanostructures”. In: *ACS Nano* 13.9 (2019), pp. 9927–9935.

[108] Chao-Min Huang et al. “Integrated computer-aided engineering and design for DNA assemblies”. In: *Nature Materials* 20.9 (2021), pp. 1264–1271.

[109] Daniel M. Hinckley et al. “An experimentally-informed coarse-grained 3-site-per-nucleotide model of DNA: Structure, thermodynamics, and dynamics of hybridization”. In: *Journal of Chemical Physics* 139.14 (2013), p. 144903.

[110] Debayan Chakraborty, Naoto Hori, and D. Thirumalai. “Sequence-Dependent Three Interaction Site Model for Single- and Double-Stranded DNA”. In: *Journal of Chemical Theory and Computation* 14.7 (2018), pp. 3763–3779.

[111] Natalia A. Denesyuk and D. Thirumalai. “Coarse-grained model for predicting RNA folding thermodynamics”. In: *Journal of Physical Chemistry B* 117.17 (2013), pp. 4901–4911.

[112] Samuela Pasquali and Philippe Derreumaux. “HiRE-RNA: A high resolution coarse-grained energy model for RNA”. In: *Journal of Physical Chemistry B* 114.37 (2010), pp. 11957–11966.

[113] Thomas E Ouldridge, Ard A Louis, and Jonathan PK Doye. “Structural, mechanical, and thermodynamic properties of a coarse-grained DNA model”. In: *The Journal of chemical physics* 134.8 (2011), 02B627.

[114] Benedict EK Snodin et al. “Introducing improved structural properties and salt dependence into a coarse-grained model of DNA”. In: *The Journal of chemical physics* 142.23 (2015), 06B613_1.

[115] Petr Šulc et al. “A nucleotide-level coarse-grained model of RNA”. In: *The Journal of chemical physics* 140.23 (2014), p. 235102.

[116] Petr Šulc et al. “Sequence-dependent thermodynamics of a coarse-grained DNA model”. In: *Journal of Chemical Physics* 137.13 (2012), p. 5101.

[117] Megan C. Engel et al. “Force-Induced Unravelling of DNA Origami”. In: *ACS Nano* 12.7 (2018), pp. 6734–6747.

[118] Antonio Suma et al. “TacoxDNA: A user-friendly web server for simulations of complex DNA structures, from single strands to origami”. In: *Journal of Computational Chemistry* 40.29 (2019), pp. 2586–2595.

[119] Jonathan P.K. Doye et al. “Coarse-graining DNA for simulations of DNA nanotechnology”. In: *Physical Chemistry Chemical Physics* 15.47 (2013), pp. 20395–20414.

[120] Michael Matthies et al. “Triangulated Wireframe Structures Assembled Using Single-Stranded DNA Tiles”. In: *ACS Nano* 13.2 (2019), pp. 1839–1848.

[121] Adam K. Sieradzan et al. “A new protein nucleic-acid coarse-grained force field based on the UNRES and NARES-2P force fields”. In: *Journal of Computational Chemistry* 39.28 (2018), pp. 2360–2370.

[122] Garima Mishra and Yaakov Levy. “Molecular determinants of the interactions between proteins and ssDNA Molecular determinants of the interactions between proteins and ssDNA”. In: *Proceedings of the National Academy of Sciences of the United States of America* 112.16 (2015), pp. 5033–5038.

[123] Cheng Tan, Tsuyoshi Terakawa, and Shoji Takada. “Dynamic Coupling among Protein Binding, Sliding, and DNA Bending Revealed by Molecular Dynamics”. In: *Journal of the American Chemical Society* 138.27 (2016), pp. 8512–8522.

[124] Cheng Tan and Shoji Takada. “Dynamic and Structural Modeling of the Specificity in Protein-DNA Interactions Guided by Binding Assay and Structure Data”. In: *Journal of Chemical Theory and Computation* 14.7 (2018), pp. 3877–3889.

[125] Bin Zhang et al. “Exploring the free energy landscape of nucleosomes”. In: *Journal of the American Chemical Society* 138.26 (2016), pp. 8126–8133.

[126] Rodrigo V Honorato, Jorge Roel-Touris, and Alexandre MJJ Bonvin. "MARTINI-based protein-DNA coarse-grained HADDOCKing". In: *Frontiers in molecular biosciences* 6 (2019), p. 102.

[127] Aram Davtyan et al. "AWSEM-MD: protein structure prediction using coarse-grained physical potentials and bioinformatically based local structure biasing". In: *The Journal of Physical Chemistry B* 116.29 (2012), pp. 8494–8503.

[128] Ali Rana Atilgan et al. "Anisotropy of fluctuation dynamics of proteins with an elastic network model". In: *Biophysical journal* 80.1 (2001), pp. 505–515.

[129] Sambit Kumar Mishra and Robert L. Jernigan. "Protein dynamic communities from elastic network models align closely to the communities defined by molecular dynamics". In: *PLoS ONE* 13.6 (2018).

[130] M. Gur, E. Zomot, and I. Bahar. "Global motions exhibited by proteins in micro- to milliseconds simulations concur with anisotropic network model predictions". In: *Journal of Chemical Physics* 139.12 (2013), p. 121912.

[131] Lei Yang et al. "Close Correspondence between the Motions from Principal Component Analysis of Multiple HIV-1 Protease Structures and Elastic Network Modes". In: *Structure* 16.2 (2008), pp. 321–330.

[132] Lorenzo Rovigatti et al. "A comparison between parallelization approaches in molecular dynamics simulations on GPUs". In: *Journal of computational chemistry* 36.1 (2015), pp. 1–8.

[133] Zhoutong Sun et al. "Utility of B-Factors in Protein Science: Interpreting Rigidity, Flexibility, and Internal Motion and Engineering Thermostability". In: *Chemical Reviews* (2019).

[134] Edvin Fuglebakk, Nathalie Reuter, and Konrad Hinsen. “Evaluation of protein elastic network models based on an analysis of collective motions”. In: *Journal of Chemical Theory and Computation* 9.12 (2013), pp. 5618–5628.

[135] R Elber and M Karplus. “Low-frequency modes in proteins: Use of the effective-medium approximation to interpret the fractal dimension observed in electron-spin relaxation measurements”. In: *Physical Review Letters* 56.4 (1986), pp. 394–397.

[136] Turkan Haliloglu, Ivet Bahar, and Burak Erman. “Gaussian dynamics of folded proteins”. In: *Physical Review Letters* 79.16 (1997), pp. 3090–3093.

[137] Fei Xia, Dudu Tong, and Lanyuan Lu. “Robust heterogeneous anisotropic elastic network model precisely reproduces the experimental b-factors of biomolecules”. In: *Journal of Chemical Theory and Computation* 9.8 (2013), pp. 3704–3714.

[138] Kelin Xia. “Multiscale virtual particle based elastic network model (MVP-ENM) for normal mode analysis of large-sized biomolecules”. In: *Physical Chemistry Chemical Physics* 20.1 (2017), pp. 658–669.

[139] Mingyang Lu, Billy Poon, and Jianpeng Ma. “A new method for coarse-grained elastic normal-mode analysis”. In: *Journal of Chemical Theory and Computation* 2.3 (2006), pp. 464–471.

[140] Min Yeh Tsai et al. “Electrostatics, structure prediction, and the energy landscapes for protein folding and binding”. In: *Protein Science* 25.1 (2016), pp. 255–269.

[141] Vinod K. Misra et al. “Electrostatic contributions to the binding free energy of the λcl repressor to DNA”. In: 75.5 (1998), pp. 2262–2273.

[142] Amir Marcovitz and Yaakov Levy. “Weak frustration regulates sliding and binding kinetics on rugged protein-DNA landscapes”. In: *Journal of Physical Chemistry B* 117.42 (2013), pp. 13005–13014.

[143] Alex Buchberger et al. “Hierarchical assembly of nucleic acid/coiled-coil peptide nanostructures”. In: *Journal of the American Chemical Society* 142.3 (2019), pp. 1406–1416.

[144] Leela S. Dodda et al. “LigParGen web server: An automatic OPLS-AA parameter generator for organic ligands”. In: *Nucleic Acids Research* 45.W1 (2017), W331–W336.

[145] Leela S. Dodda et al. “1.14CM1A-LBCC: Localized Bond-Charge Corrected CM1A Charges for Condensed-Phase Simulations”. In: *Journal of Physical Chemistry B* 121.15 (2017), pp. 3864–3870.

[146] William L. Jorgensen and Julian Tirado-Rives. “Potential energy functions for atomic-level simulations of water and organic and biomolecular systems”. In: 102.19 (2005), pp. 6665–6670.

[147] H. J.C. Berendsen, D. van der Spoel, and R. van Drunen. “GROMACS: A message-passing parallel molecular dynamics implementation”. In: *Computer Physics Communications* 91.1-3 (1995), pp. 43–56.

[148] Erik Poppleton et al. “Design, optimization and analysis of large DNA and RNA nanostructures through interactive visualization, editing and molecular simulation”. In: *Nucleic Acids Research* 48.12 (2020), e72.

[149] Andrej Šali and Tom L. Blundell. “Comparative protein modelling by satisfaction of spatial restraints”. In: *Journal of Molecular Biology* (1993).

[150] Stephen Albert Johnston et al. “A simple platform for the rapid development of antimicrobials”. In: *Scientific reports* 7.1 (2017), pp. 1–11.

[151] Jianyi Yang and Yang Zhang. “I-TASSER server: new development for protein structure and function predictions”. In: *Nucleic acids research* 43.W1 (2015), W174–W181.

[152] Alexey Drozdetskiy et al. “JPred4: a protein secondary structure prediction server”. In: *Nucleic acids research* 43.W1 (2015), W389–W394.

[153] Steve Plimpton. “Fast parallel algorithms for short-range molecular dynamics”. In: *Journal of computational physics* 117.1 (1995), pp. 1–19.

Chapter 3

APPLICATIONS OF HYBRID NUCLEIC ACID-PROTEIN MODEL

This chapter was published in Narayanan, R. P.$^{+}$, **Procyk, J.**$^{+}$, Nandi, P.*, Prasad, A.*, Xu, Y.*, Poppleton, E., Williams, D., Zhang, F., Yan, H., Chiu, P. L., Stephanopoulos, N., & Šulc, P. (2022). Coarse-Grained Simulations for the Characterization and Optimization of Hybrid Protein-DNA Nanostructures. *ACS Nano*, 16, 14086–14096.

3.1 Abstract

We present here the combination of experimental and computational modeling tools for the design and characterization of protein-DNA hybrid nanostructures. Our work incorporates several features in the design of these nanostructures: (1) modeling of the protein-DNA linker identity and length; (2) optimizing the design of protein-DNA cages to account for mechanical stresses; and (3) probing the incorporation efficiency of protein-DNA conjugates into DNA nanostructures. The modeling tools were experimentally validated using structural characterization methods like cryo-TEM and AFM. Our method can be used for fitting low-resolution electron density maps when structural insights cannot be deciphered from experiments, as well as enable *in-silico* validation of nanostructured systems before their experimental realization. These tools will facilitate the design of complex hybrid protein-DNA nanostructures that seamlessly integrate the two different biomolecules.

3.2 Introduction

The field of DNA nanotechnology[154, 155] has made great strides in bionanotechnology over the past three decades. It relies on using the predictable Watson-Crick base pairing[156] of oligonucleotides in order to assemble them into desired 2D and 3D shapes. The nano-objects thus formed have been utilized for a variety of applications, including molecular storage,[157, 158] logic gate circuits,[159, 32, 160, 161] and drug delivery machines.[102, 162] Despite the tremendous progress the field has made in the past few decades, the limited chemical functionality of oligonucleotides has prevented DNA nanostructures from realizing many behaviors and interactions that proteins achieve in living organisms. One way to circumvent this limitation and construct more complex nanostructures—like “nano-robots” that can interact in a programmable way with biological systems—is to include functional protein units on a DNA scaffold. This approach has certain advantages compared with designing structures from amino acids alone: currently, *de novo* design[163] of protein nanostructures that rival the complexity of DNA origami is not possible, mainly because protein self-assembly lacks the predictability and orthogonal interactions inherent to nucleic acids. The most commonly used technique to design protein nanostructures revolves around the software Rosetta,[164] but this approach is still limited to experts in the field due to its complexity. Hence, despite impressive achievements in recent years, nanotechnology based on designed proteins has not yet achieved the level of versatility, structural complexity, and logic-gated control ability that has been developed for DNA nanotechnology.[165, 166] Methods for the design and characterization of protein-DNA hybrid nanostructures,[107, 143] however, still lag behind all-DNA structure design software like Tiamat,[69] CaDNAno,[68] Adenita,[66] and MagicDNA.[108] The design rules for hybrid nanomaterials have yet to be figured out completely, so most structures are designed in a heuristic and *ad hoc* fashion, and designer software and simulation methods integrating both DNA and protein nanostructures have only started to be developed recently.[66, 167] In this work, we aim to provide efficient tools for the design and verification of hybrid nanostructures in conjunction with experimental characterization.

In order to scale up protein-DNA nanostructure design and synthesis, basic building blocks and model systems still need to be designed and fully characterized. Designs utilizing DNA binding proteins have shown impressive and tantalizing results in this direction,[143] but they severely limit the protein functionality that can be incorporated into the design. For example, a given protein of interest would have to fused to a DNA-binding domain, which increases the molecular weight by a non-trivial amount, and could affect the presentation of the final protein if a flexible linker is used. Furthermore, DNA-binding proteins interact with oligonucleotides in a reversible manner, so even with dissociation constants in the nanomolar regime there could be protein detachment under the nanomolar concentrations used with many DNA origami nanostructures. We instead focus on chemically conjugating desired proteins to DNA in a site-specific manner, followed by hierarchical incorporation of these building blocks into DNA structures bearing complementary handles. Covalent conjugation is generally irreversible, and direct attachment to a DNA handle allows for a high degree of orthogonality due to Watson-Crick pairing. Furthermore, DNA strands can be attached to any point on a protein surface (by introducing a suitable reactive amino acid), whereas DNA-binding proteins must be fused to one of the two protein termini.

Understanding the design of these building blocks, and how they can best form hybrid nano-assemblies, requires us to have insight into various molecular parameters: 1) the ideal site for DNA conjugation on the protein; 2) the choice of chemical bioconjugation reaction used; 3) the flexibility and length of the small molecule linker between the DNA backbone and the protein surface. Once a protein-DNA building block has been synthesized, incorporating it into a hybrid system presents a distinct challenge. Often, the incorporation efficiency of the conjugate into the nanostructure is low, and it may not be immediately clear why this is the case. Possibilities include the misincorporation of complementary DNA handle sites, unintended steric and electrostatic clashes, or mechanical strain experienced by the hybrid nanostructure. To efficiently synthesize next-generation systems it will be critical to model the composite, *integrated* nanostructure, and take into account the properties of both the DNA and protein components, as well as the linkers that join them. In order to address these challenges and work towards design principles for these nanostructures, we used our recently developed protein-DNA hybrid model to characterize experimental results and optimize the design of two protein-DNA cage systems (Figure 11). In particular, we use a trimeric protein-DNA building block based on the KDPG aldolase building block reported by the Stephanopoulos lab in a previous report.[65]

The ability to construct defined three-dimensional cages with protein "walls" will yield applications in drug delivery (*e.g.*, "artificial viruses"), novel vaccine platforms, or synthesis of enzymatic nano-reactors. Towards this end, we first explored integrating the KDPG aldolase-DNA conjugate into a tetrahedral DNA origami cage using three complementary handles on each of the four faces of the cage (Figure 11A,B). We chose this system in order to: 1) gain structural insights into protein-DNA hybrids of large size (> 14,000 nucleotides) by both simulation and experiment; 2) simulate the chemical linker between the protein and DNA handle, and investigate the flexibility of the origami design; and 3) demonstrate the applicability of our methods in characterizing DNA nanostructures by cryogenic transmission electron microscopy (cryo-EM). Our modeling approaches are based on two tools that we recently developed: a coarse-grained model of DNA and proteins, called ANM-oxDNA,[168] and the OxView design tool,[148] originally developed for DNA nanostructures but since extended to support visualization and editing of protein-DNA nanostructures.[68] Additionally, we extended the online simulation server, oxDNA.org[75] to support ANM-oxDNA simulations and performed many of the simulations in this paper as part of that service, which we make freely available to the community for *in-silico* testing and verification of protein-DNA hybrid designs. We first designed a DNA origami tetrahedral cage with four available triangular void spaces for incorporating the KDPG aldolase-DNA conjugate (Figure 11A and B). This cage was characterized by cryo-EM to obtain an electron density map by single-particle reconstruction, and the density was fit with a mean structure obtained from coarse-grained simulations to verify that our models can correctly capture the hybrid nanostructure shape and structure.

In parallel, we applied our simulation model to a different assembly: a tetrahedral protein-DNA cage, with the aldolase capping a wireframe structure with six edges of four DNA helical turns each. We term this structure the Protein-DNA tetrahedron (PDTet) (Figure 11C). This structure formed with only modest yield in our initial publication reporting its design and synthesis.[65] We thus asked whether the simulation could provide insight into this low efficiency and suggest modifications to the structure design that would improve successful formation. Crucially, this system could also probe whether our computational model could be applied to hybrid nanostructures where, unlike the larger origami cage, the protein comprises a significant fraction of the assembly. We especially note that with PDTet, the final structure does not form in the absence of the protein vertex, and the homotrimeric protein-DNA conjugate is necessary for helping “fold” the triangular base into a wireframe cage. We simulated different PDTet structures with a varying number of unpaired polythymidine residues at the vertices of this nanostructure, and experimentally optimized the yield of structure formation (as visualized by AFM) by tuning the flexibility at these sites.

3.3 Results and Discussion

To probe the assembly of the hybrid protein-origami cage, we first synthesized the homotrimeric aldolase protein-DNA building block (PDNA) according to the previous report,[65] and as described in the methods section below. With this purified building block in hand, we proceeded to attach it to the four sides of the tetrahedral origami cage.

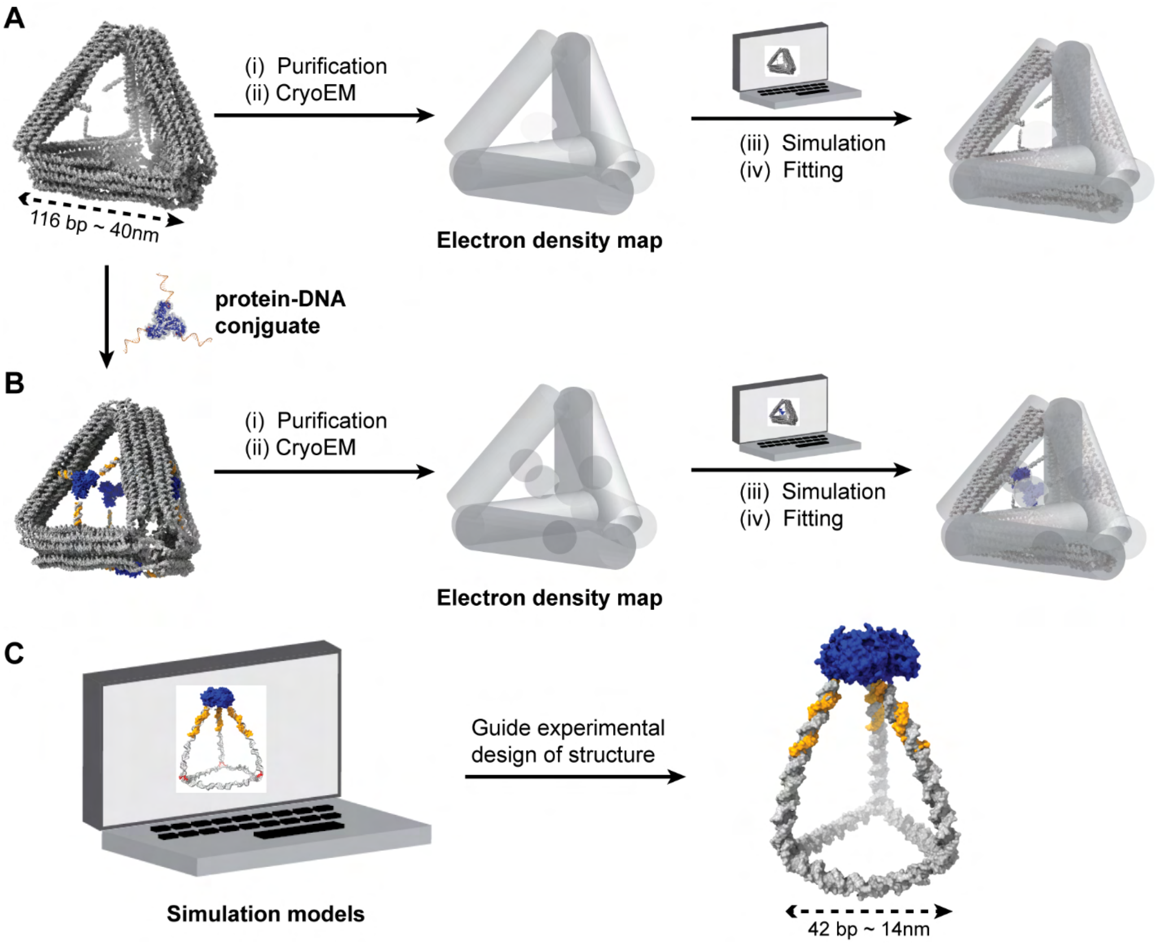


Figure 11. Using computational simulations to guide protein-DNA cage design. Elucidating the cryo-EM density map of the empty tetrahedral origami cage (A) and the origami with the trimeric protein incorporated (B), then using the density map to fit the simulated models to find the best correlation. C) Simulating a protein-DNA tetrahedral cage (PDTet) in order to predict the optimal design.

3.3.1 Design and synthesis of the tetrahedral origami cage with PDNA incorporated

The origami cage was designed using the software Cadnano,[68] with each arm consisting of 10 helices arranged on a honeycomb lattice. We opted for a tetrahedral geometry in order to avoid the preferred orientation problem that often hinders single-particle cryo-EM reconstruction.[169] The details of the origami design can be found in Figure 36. Each side was designed to have a length of 35 nm. The handles for the incorporation of the PDNA were positioned in such a way that one conjugate would bind onto each of the four faces of the tetrahedron, giving a maximum of four aldolase trimers per structure. In designing this nanostructure, we incorporated flexibility at the vertices of the tetrahedral cage by introducing polythymidine linkers (5 to 11 dT residues) to promote efficient formation. These samples were subjected to agarose gel electrophoresis (AGE), followed by excision of the desired band, elution of the origami, and verification of its structure by negative-stain EM (Figure 36 and section A.8). From the AGE analysis (Figure 36C), we concluded that the 11T version gave the best yields, so the rest of our studies were performed using this version of the cage. After visual confirmation by negative-stain EM, the purified origami cage was plunge-frozen (Section A.2,A.3) and characterized by cryo-EM (Figure 12A). Images were processed (section A.4 and Figure 39) using RELION 3.0 (Figure 12C). After characterizing the empty cages, we proceeded to probe the formation of the cage incorporating PDNA.

The PDNA-bearing cages (Figure 12B) were synthesized as described in section A.2. The samples were first characterized by negative-stain EM and then by cryo-EM (Figure 40) as before. The resulting reconstruction (Figure 12D, A.4) shows a clear electron density in the center of each face, supporting the incorporation of protein into the tetrahedral frame. These maps were later used to validate the ability of our coarse-grained model to correctly capture the experimentally determined structure.

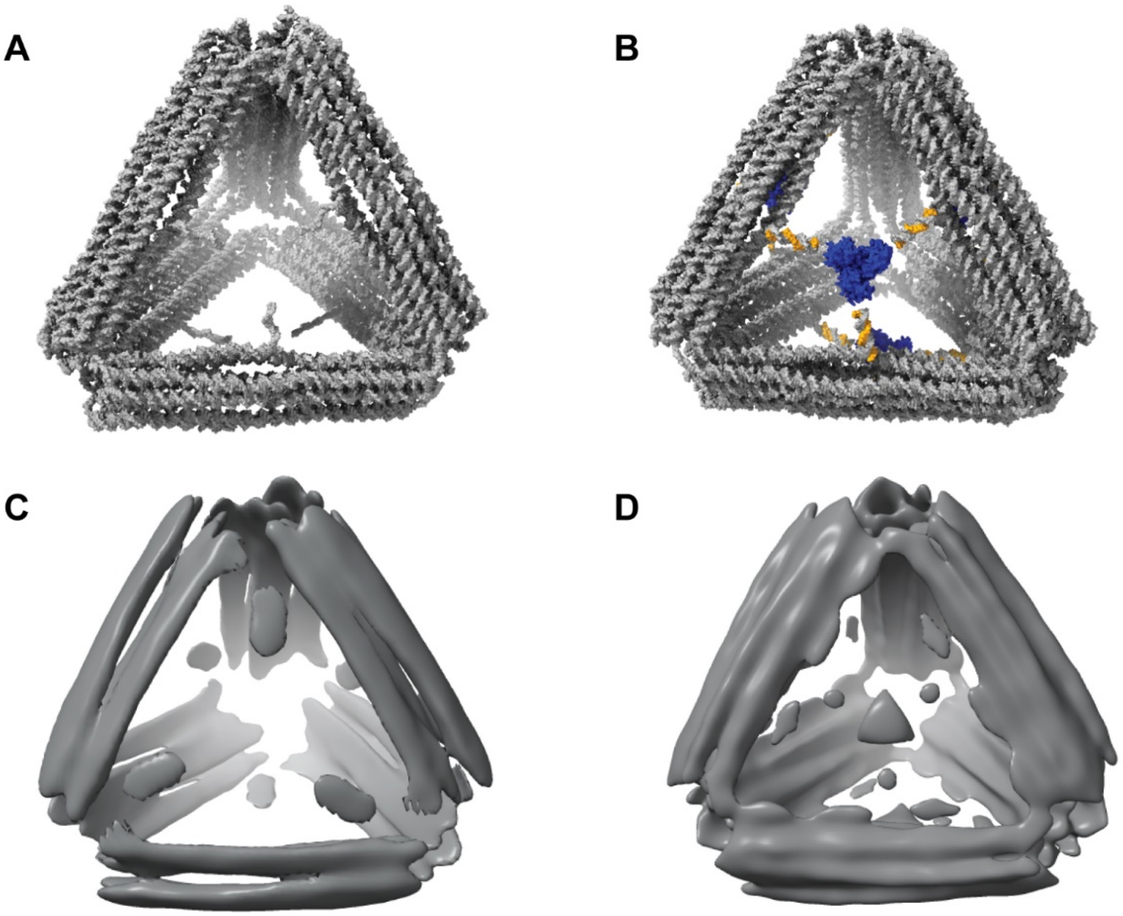


Figure 12. Cryo-EM reconstruction of tetrahedral origami cages. A) Schematic of the empty origami cage. B) Schematic of the origami cage incorporating PDNA. C) Cryo-EM reconstruction of (A) at 26 Å. D) Cryo-EM reconstruction of (B) at 28 Å.

3.3.2 Simulation Development for Protein-DNA Hybrid Systems

To characterize the cages with the PDNA incorporated, we developed a molecular simulation pipeline. Our ultimate goal is to provide tools and methods that aid in the nanostructure design and validation process *in-silico*, thus speeding up the development of novel designs, as well as offloading part of the process to computational modeling. Ideally, one would like to simulate and model protein-DNA hybrids at atomistic resolution. However, the system sizes (up to several tens of thousands of base pairs) and long timescales required for the characterization of such nanostructures present an enormous challenge. As a result, coarse-grained models have become increasingly more popular in nucleic acid nanotechnology. We used a recently introduced protein-DNA hybrid model[168], based on the oxDNA coarse-grained model of DNA.[116, 119, 114, 79] This model was previously used to study a wide range of DNA nanostructures and devices, and could reproduce their thermodynamics, mechanical properties, and kinetics.[116, 119, 114, 79] To incorporate proteins, the oxDNA model was extended with an Anisotropic Network Model[118] (ANM) that represents the polypeptides as beads connected by springs, parametrized to per residue fluctuation data—i.e. crystal B factors or a fully atomistic simulation trajectory—in order to capture the basic fluctuations and flexibility of the protein. Using the ANM-oxDNA model, we investigate how differences in protein incorporation and spacer length affected the mechanical properties of the DNA nanostructures and compared our results to those obtained experimentally.

3.3.3 Simulation of the Tetrahedral Origami Cages

The Cadnano design of the DNA origami was first converted into oxDNA using tacoxDNA[118] and further modified using our design tool oxView,[148] which was extended to also support protein representations for nanostructure design.[68] Modifications were made to include 11T spacers at the origami vertices, and to add handles for the incorporation of the PDNA. Five different simulation models were made by first parameterizing an ANM to the PDNA protein KDPG and subsequently adding the ANM to each model according to its PDNA incorporation. To finish the preparation of the simulation models the ANM was parameterized, the linker was introduced, and simulation topology relaxed as stated in the Methods section. Ten total simulation systems were prepared using each of the five models with different PDNA incorporation at 1 M salt concentration with two different temperatures: (1) 300 K (“high temperature”), and (2) 113 K (“low temperature”). Figure 13B shows the atomic model of the DBCO-NHS ester linker represented by a spring potential. Figures 3A, C-G show the mean structures for the different PDNA-bearing tetrahedral cages

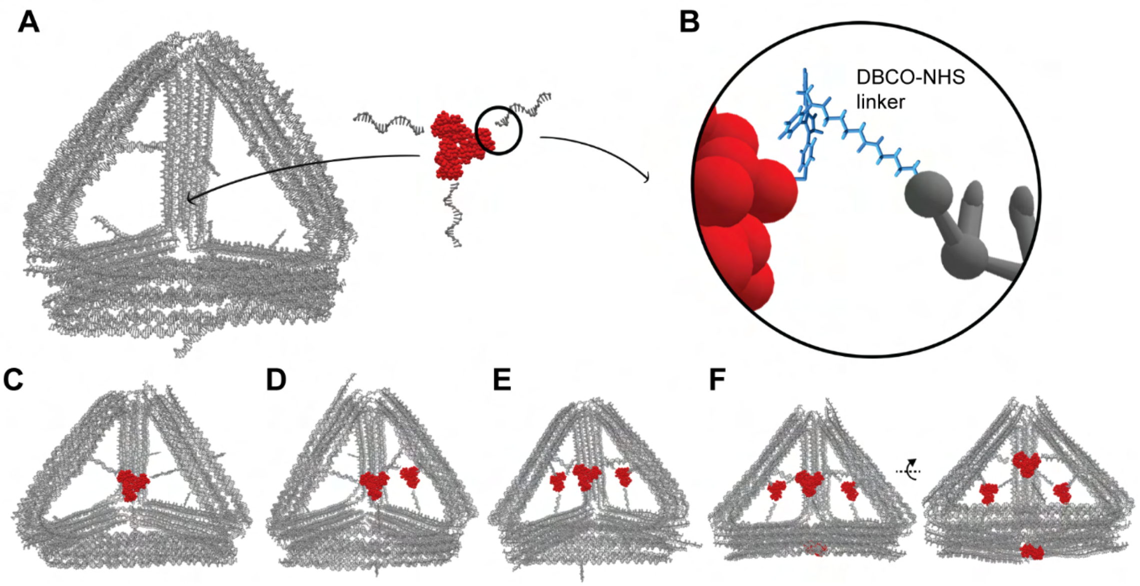


Figure 13. **A)** Schematic of PDNA incorporation in simulation models, using the empty cage mean structure at low temperature (113K). **B)** Atomic model of the DBCO-NHS ester linker, which is represented by a spring potential in the simulation. **C-F)** mean structures of origami bearing 1-4 PDNA building blocks, respectively, at low temperature conditions. Panel **(F)** includes a second view of the model with 4 PDNA incorporated so that the bottom protein is visible.

at low-temperature conditions. For our production simulations, each of the ten systems was simulated for 1 x 10[161] molecular dynamic simulation steps or approximately 3 μs.

3.3.4 Simulation Results for the Tetrahedral Protein Origami Cage

To characterize the differences between systems with different numbers (1-4) of protein trimers incorporated, we first analyzed the effect of adding PDNA on the origami cage flexibility, given that the protein trimer effectively crosslinks the three arms of the face it binds to. By comparing the root mean squared fluctuations (RMSF) of each model's identical DNA cage, we can see how the addition of the PDNA to the system affects the flexibility of the tetrahedral cage at the individual nucleotide level.

Figure 14 depicts the difference between the RMSF values for each pair of simulation

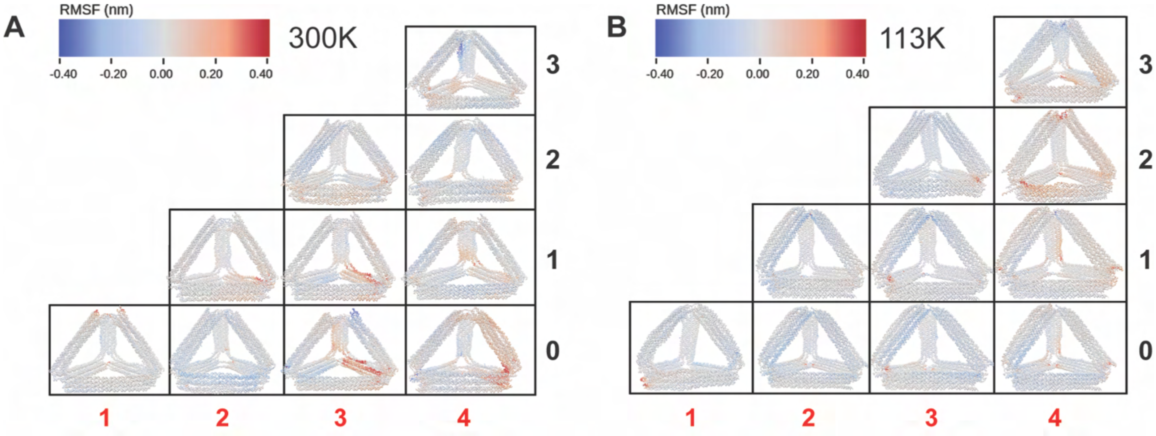


Figure 14. PDNA effect on cage flexibility. Difference in RMSF between the column model (red index denoting the number of PDNAs incorporated) and row model (black index denoting the number of PDNAs incorporated). RMSF differences are calculated as the column model RMSF minus the row model RMSF. Differences are displayed on the simulation mean structures of the row index with (**A**) being the relative differences in RMSF between all high temperature (300K) simulation models and (**B**) being the relative differences in RMSF between all low temperature mean structures (113K). The incorporated PDNA is not shown in the mean structures, as the RMSF was calculated only using the DNA component of the DNA-protein hybrid nanostructure.

models with differing number of PDNA incorporation, calculated per nucleotide as the column model's RMSF minus the row model's RMSF. Both the mean structure and RMSF of each model's DNA cage were averaged over the simulation trajectory using oxDNA analysis tools.[148] Higher (red) values indicate an increase in flexibility in the structure, while lower (blue) values indicate an increase in rigidity. In both conditions (high and low temperatures) the PDNA caused a clear decrease in the RMSF values of the arms with occupied handles. The decrease in RMSF corresponds to a local increase in rigidity, arising from the crosslinking by the PDNA (via the DNA handles) of the scaffold of the DNA origami. However, the addition of each subsequent PDNA introduces additional pulling forces on the adjacent faces, resulting in an increase of flexibility in arms that have *both* DNA handles bound by PDNA building blocks. This perhaps counterintuitive result can be explained by the pulling forces of the proteins disrupting some of the stacking interactions along the ten-helix bundle arm, thereby causing an increase in flexibility.

Beyond RMSF, differences in the mean structures suggest that the PDNA has a rigidifying effect on the face of the DNA cage to which it is attached. The mean structure for 4 PDNAs incorporated shows a significant change in the origami curvature, as evidenced by its straighter arms relative to all other mean structures. Figure 13A, F depict the mean structures of the bare origami and the four-PDNA mean structures at low-temperature conditions, where the largest difference in curvature can be observed.

Mean structures from each simulation trajectory were compared to the experimentally generated cryo-EM maps of the tetrahedral cage and PDNA-incorporated tetrahedral cage with the resulting fits shown in Figure 15. The mean structure files were stripped of their protein and DNA handles to avoid biasing the fitting, and the structures were exported from a coarse-grained nucleotide-level representation to a fully atomistic PDB format. Using UCSF Chimera,[170] the volume maps of the mean structures were generated from the atomic coordinates and fit to the experimental cryo-EM maps at 27 Å for both cryo-EM maps.

The generated density from the atomic model (translucent pink in Figure 15) closely fit the experimental maps (blue in Figure 15). The PDNA density in the cryo-EM map matched its position in simulation and confirmed the PDNA incorporation. These results corroborate that our coarse-grained model can indeed fit the cryo-EM map. We then analyzed the fittings to determine whether the slight differences in curvature between the cryo-EM maps could indicate the preferred level of incorporation of PDNA into the system.

Unfortunately, the resolution of the obtained cryo-EM map of the hybrid nanostructure was not sufficient to distinguish the difference between the models with different number of PDNA incorporated. The bulk assay, and low-resolution nature of the cryo-EM maps, combined with the subtle differences between models, made it impossible to determine a preference for PDNA incorporation from minor deviations in curvature. The correlation coefficients for fitting and associated images for both the filled and empty cryo-EM maps are available in section A.8.

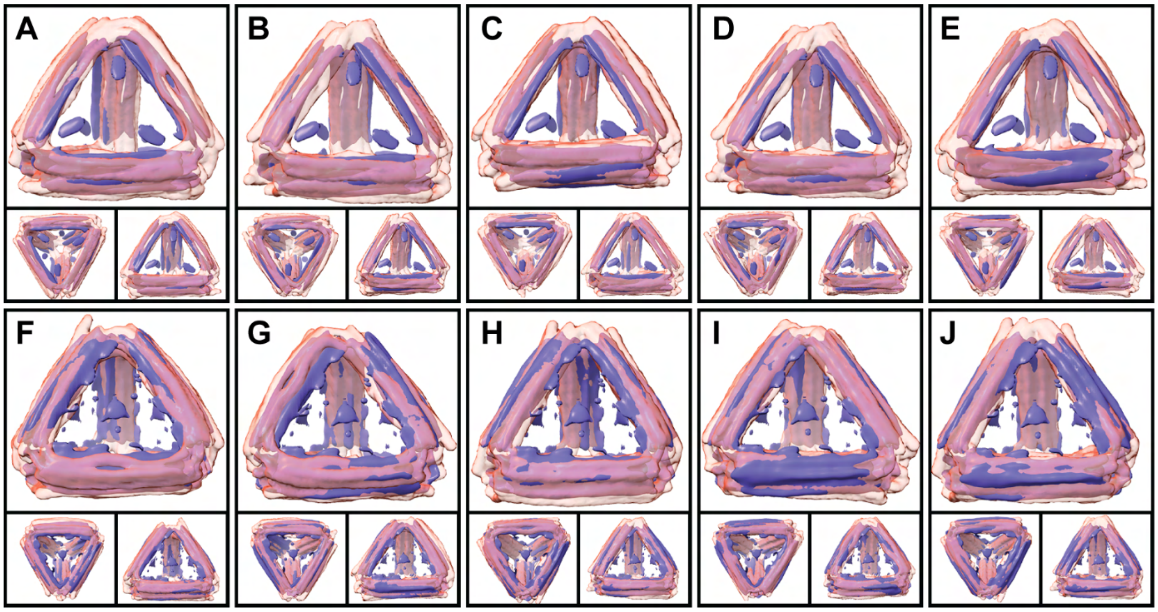


Figure 15. Fitting cryo-EM maps with mean structures obtained from the simulations at 300K. The densities generated from the mean atomic models at the same resolution as the cryo-EM map are shown in translucent pink and the cryo-EM map itself shown in purple. Each sub-figure depicts three views of the same fitting. **A)** 0 PDNA fit to empty cage. **B)** 1 PDNA fit to empty cage. **C)** 2 PDNA fit to empty cage. **D)** 3 PDNA fit to empty cage. **E)** 4 PDNA fit to empty cage. **F)** 0 PDNA fit to filled cage. **G)** 1 PDNA fit to filled cage. **H)** 2 PDNA fit to filled cage. **I)** 3 PDNA fit to filled cage. **J)** 4 PDNA fit to filled cage.

3.3.5 Fluorophore Assay for Determining the Number of Proteins per Cage

Because our reconstruction was performed with a small data set and was reconstructed with a tetrahedral symmetry, we wanted to probe PDNA incorporation in a cost-effective and more dispositive way than cryo-EM experiments. For this we carried out a fluorophore-based assay, wherein the PDNA was synthesized using a DNA handle with a FAM dye at the 5' end (Figure 16A) and the origami structure included a Cy5 dye. Then we proceeded to use fluorescence to elucidate the average number of proteins bound to the tetrahedral frame.

For this, we first obtained a calibration curve using known concentrations of the Cy5 handle strand and a FAM-labeled PDNA (Figure 6C). We made sure to perform these experiments using double-stranded DNA-dye conjugates to better match the experimental system,

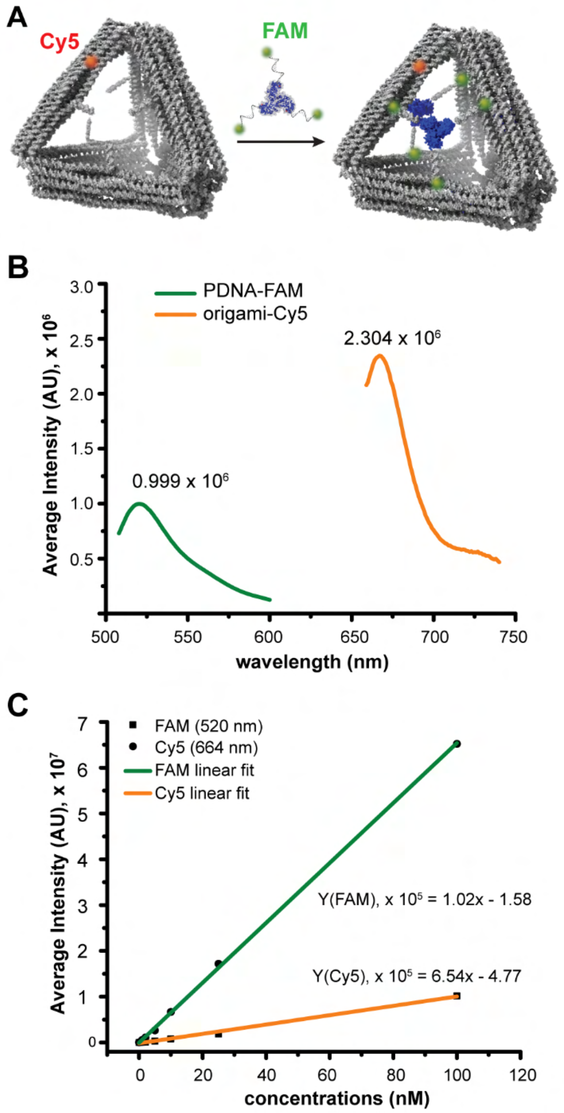


Figure 16. Fluorophore assay. A) Schematic showing the design of the assay. B) Fluorescence spectra of the PDNA-FAM and origami-Cy5. C) Calibration curve obtained from using known concentrations of double stranded DNA-dye conjugates (either FAM or Cy5).

where the protein is attached to the cage through hybridized handles. We then made our PDNA-incorporated tetrahedral cage as before and obtained emission values for this sample at the respective emission wavelengths (Figure 6B). We used the calibration curves to obtain the concentrations of the sample, yielding values of 3.59 nM for the tetrahedral frame, and 11.33 nM for PDNA, corresponding to ~78.9% protein incorporation (assuming four possible proteins), or ~3 proteins per cage on average.

We next turned to a different nanostructure, where PDNA is used as a structural building block. PDTet (Figure 11C) was chosen for this purpose for several reasons: 1) PDNA act as a critical structural building block to form a closed nano-structural cage; and 2) experimental characterization of the system can be realized using AFM, a technique less time and cost intensive than cryo-EM. We started out by simulating different PDTet structures (Figure 17) having varying number of poly-Thymidines at the vertices of the nanostructure.

3.3.6 Simulation-Based Predictions of PDTet Assembly Yield

The experimental yields of hybrid DNA-protein nanostructures rely on a number of factors, many of which are system-specific. For our PDTet cage system, a key concern is the flexibility of the DNA cage arms—i.e. their ability to bend upwards and form base pairs between the handles on the PDNA—and the resulting strain on the DNA cage when the structure is fully formed. By assessing these features, we aimed to predict the relative yields of each cage design as we introduced unpaired thymidine residues at the three vertices of the triangular DNA base structure.

Simulation files of the protein-DNA cage were prepared by first converting the Tiamat design of the origami cage with 3T spacers at the vertices into oxDNA via TacoxDNA.[118] Variations of this same cage with a different number of T spacers were created and relaxed (Methods) using oxView. All versions of the cage were simulated using molecular dynamics (1 x 10[160] steps; ~ 3 μs) at 300 K with 1 M salt concentration. Each cage was also simulated

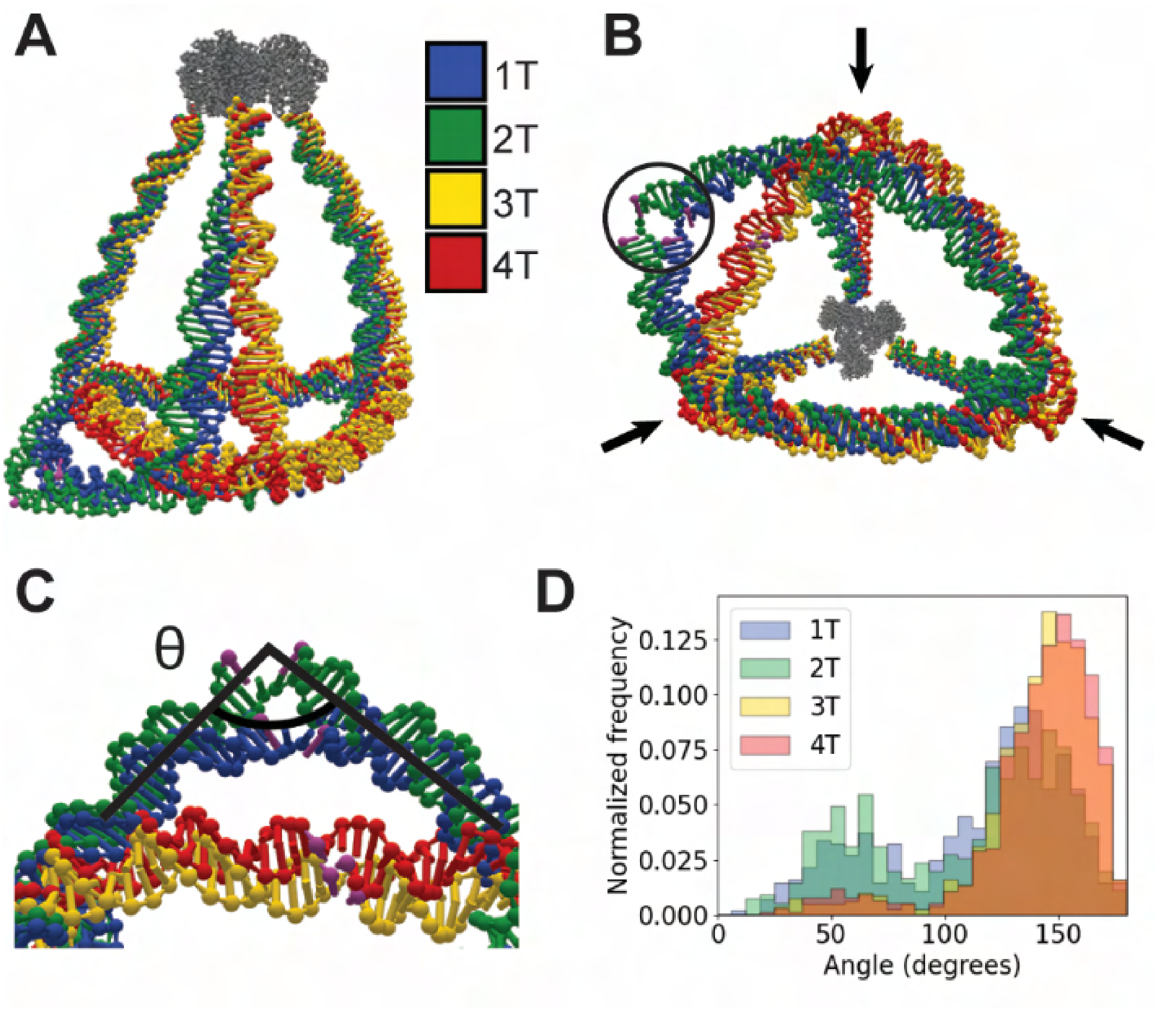


Figure 17. Simulating PDTet cages with varying linkers at the corners. (**A, B**) Two views of the aligned mean structures for cages with 1T, 2T, 3T, and 4T spacers, superimposed on one another. Arrows in (**B**) indicate the location of the thymidine spacers and the circle in (**B**) indicates the nick point for the 1T and 2T models. **C)** Depiction of angle measured across the nick point (Figure 41A). **D)** Angle distribution in (**C**) across all four simulation trajectories.

while attached to the same high temperature ANM representation of the aldolase protein used for the larger tetrahedron.

The aligned mean structures show significant differences in the DNA cage curvature depending on the number of T residues in the spacers in the vertices (Figure 17 A, B). At the site of the nick in the base of the DNA cage, the 1T and 2T structures show a bend in one arm (Figure 17 A-C), which is a mix of bent and straight arm configurations in the mean calculation. As more T residues are introduced into the spacers, the bent arm configurations are visited less often. Measuring the angle distribution between one side of the nicked helix to the other side of the nicked helix (Figure 17C) over the entire simulation trajectory illustrates the topological differences between varying the number of T spacers (Figure 17D). The configurations in Figure 17D with angles from 100-180° are considered "straight-arm" configurations, whereas angles 20-90° are considered "bent-arm" configurations. The key difference between the two populations is the ability of the nucleotides across the nick to maintain a coaxial stacking interaction. The disruption of this interaction is caused by mechanical strain induced on the base from the incorporation of the PDNA and the geometrical restrictions it imposes on the final hybrid structure.

Measuring the average energy of the two nucleotides before and after the nick in the DNA structure (A.7 Table 8), and comparing to simulations of the DNA structure without the protein—i.e. the triangular base with the single-stranded complementary arms (A.7 Table 5)—elucidates an energetic penalty stemming primarily from the disruption of the coaxial stacking and hydrogen bonding of the nucleotides at the nick in the bent configurations. The trend in energy from (A.7 Table 8) demonstrates that adding more dT nucleotides to the spacer mitigates this energetic penalty. However, the 3T model had more slightly more favorable coaxial and cross stacking interactions than the 4T model. Energy differences averaged over the T spacer nucleotides in each model were also examined. The same trend—i.e. lower average energy with increased length of T spacers—was observed, with the primary cause being a more favorable stacking interaction (A.7 Table 31). This trend was not observed in simulations of the triangular base alone (A.7 Table 32).

Overall, the aligned mean structures and energetic penalties incurred by the T spacer and nick nucleotides indicate that the strain in the structure decreases with increasing T spacer incorporation. From the above analysis, we can hypothesize that the 3T and 4T variants will have higher relative assembly yields, as they avoid the energetic penalties of the 1T and 2T variants. The slightly less favorable energy at the nick point (Figure 41A) of the 4T variant could indicate that this species will not form as well as the 3T.

To further explore the positional dependence (by individual arm) of T spacer incorporation, two sets of asymmetric cages were designed. One set of asymmetric systems was created by holding the arm across from the nick point constant as a 2T spacer and varying the T spacers in the other two arms of the DNA cage to have either 1T, 3T, or 4T spacers. Respectively these designs were named 1.1, 1.3, and 1.4. The second set of asymmetric systems was created by holding the two arms attached to the nick point constant at 2T spacers and varying the T spacer amount of the one arm across from the nick point to have either 1T, 3T, or 4T spacers. Respectively these designs were named 2.1, 2.3, and 2.4. All six asymmetric designs were relaxed, equilibrated, and simulated using the same exact methodology as the symmetric cages.

Figure 34 depicts the mean structures and accompanying nick point angle distributions for all six designs. As expected, the nick angle distribution is significantly affected by altering the two arms attached to the nick point and much less so for altering the arm across from the nick point. Raising the T spacer content of the two arms attached to the nick results in the cage visiting a bent configuration less often with a lower average energy at the nick due to more favorable stacking, cross stacking, and coaxial stacking interactions (Table 35). Alternatively, raising the T spacer content at the arm across from the nick point resulted in a marginally larger population of bent configurations and less favorable stacking, cross stacking, and coaxial stacking interactions at the nick (Table 39).

Assessing the average energy of the T spacers in the individual arms reveals some interesting trends. In designs that varied the two arms connected to the nick point, the left arm's (when viewed with the nick point in front and arm held constant in the back) average energy stays very similar across designs due to compensatory effects of a more favorable stacking interaction but less favorable cross stacking and coaxial stacking interactions (Table 37). The right arm's average energy has the same tradeoff of stacking vs. cross stacking and coaxial stacking interactions but has a significantly lower average energy due to a stronger stacking interaction (Table 38). The T spacers in the arm across from the nick (held constant at 2T spacers) showed a more favorable stacking interaction with increasing T spacers in the other 2 arms (Table 36). In designs that held the two arms connected to the nick constant, the left and right arms showed almost identical trends of a slightly more favorable stacking interactions with increasing T spacer number in the arm across from the nick (Table 40 and 41). The arm with the increased number of T spacers, however, showed no clear pattern in the average energy. Though not tested experimentally in this work, we would expect similar yields to their symmetric cage counterparts, in designs that hold the arm across from the nick point constant. Conversely, designs with two 2T arms and altering the arm across from the nick point may result in poorer yields compared to the symmetric 2T cage due to the slight promotion of bent configurations with increasing T spacer nucleotides in the altered arm.

3.3.7 Experimental Validation of T Spacer Effect on Protein-DNA Cage Assembly

Given the simulation predictions above, we sought to probe the effect of the dT linker on cage assembly via experiments. To form the cages, we first mixed the component oligonucleotide strands and assembled the triangular DNA structures with varying linkers (1T, 2T, 3T and 4T), without the PDNA attached to it, as described in Supporting Information section A.2. We characterized the system by native PAGE, extracted the band of interest, and confirmed that the triangular structure formed via AFM, as shown in Figures 42, 43, 44 and 45. We then added the aldolase PDNA to these triangular structures, annealed them as described in section A.2, and analyzed again by native PAGE (18B). The bands showed a significant shift from their open counterparts, indicating successful formation of the protein-DNA tetrahedral cages. To confirm nanostructure formation, we visualized the samples via AFM, examining both the crude samples (Figure 46, 47, 48 and 49) and the samples after gel extraction of the desired band (Figure 18C). Similar to the previous report,[65] we saw a varying fraction of cages that clearly corresponded to the four-turn tetrahedron with a protein vertex. We manually counted structures in the AFM images to determine the approximate yields of cage formation (Figure 50, 51, 52 and 53), with the results plotted in 18D. It was apparent that the 3T version formed the best with a yield of 67.8% (or 78.6% if we include particles that may be cages but could not be unambiguously assigned as such in the images). The 4T version was the next best at 58.6% (68.67%), followed by a significant drop in yield for the 2T version at 34.11% (45.29%) and 1T at 32.3% (43.08%). This result tracks well with the predictions from our simulation and suggests that coarse-grained modeling can indeed be used to probe the relative stability of various protein-DNA nanostructure designs. We suggest that this interplay between simulation and experiment will be especially critical for more complex protein-DNA nanostructures, and guide the choice of DNA sequence/length, linker design, site of protein-DNA conjugation, and choice of protein building block.

3.4 Conclusions

In this work, we successfully elucidated a low-resolution cryo-EM density map for the tetrahedral DNA origami cage, both with and without the PDNA attached to it. We simulated models ranging from zero to four proteins in the origami cage and fit our experimental data to this model. Although the correlation factors could not give us an exact insight into the incorporation efficiency, we could determine an average number of three proteins per cage using a fluorophore assay. We also simulated protein-DNA hybrid wireframe cages and found that the mechanical strain in the DNA wireframe nanostructures after the PDNA incorporation plays a critical role. Future hybrid nanostructure designs can be guided by our coarse-grained model, *e.g.*, by suggesting linker incorporation (such as unpaired thymidines), changing the DNA handle length, or selecting a different protein building block in order to minimize these strains. In this way, the simulations can reduce the number of designs that have to be tested experimentally, as well as reveal shortcomings of the initial design that might not be trivial to solve by simple trial-and-error experimental design.

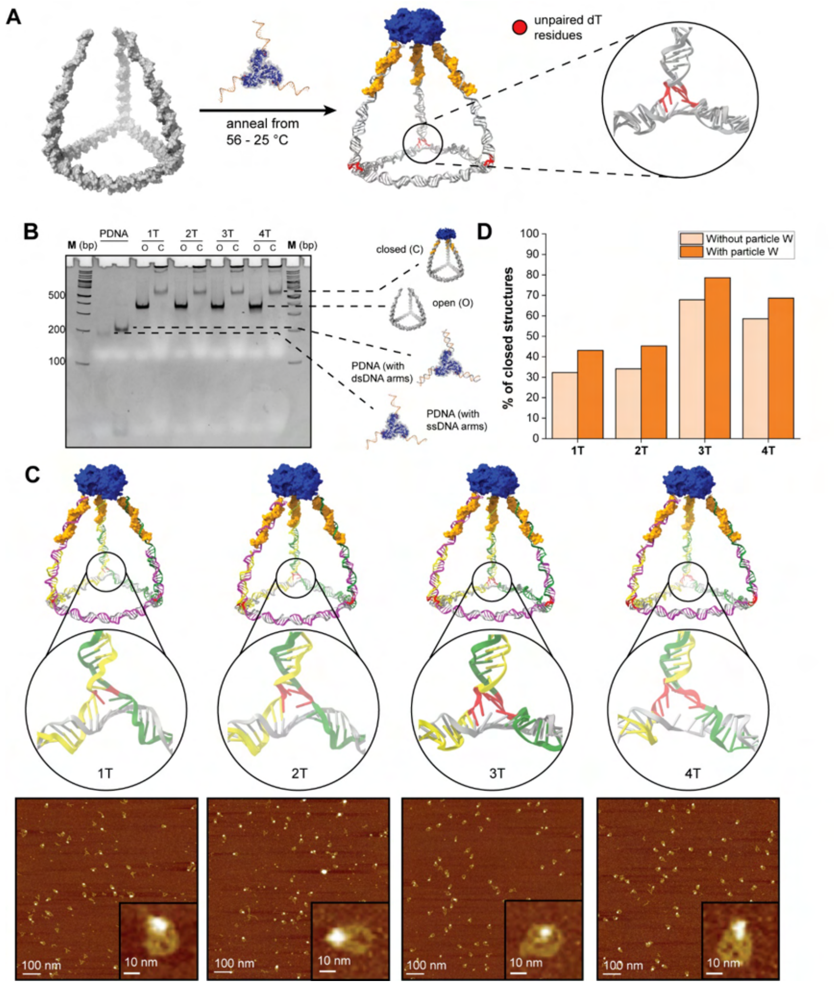


Figure 18. Experimental characterization of 4-turn protein-DNA tetrahedral cages. **A)** Schematic showing the formation of the hybrid nanostructure by the addition of the PDNA, including the location of unpaired poly-dT residues (shown in red). **B)** Native PAGE analysis of the open and closed tetrahedrons with varying number of poly-dT residues, alongside controls of the single and double stranded versions of the PDNA. The PDNA is hard to visualize when the handles are single-stranded (lane 2), so the complementary strand was added to improve the staining (lane 3). (Lane M: 100-bp dsDNA ladder). **C)** Schematic versions of the PDTet cages (1T, 2T, 3T and 4T) with their zoomed in detail showing the variations at the vertices. Below each image are AFM images to illustrate hybrid structures. **D)** Bar plot of the percentage of well-formed PDTet cages, as analyzed from AFM images (With and Without particles W means the with/without the inclusion of ambiguous particles as described in section A.3 and Figure 50, 51, 52, 53

Going forward, the computational model can be improved by more explicitly incorporating protein-DNA interactions (*e.g.,* electrostatics), as currently our tools rely solely on user-specified interactions, like a linker attaching the protein to the DNA. However, given the presence of cationic patches on many proteins, nonspecific electrostatic interactions with DNA could play a role in more complex designs. Proteins could also have unintended interactions with DNA through the presence of hydrophobic patches, which could for example interact with the blunt ends of helices, or nick sites in DNA duplexes. Furthermore, sequestering multiple proteins in close proximity on a DNA nano-scaffold could result in enhanced, nonspecific aggregation between them due to the high effective concentration. Nevertheless, despite these limitations, we have demonstrated a protein-DNA simulation tool that can guide the design of hybrid nanostructures, including the explicit incorporation of linker models. We foresee the use of this model in designing a range of protein-DNA nanosystems, especially when the protein plays a key structural role in the final assembly. The script to convert PDNA structures from oxView to PDB format is available at https://github.com/sulcgroup/oxdna_analysis_tools, along with tools to produce mean structures and quantify their flexibility. Furthermore, we have made the ANM-oxDNA model freely available on our public GPU webserver, oxDNA.org, to make this resource easily accessible to the bionanotechnology community. The interactive design that supports design of DNA and protein nanostructures, as well as setting up ANM-oxDNA simulations, has been implemented in oxView tool, available at oxview.org and https://github.com/sulcgroup/oxdna-viewer. The structures designed in this work are available in nanobase.org, an online repository of nanostructures.[75]

3.5 Methods

3.5.1 Synthesis of KDPG Aldolase Protein-DNA Building Blocks (PDNAs)

As previously reported,[65] the PDNA was synthesized by expressing and purifying KDPG aldolase protein containing the non-canonical amino acid 4-azidophenylalanine (azF) at position 54 (the E54(azF) mutant). The purified KDPG aldolase was conjugated to a 21-base single-stranded DNA (ssDNA) strand via strain-promoted azide-alkyne click chemistry. The dibenzylcyclooctyne-(DBCO) modified DNA was synthesized by conjugating an amine modified DNA strand with a DBCO-sulfo(NHS) ester conjugation as previously reported.[65] This conjugate was used for both the tetrahedral cages reported here. The same procedure was used for synthesizing the FAM-modified PDNA as well (described in the fluorophore assay section), where the strand used for conjugation to the protein was purchased from IDT having a FAM modification at the 5' end. The sequence of the strand attached to the protein is (5' to 3'):
(5AmMC6)TGAGTTCCGTCAGGTCTGCTC.

3.5.2 Parameterization of KDPG Aldolase Anisotropic Network Models

To approximately mimic the long-term dynamics of the protein for both sets of simulation conditions, two Anisotropic Network Models (ANMs) were parameterized. An ANM starts from a single configuration, usually the native state of the protein. Each ANM contains two free parameters: the cutoff distance (within which residues are connected by a harmonic potential) and the global force constant (used in all harmonic potentials). The low temp (113K) ANM was linearly fit to the crystallographic B factors of the trimer KDPG aldolase PDB file (1WA3) at a cutoff of 13 Å and a global force constant of 15.039 pN/Å. Comparison between the crystallographic B factors and the calculated B factors of the ANM match closely at 100 K (section A.6). Since B factors are collected at low temperature and electron microscopy model B factors have been shown to be meaningless[171], our high temp (300 K) ANM required high resolution simulation data. To this end, PDB file 1WA3 was used to generate a CHARMM model of our protein for a fully atomistic simulation. Our simulation system files were generated using CHARMM-GUI[172] with the CHARMM-36 forcefield[173] and TIP3P water molecules. After relaxation and equilibration, our system was simulated for 10 ns at 300 K using GROMACS[174]. The B factors of the C-Alpha carbons from our fully atomistic simulation were then used to parameterize our high temp ANM at a cutoff of 13 Å and a global force constant of 15.982 pN/ Å. The fully atomistic B factors from simulation and the calculated B factors of the high temp ANM fit well at 300 K (section A.6).

3.5.3 Linker Parameterization

The DBCO-based linkers used experimentally to conjugate the KDPG aldolase to DNA were previously modeled by fitting the length distribution observed in the fully atomistic simulation of the linker to a spring potential[167]. A molecular schematic of the linker and the spring potential parameters are included in section A.7.

3.5.4 Relaxation Procedure

First all linkers and ANMs were added to each simulation topology via the oxView design tool. Each system was then exported for simulation and subjected to a short Monte Carlo sampling (to remove any excluded volume clashes), then a MD simulation (1 x 10^9 steps) with external forces enforcing the designed DNA base pairing to relax each structure into the ANM-oxDNA forcefield. Another MD simulation (1 x 10^9 steps) was performed without the forces enforcing the DNA base pairing to allow each system to equilibrate.

3.6 Acknowledgements

This work was supported by NSF Grant no 1931487 and ONR Grant N000142012094. Nicholas Stephanopoulos acknowledges support from the National Science Foundation (DMR-BMAT CAREER award 1753387). Research reported in this publication was supported by The National Institute of General Medical Sciences of the National Institutes of Health under grant number DP2GM132931. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health. Po-Lin Chiu acknowledges the support from the US Department of Energy (DE-SC0002423).

3.7 Funding Sources

NSF Grant no 1931487. NSF Grant no 1753387. NIH Grant DP2GM132931. DOE grant DE-SC0002423. ONR Grant N000142012094

3.8 Notes

The authors declare no competing financial interest.

References

[32] Lulu Qian and Erik Winfree. “Scaling up digital circuit computation with DNA strand displacement cascades”. In: *Science* 332.6034 (2011), pp. 1196–1201.

[65] Yang Xu et al. “Tunable Nanoscale Cages from Self-Assembling DNA and Protein Building Blocks”. In: *ACS Nano* 13.3 (2019), pp. 3545–3554.

[66] Elisa de Llano et al. “Adenita: interactive 3D modelling and visualization of DNA nanostructures”. In: *Nucleic Acids Research* 1 (2020).

[68] Shawn M. Douglas et al. “Rapid prototyping of 3D DNA-origami shapes with caDNAno”. In: *Nucleic Acids Research* 37.15 (2009), pp. 5001–5006.

[69] Sean Williams et al. “Tiamat: a three-dimensional editing tool for complex DNA structures”. In: *International Workshop on DNA-Based Computers*. Springer. 2008, pp. 90–101.

[75] Erik Poppleton et al. “Nanobase. org: a repository for DNA and RNA nanostructures”. In: *Nucleic Acids Research* 50.D1 (2022), pp. D246–D252.

[79] Niranjan Srinivas et al. “On the biophysics and kinetics of toehold-mediated DNA strand displacement”. In: *Nucleic Acids Research* 41.22 (2013), pp. 10641–10658.

[102] Suping Li et al. “A DNA nanorobot functions as a cancer therapeutic in response to a molecular trigger in vivo”. In: *Nature biotechnology* 36.3 (2018), p. 258.

[107] Juan Jin et al. “Peptide assembly directed and quantified using megadalton DNA nanostructures”. In: *ACS Nano* 13.9 (2019), pp. 9927–9935.

[108] Chao-Min Huang et al. “Integrated computer-aided engineering and design for DNA assemblies”. In: *Nature Materials* 20.9 (2021), pp. 1264–1271.

[114] Benedict EK Snodin et al. “Introducing improved structural properties and salt dependence into a coarse-grained model of DNA”. In: *The Journal of chemical physics* 142.23 (2015), 06B613_1.

[116] Petr Šulc et al. “Sequence-dependent thermodynamics of a coarse-grained DNA model”. In: *Journal of Chemical Physics* 137.13 (2012), p. 5101.

[118] Antonio Suma et al. “TacoxDNA: A user-friendly web server for simulations of complex DNA structures, from single strands to origami”. In: *Journal of Computational Chemistry* 40.29 (2019), pp. 2586–2595.

[119] Jonathan P.K. Doye et al. “Coarse-graining DNA for simulations of DNA nanotechnology”. In: *Physical Chemistry Chemical Physics* 15.47 (2013), pp. 20395–20414.

[143] Alex Buchberger et al. “Hierarchical assembly of nucleic acid/coiled-coil peptide nanostructures”. In: *Journal of the American Chemical Society* 142.3 (2019), pp. 1406–1416.

[148] Erik Poppleton et al. “Design, optimization and analysis of large DNA and RNA nanostructures through interactive visualization, editing and molecular simulation”. In: *Nucleic Acids Research* 48.12 (2020), e72.

[154] Yamuna Krishnan and Nadrian C Seeman. “Introduction: nucleic acid nanotechnology”. In: *Chemical Reviews* 119.10 (2019), pp. 6271–6272.

[155] Fan Hong et al. “DNA origami: scaffolds for creating higher order structures”. In: *Chemical Reviews* 117.20 (2017), pp. 12584–12640.

[156] James D Watson and Francis HC Crick. “Molecular structure of nucleic acids: a structure for deoxyribose nucleic acid”. In: *Nature* 171.4356 (1953), pp. 737–738.

[157] Kaikai Chen et al. “Digital data storage using DNA nanostructures and solid-state nanopores”. In: *Nano Letters* 19.2 (2018), pp. 1210–1215.

[158] Kaikai Chen et al. “Nanopore-based DNA hard drives for rewritable and secure data storage”. In: *Nano Letters* 20.5 (2020), pp. 3754–3760.

[159] Tianqi Song et al. “Fast and compact DNA logic circuits based on single-stranded gates using strand-displacing polymerase”. In: *Nature Nanotechnology* 14.11 (2019), pp. 1075–1081.

[160] Georg Seelig et al. “Enzyme-free nucleic acid logic circuits”. In: *Science* 314.5805 (2006), pp. 1585–1588.

[161] Anupama J Thubagere et al. “Compiler-aided systematic construction of large-scale DNA strand displacement circuits using unpurified components”. In: *Nature Communications* 8.1 (2017), pp. 1–12.

[162] Qiao Jiang et al. “DNA origami as a carrier for circumvention of drug resistance”. In: *Journal of the American Chemical Society* 134.32 (2012), pp. 13396–13403.

[163] Po-Ssu Huang, Scott E Boyken, and David Baker. “The coming of age of de novo protein design”. In: *Nature* 537.7620 (2016), pp. 320–327.

[164] Rhiju Das and David Baker. “Macromolecular modeling with rosetta”. In: *Annual Review of Biochemistry* 77.1 (2008), pp. 363–382.

[165] Qinqin Hu et al. “DNA nanotechnology-enabled drug delivery systems”. In: *Chemical Reviews* 119.10 (2018), pp. 6459–6506.

[166] Shawn M Douglas, Ido Bachelet, and George M Church. “A logic-gated nanorobot for targeted transport of molecular payloads”. In: *Science* 335.6070 (2012), pp. 831–834.

[167] Wei Lu et al. “OpenAWSEM with Open3SPN2: A fast, flexible, and accessible framework for large-scale coarse-grained biomolecular simulations”. In: *PLoS Computational Biology* 17.2 (2021), e1008308.

[168] Jonah Procyk, Erik Poppleton, and Petr Šulc. “Coarse-grained nucleic acid–protein model for hybrid nanotechnology”. In: *Soft Matter* 17.13 (2021), pp. 3586–3593.

[169] Dmitry Lyumkis. “Challenges and opportunities in cryo-EM single-particle analysis”. In: *Journal of Biological Chemistry* 294.13 (2019), pp. 5181–5197.

[170] Eric F Pettersen et al. “UCSF Chimera—a visualization system for exploratory research and analysis”. In: *Journal of Computational Chemistry* 25.13 (2004), pp. 1605–1612.

[171] Alexander Wlodawer, Mi Li, and Zbigniew Dauter. “High-resolution cryo-EM maps and models: a crystallographer’s perspective”. In: *Structure* 25.10 (2017), pp. 1589–1597.

[172] Sunhwan Jo et al. “CHARMM-GUI: a web-based graphical user interface for CHARMM”. In: *Journal of Computational Chemistry* 29.11 (2008), pp. 1859–1865.

[173] Jing Huang and Alexander D MacKerell Jr. “CHARMM36 all-atom additive protein force field: Validation based on comparison to NMR data”. In: *Journal of Computational Chemistry* 34.25 (2013), pp. 2135–2145.

[174] Mark James Abraham et al. “GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers”. In: *SoftwareX* 1 (2015), pp. 19–25.

Chapter 4

THROMBIN APTAMER DESIGN

This chapter appears in Di Gioacchino A.$^+$, **Procyk J.**$^+$, Molari M, Schreck, J. S., Zhou, Y., Liu, Y., Monasson, R., Cocco, S., & Šulc, P. (2022) Generative and interpretable machine learning for aptamer design and analysis of in vitro sequence selection. *Plos Computational Biology*, 18(9), e1010561.

4.1 Abstract

Selection protocols such as SELEX, where molecules are selected over multiple rounds for their ability to bind to a target of interest, are popular methods for obtaining binders for diagnostic and therapeutic purposes. We show that Restricted Boltzmann Machines (RBMs), an unsupervised two-layer neural network architecture, can successfully be trained on sequence ensembles from single rounds of SELEX experiments for thrombin aptamers. RBMs assign scores to sequences that can be directly related to their fitnesses estimated through experimental enrichment ratios. Hence, RBMs trained from sequence data at a given round can be used to predict the effects of selection at later rounds. Moreover, the parameters of the trained RBMs are interpretable and identify functional features contributing most to sequence fitness. To exploit the generative capabilities of RBMs, we introduce two different training protocols: one taking into account sequence counts, capable of identifying the few best binders, and another based on unique sequences only, generating more diverse binders. We then use RBMs model to generate novel aptamers with putative disruptive mutations or good binding properties, and validate the generated sequences with gel shift assay experiments. Finally, we compare the RBM's performance with different supervised

learning approaches that include random forests and several deep neural network architectures.

4.2 Introduction

Discovery and design of molecules that can specifically bind a given target molecule is a key problem in diagnostics, therapeutics and molecular biology in general. Multiple different experimental approaches exist to select specific molecular target binder such as antibodies, short peptides, proteins or small molecules. Single stranded oligonucleotides (DNA or RNA) have also been shown to be able to specifically bind with high affinity to a plethora of various targets, including small metabolites, proteins, nucleic acids, viruses, exosomes, and cells of specific tissue [175, 176, 177, 18, 178, 179, 180, 181, 182, 183], showing promise for applications that range from diagnostics to targeted disease therapy [184]. These short oligonucleotides, called aptamers, are selected from an initial pool of sequences by a procedure known as Systematic Evolution of Ligands by Exponential Enrichment (SELEX) [185, 186]. This method consists of multiple rounds of selection, where aptamers that bind strongly enough to the protein target are selected and amplified for the next round, until few strong binders are obtained. The advantages of using DNA or RNA include low cost of synthesising these molecules and relative ease of their manipulation in the laboratory setting as opposed to other selection methods such as peptide or antibody selection [187, 188]. Oligonucleotides can be denatured and refolded many times, allowing for multiple selection rounds. On the other hand, as they are composed of four possible types of bases (A, C, G and T/U), they do not offer such chemical diversity as antibodies. Thus, the range of targets that aptamers can be selected to bind strongly to is limited to some extent. However, chemical modifications of the nucleic bases can increase the chemical space of the aptamers and provide diverse sequence libraries from which strong binders can be selected against a variety of targets [189].

With the advance of next generation sequencing and high-throughput biological and molecular dataset production, various machine learning methods have been used to process biological sequences datasets, with applications including classifications, binding prediction, and molecular design [190]. While a significant improvement has recently been achieved in using deep learning for protein or RNA structure predictions [83, 191], predictions of binding interactions and *de novo* design of molecular binders remain outstanding significant challenges. So far, it is primarily the prediction of interaction between a small molecule ligand and a target protein that has received attention from the machine learning community, as such approaches are at the basis of the drug screening pipeline [192]. Motif-finding and clustering-based methods, combined with secondary structure prediction tools, have been previously developed for processing SELEX datasets [193, 194, 195, 195, 196, 197]. Currently, the SELEX dataset processing typically involves clustering and identifying a common motif in aligned sequences and then selecting representative aptamers from the last round of selection and verifying their binding affinity to the target.

A challenging task in the analysis of SELEX experiments is the quantification of the aptamer fitness, which determines the sequence landscape evolution at each selection round. Several approaches have been introduced in the past, based on *in silico* molecular dynamics simulations [198, 199], on clustering in sequence space together with enrichment measurements [200], and on additional, direct fitness estimation experiments [201]. These methods proved useful to estimate the fitness of a limited number of selected sequences or of large classes of similar sequences, but they seem unable to assign in a reliable way a fitness score to each molecule observed in final rounds of SELEX.

Over the last decade, deep neural networks (DNN) have become a popular machine learning tool in many areas, such as image recognition or natural language processing, and are now increasingly applied in chemical and biological data processing workflow [202, 203, 204, 205]. However, training DNNs typically requires large datasets, which can be challenging and expensive to obtain from biological experiments. DNNs have many free parameters, which makes it difficult to identify and interpret particular features of the molecule that are attributed to its ability to bind a given target. The presence of errors in the sequence dataset, coming e.g. from experimental error in affinity measurements or sequencing errors, adds further difficulties to training as well as to interpretability. Machine-learning methods for sequence ensembles include inverse models from statistical physics, such as direct coupling analysis (DCA) methods [206], which have been previously successfully used to infer native contacts and guide folding of RNA and proteins based on homologous sequence alignment [207, 94], as well as to generate functional enzymes based on functional protein alignments [208] and protein recognizing RNA [209]. They infer parameters of maximum-entropy models, which are fixed by the requirement that the conservation of single residues and pairs of residues given by the model match the values observed in the sequence alignment. More recently, Restricted Boltzmann Machine (RBM) architectures, a neural network with a bipartite graph structure, have been successfully applied as a generative model for protein domain sequences [97], as well as a predictor of peptides that will be presented on Major Histocompatibility Complexes [99]. They present an intermediate level of complexity between the direct coupling models and DNNs, as they can be trained to recognize multi-residue coupling as opposed to pairwise interactions, but due to limited number of weights between the two neuron layers, the parameters can still be interpreted and rationalized.

Here, we apply RBM models to a set of DNA sequences obtained from the prior experimental work of some of us that used SELEX method to obtain thrombin aptamers [210] (Fig. 19). We show that the sequence likelihood assigned by the RBM can be directly related to the fitness of that sequence in the experimental selection. Moreover an RBM model that is trained on an earlier round of the selection is able to predict fitness of sequences in the next rounds not seen during the training, showing remarkable generalization capabilities. We further show that we can identify the sequence motifs conferring large likelihood to an aptamer sequence and that RBM's hidden unit input can be used to cluster sequences. We show the capability of the RBM to predict binding affinity and generate new monovalent aptamers, which are good binders to one of the two thrombin binding sites, by gel shift assays. We investigate how taking into account the individual sequence counts from the experiment in the training data changes the properties of the inferred RBM model. Lastly, we also ex-

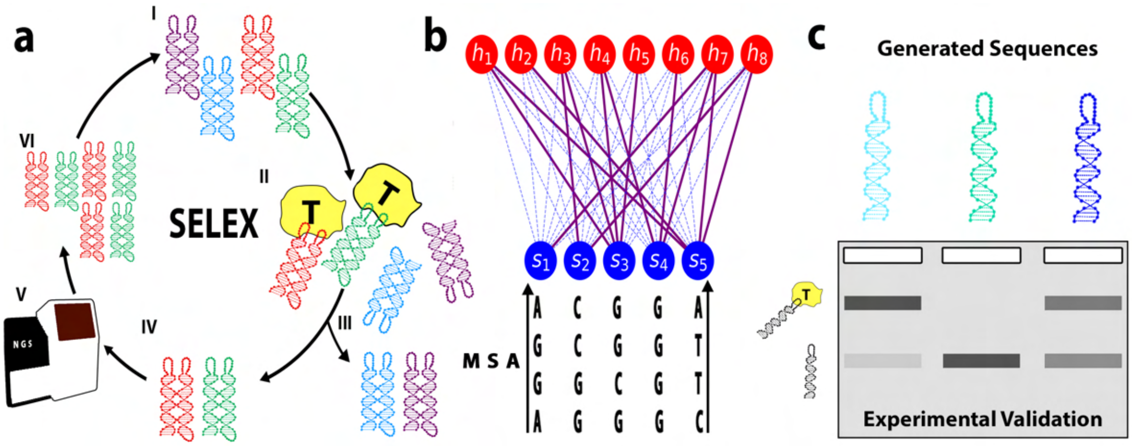


Figure 19. **Schematic view of the SELEX experiment and the RBM-based analysis.** **a:** The SELEX procedure used to obtain DNA aptamers that bind to thrombin consists of the following steps: I) We start with an initial library of DNA sequences. II) DNA aptamers compete with each other to bind to thrombin. III) Sequences that are unbound (or bound too weakly) are washed away. IV) Remaining bound sequences dissociate after the sample is heated up. V) Binding sequences are sequenced. VI) Using polymerase chain reaction (PCR), multiple copies are made of the remaining sequences, resulting into a new library of aptamers for the next round of selection. **b:** The sequenced aptamers from respective rounds of the SELEX protocol are used to train the parameters of the Restricted Boltzmann Machine model. In this unsupervised neural network architecture, a layer of visible units carry the aptamer sequence, while the layer of hidden units extract representations. The weighted connections between the two layers are learned through maximization of the log-likelihood of the sequences obtained through SELEX. **c:** Single loop sequences generated using the Restricted Boltzmann Machine model are experimentally validated using gel assays.

plore several supervised learning approaches that include random forest and various DNN architectures, but find them difficult to train and with poor generalization performance on our dataset.

4.3 Results

4.3.1 Dataset Obtained from SELEX Procedure

In a prior work [210], some of us used the SELEX method to obtain a bivalent DNA nanostructure that binds to a thrombin protein. In this DNA SELEX procedure, an initial library of about 10^{15} unique DNA sequences with all about the same length were exposed to the target tethered to a surface. The non-binding sequences were then washed away, while the binding sequences were collected (and optionally also sequenced). After amplification with PCR they served as the sequence library for the next cycle of SELEX. Cycles were repeated until binders of the desired binding affinity were found. The washing intensity was increased in later rounds to obtain stronger binders. In the particular experimental dataset used in Ref. [210], the SELEX procedure was performed on a DNA nanotile (Fig. 19), consisting of a joined-double helix region with two loops of 20 nucleotides each. While the double-helix nanotile structure was conserved across all DNA structures, the two respective loops were variable, starting from the initial random library. The SELEX procedure is schematically shown in Fig. 19 and consisted of eight selection rounds. The binding molecules were sequenced in rounds 5 (891959 sequences out of which 891914 unique), 6 (736436 sequences out of which 735974 unique), 7 (750926 sequences out of which 744597 unique) and 8 (725431 sequences out of which 719413 unique), and form the datasets we use here for training our models.

For each round, our dataset includes the sequence of the two (left and right) respective variable loop regions of the DNA nanotile, as well as the number of counts of the two-loop sequence, corresponding to the number of times it was sequenced in the experiment. In typical SELEX protocols, the sequences with the largest number of counts in the last rounds are considered the best binders.

4.3.2 Restricted Boltzmann Machine Model

We use a Restricted Boltzmann Machine (RBM) to learn the probability distribution over the set of aptamers based on the sequences collected through the SELEX procedure. An RBM is a probabilistic model, represented by a bipartite graph consisting of L "visible" and M "hidden" units (shown schematically in Fig. 19b). It assigns a probability $p(\boldsymbol{s}, \boldsymbol{h})$ to a system state, given by two parts: the configuration of visible units, $\boldsymbol{s} = (s_1, \dots, s_L)$, where s_i = A, C, G or T are the nucleotides on site i along the aptamer sequence, and the configuration of the hidden units, $\boldsymbol{h} = (h_1, \ldots, h_M)$, meant to extract latent factors of variation in the visible configurations. The likelihood of a sequence $\boldsymbol{s}$ is formally obtained by marginalizing over all possible latent configurations (not observed in the data), $p(\boldsymbol{s}) = \int d\boldsymbol{h}\, p(\boldsymbol{s}, \boldsymbol{h})$. The number L of visible units can be set to 40 to model full two-loop sequences or restricted to 20 to describe each loop independently. These two possibilities will be referred to as, respectively, D (Double loop) and S (Single loop) in the following.

Training a RBM consists in finding the parameters (in particular, the couplings between the layers) so that the log-likelihood of the observed data,

$$\mathcal{L} = \sum_{\boldsymbol{s} \in \text{round r}} \log p(\boldsymbol{s}) \ , \tag{4.1}$$

is maximized. Here the sum over $\boldsymbol{s}$ is over the sequences observed at a fixed selection round, say, r, of the SELEX experiment. Each sequence may therefore appear multiple times, depending on the number of its counts. We will denote this model with C (Count). An alternative is to include in the sum in Eq. (4.1) unique sequences only. The resulting model, labelled with U (Unique), has different properties, which we will discussed below.

The maximization of $\mathcal{L}$ is a computationally difficult problem, but several effective techniques to obtain good parameter values have been developed, for instance contrastive divergence [91] and persistent contrastive divergence [211]. As described in Methods Sec. 4.6.2, we train, following Ref. [97], the RBM using persistent contrastive divergence and using double Rectified Linear hidden units, with a L_1^2 regularization scheme. This regularization favors sparse weights, and enhances interpretability of the trained model.

4.3.3 RBM's Log-Likelihood is an Accurate Predictor of the Aptamer's Fitness

Fig. 20a shows the distributions of log-likelihoods of sequences collected at SELEX rounds $r = 5$ to 8, estimated with an RBM trained on double-loop aptamer sequences with counts measured at round 6 (RBM-DC, see Sec. B.3). At round 5 three peaks are apparent.

The logos of the sequences in each peak are shown in Fig. 20b. The peak at low log-likelihoods is characterized by highly variable sequences, weakly enriched in C, G nucleotides. The peak at intermediate values correspond to sequences with a structured loop (the left one, for most sequences), including a G-quadruplex motif. In the high log-likelihood peak a similar G-quadruplex motif appears on both left and right loops (for more details, see also Sec. 4.3.4). From round 6 to 8 the peaks at low and intermediate log-likelihood values are progressively depleted, and the peak at high log-likelihood gets more and more populated. This enrichment strongly suggests a positive correlation between the score assigned by the RBM and the fitness.

In a population genetic framework, the fraction q of aptamers with sequence $\mathbf{s}$ changes from round $r-1$ to round r according to

$$q_r(\mathbf{s}) = \frac{e^{\alpha_{r-1} F(\mathbf{s})}}{\langle e^{\alpha_{r-1} F(\mathbf{s}')} \rangle_{\mathbf{s}' \in r-1}} \, q_{r-1}(\mathbf{s}), \tag{4.2}$$

where $\langle O(\mathbf{s}') \rangle_{\mathbf{s}' \in r-1} = \sum_{\mathbf{s}'} q_{r-1}(\mathbf{s}') O(\mathbf{s}')$ denotes the average of the observable O over the distribution of sequences at round $r-1$. The fitness $F(\mathbf{s})$ encompasses the capability of an aptamer $\boldsymbol{s}$ of binding its target, as well as other chemical properties, such as its affinity to PCR amplification. Parameter α_{r-1} represents the selection strength from round $r-1$ to r, which can be tuned in practice e.g. by varying with the intensity of washing in SELEX selection.

According to Eq. (4.2), formally valid for an infinite-size population only, the fitness $\alpha_{r-1}\, F(\mathbf{s})$ is, up to a sequence-independent additive constant, equal to the logarithm of the enrichment ratio $\mathcal{E}_r(\mathbf{s}) = C_r(\mathbf{s})/C_{r-1}(\mathbf{s})$, where $C_r(\mathbf{s})$ is the number of counts of sequence s at round r. However, the extreme subsampling of sequences at each round in our dataset prevents us from using empirical enrichment ratios $\mathcal{E}$ to estimate the fitnesses, and their correlation with log-likelihoods, see Fig. 72. For instance, only $f_{\text{shared}} = 0.5\%$ of the sequences observed in round 7 or round 8 are present in both rounds, and among these sequences, about $f_1 = 70\%$ have count $C = 1$ in both rounds. In earlier rounds, e.g. 5 and 6, the situations is even worse, with fractions $f_{\text{shared}} = 0.01\%$ and $f_1 = 93\%$.

To obtain more reliable enrichment ratios we gather all sequences s having similar log-likelihoods $\log p(\mathrm{s})$, and introduce their cumulative number of counts, $C(\ell, r)$. More precisely, $C(\ell, r)$ is defined as the number of counts in the ℓ^{th} bin of the histogram of log-likelihoods in Fig. 20a. We then define the effective enrichment ratio of bin ℓ through $\mathcal{E}_r(\ell) = C_r(\ell)/C_{r-1}(\ell)$. Fig. 20c shows the scatter plots of the enrichment log-ratios $\log \mathcal{E}_r(\ell)$ vs. the log-likelihoods ℓ, for rounds $r = 6, 7, 8$. Very strong correlations are observed, with coefficients of determination $R^2 = 0.99$, 0.83 and 0.66 and slopes 0.16, 0.07, 0.01 for, respectively, the pairs of rounds 5 → 6, 6 → 7, and 7 → 8. The smaller values of the slopes of the linear regressions at later rounds suggests that the effective selection strength α_{r-1} appearing in Eq. (4.2) is weaker in the last SELEX rounds than in the previous ones. This interpretation is supported by the fact that the 10 different single-loop aptamers with largest count numbers at round 8 do not increase exponentially in the last rounds considered here, as shown in Fig. 68.

The linear relationship between the RBM log-likelihood $\log p(\boldsymbol{s})$ and the sequence fitness $F(\boldsymbol{s})$ suggests an alternative way to estimate the selection strengths α_r. Fisher's fundamental theorem (see for instance [212] for a review) postulates that the selection strength can be estimated through the ratio of the increase of the average fitness and of the its variance, $\alpha_{r-1} = (\langle F \rangle_r - \langle F \rangle_{r-1})/\mathrm{var}(F)$. We compute these Fisher's ratios using $\log p$ as a proxy for F to estimate the selection strengths at the various rounds. Results are shown in the inset of Fig. 20c, and agree with those obtained directly from the slopes of the linear regressions. The precise relation between the fitness and the log-likelihood is further examined in Discussion section.

4.3.4 The Log-Likelihoods of the Aptamers can be Explained by the Additive Contributions of their Left and Right Loops

To examine the cooperative binding of the left and right loops of the aptamer nanostructure at a given round of SELEX, we have trained RBM models on the 20 nucleotide-long

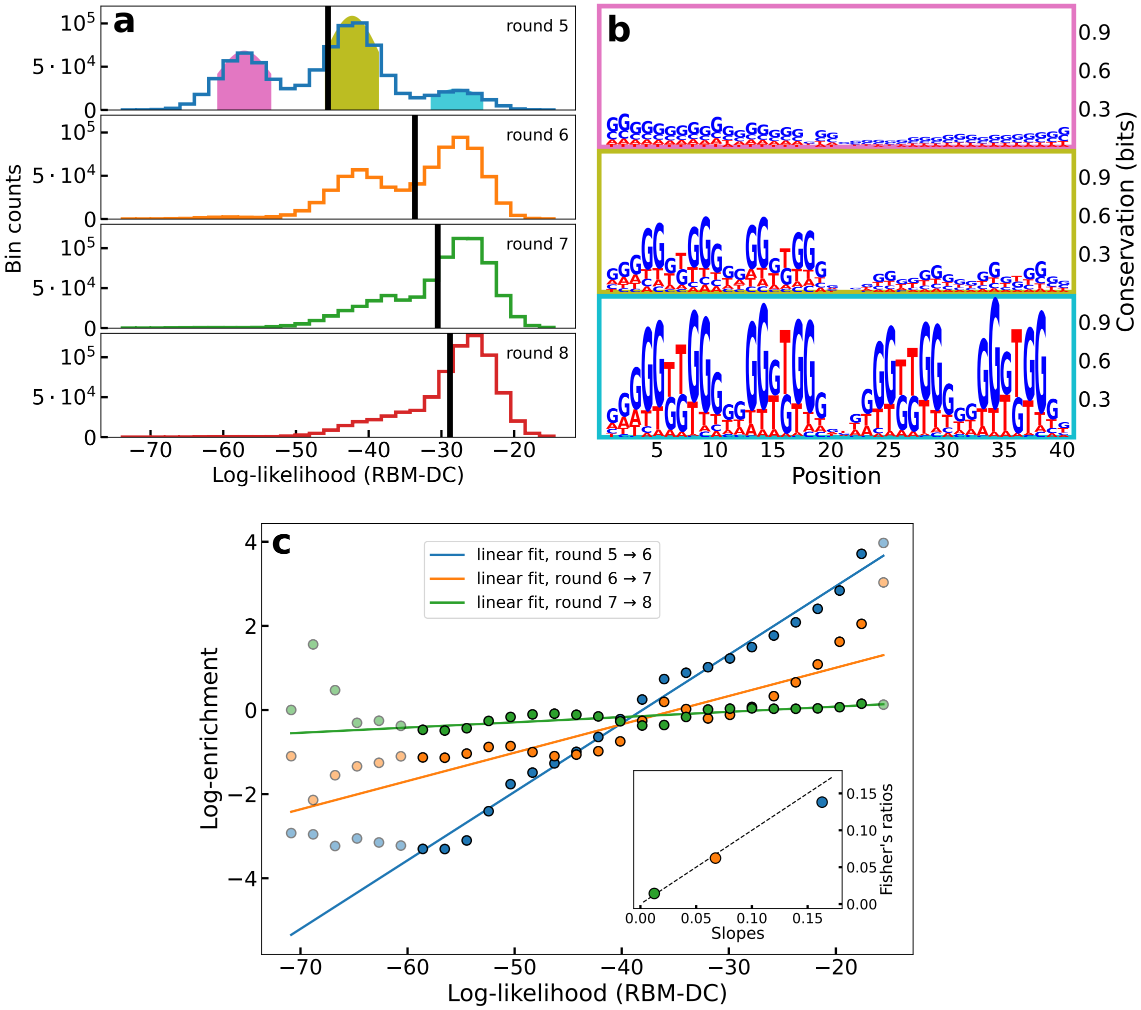


Figure 20. **The RBM log-likelihood is strongly correlated to sequence fitness.**
a: Histograms of the log-likelihood of all sequences in the dataset at different rounds, obtained through RBM-DC trained on the sequences from round 6. The black line denotes the average log-likelihood. **b**: Logos of the sequences in each colored-shaded peak of the log-likelihood observed at round 5. **c**: Enrichment log-ratios $\log \mathcal{E}$ vs. log-likelihoods $\log p$ averages over the sequences in each bin of the histograms of panel a. The three sets of points corresponds to rounds $r = 6, 7, 8$. Linear fit are estimated from points with log-likelihood in the interval $(-60, -17)$ (not shaded in the plot) only, to exclude under-sampled bins cumulatively representing 0.5%, 0.3%, and 0.3% of the sequences, respectively at rounds 6, 7 and 8. Inset: scatter plot of the slopes of the linear fits (x-axis) and of the log-likelihood Fisher ratios (y-axis); linear dashed line: $y = x$.

single loops only. In practice, RBM-SC trained on all left (L) loop subsequences, on all right (R) loop subsequences, or on both of them show very similar properties (Fig. 69), and we hereafter report results with the latter model. We show in Fig. 21a the log-likelihoods of the L and R loops for all aptamers at round 5. We observe the presence of four peaks in the joint distribution, corresponding to all possible combinations of the two peaks at, respectively, low ($\simeq \mathcal{L}_-$) and high ($\simeq \mathcal{L}_+$) log-likelihoods present in the marginal distributions for L or R loops.

As shown in Fig. 21b, aptamer sequences previously characterized as having low (in pink), intermediate (in olive) and high (in turquoise) log-likelihoods, see Fig. 20a, occupy the four corners of the joint-distribution plot. Therefore, high-log-likelihood aptamers have both L and R loops with high log-likelihoods $\mathcal{L}_+$, while the L and R loops of the low-log-likelihood aptamers have both low log-likelihoods $\mathcal{L}_-$. Intermediate aptamers have one loop, either L or R, with high loglikehood value $\mathcal{L}_+$ and the other with low log-likelihood $\mathcal{L}_-$.

Fig. 21c shows the scatter plot of the log-likelihoods of the full aptamers (estimated with RBM-DC) vs. the sums of the log-likelihoods of their L and R loops (estimated with RBM-SC). We observe an excellent linear correlation ($R^2 = 0.99$), indicating that both loops contribute additively to the score of the full aptamer. This linearity also explains the three peak structure of the aptamer log-likelihoods in Fig. 20a, approximately located at $2\mathcal{L}_-$, $\mathcal{L}_- + \mathcal{L}_+$, and $2\mathcal{L}_+$. Moreover, thanks to this linearity, the selection of the aptamer population from one SELEX round to the next one (Fig. 20) can be predicted also at the level of single-loop aptamers (see Fig. 80).

Fig. 21d shows the fractions of sequences in the four regions labelled I to IV of the L and R log-likelihoods at successive rounds of selection, see Fig. 21a. As observed in Fig. 20a for the full aptamer sequences we see a progressive enrichment in sequences for which both L and R loops have high log-likelihoods. However, we also observe a substantial fraction of sequences ($> 15\%$) at round 8, in which one loop only has high log-likelihood. The cognate 20-nucleotide sequences, with low log-likelihood on the other loop, will be called parasite in the following, as they are likely to be selected only due to the ability of the other loop to bind thrombin. To check this hypothesis we generate random aptamer sequences, in which the 40 nucleotides are drawn uniformly at random. As shown in Fig. 21b these random aptamer sequences are located in the $(\mathcal{L}_{-}, \mathcal{L}_{-})$ corner, and do not differ much from the pink sequences in terms of log-likelihood, see gray ellipse in Fig. 21b. Notice that removing the parasite sequences from the training set of RBM-SC does not significantly modify the estimation of log-likelihoods, see Supp. Fig. 70, which shows the robustness of the RBM model against the presence of random sequences in the data. The identification of parasite sequences has important consequences for the design of new aptamers based on the RBM model, as discussed in the next section.

4.3.5 RBM Parameters Reveal Functional Features of the Aptamer Sequences

We next extract the features that contribute the most to the likelihood of the sequences by studying weights between hidden units and visible layer (Fig. 19b). To enhance the interpretability of the RBM weights connecting input and and hidden layers we enforce their sparsity through appropriate regularisation (see Methods Sec. 4.6.2 and Ref. [97]). Figs. 22a-c (left) show the sequence logo of the three weights of RBM-DC with largest Frobenius norms (Fig. 73); the height of nucleotide symbol s in position i for hidden unit μ represents the value of the weight $w_{\mu i}(\mathrm{s})$.

We first observe that the weights are strongly localized either on the left or the right loop. The lack of correlation between the left and right loop sequences holds for all weights (Fig. 71), and is compatible with the additivity of their contributions to the aptamer log-likelihood in Fig. 21c.

A closer look at the sequence-dependence of the logos in Fig. 22a-c shows they are G-rich and match parts of G-quadruplex motifs. For instance, the hidden unit focusing on the right loop in Fig. 22a, is strongly activated when the motif AGGTTGG is present on the L loop in positions 33-39. Other L subsequences lead to much weaker activities (in absolute value), see right subpanel in Fig. 22a. A similar observation holds the left loop in Fig. 22c, with the motif GNNTGGTGTGGNTGG in positions 4-18 which is compatible with a G-quadruplex

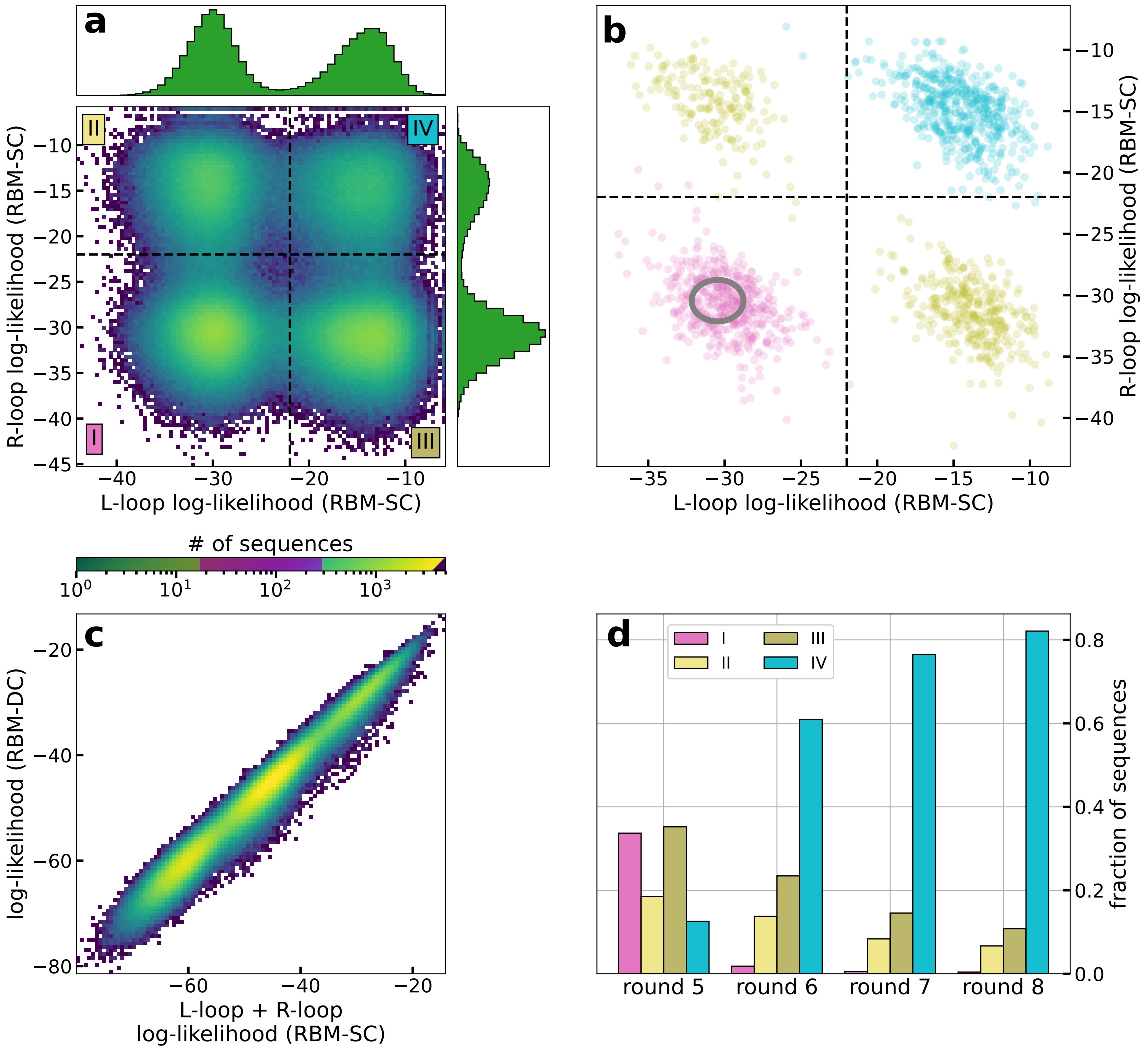


Figure 21. **Contribution of left and right loops in the RBM log-likelihood.**

a: Joint distribution of the log-likelihoods of the L and R loop subsequences at round 5, estimated with RBM-SC trained on subsequences at round 8. The insets show the marginal distributions for both loops.

b: Same as panel a for 2000 aptamer sequences attached to each of the three colored peaks in Fig. 20a (same color code). The gray ellipse shows the distribution of the log-likelihoods of uniform random sequences (center: mean values, ellipse: 2 standard deviations from the mean).

c: Log-likelihoods of the aptamers in round 5 (estimated with RBM-DC see Sec. B.3) vs. sums of the log-likelihoods of their L and R loops (estimated with RBM-SC).

d: Fractions of sequences in regions I to IV of panel a at rounds 5, 6, 7 and 8, estimated with the same RBM-SC model as in panel a.

structure. Other features are also detected by the RBM. As an example the weight logo in Fig. 22b is identifying long-range correlations across positions 1-20 associated consisting in a AT-rich motif and is present in some of the training sequences (see histogram in right subpanel).

Another relevant set of parameters learned from the data are the local fields acting on the visible variables. These parameters follow quite closely the nucleotidic profile of with the dataset, so they reflect a general enrichment in G-content, particularly in the positions most used to form G-quadruplexes (see Fig. 81 for the local fields of RBM-DC trained at round 6 and for the conservation logo of the sequences used to train the model).

We then explore the capability of RBM to provide low-dimensional representation of sequences. Prior experimental work [210] identified four different families of thrombin-binding aptamers (named A, B, C and D), based on sequence alignment and manual curation. We show in Fig. 22d the value of inputs I_μ of two hidden units of single-loop RBM-SC, ranked 2 and 7 in terms of weight Frobenius norms able to cluster these four families. Each hidden unit's activity (see Methods) has a bimodal distribution (Figs. 22e,f), and the combinations of these modes identify the four families.

4.3.6 RBM Trained from Unique Sequences Generate Diverse Aptamers Capable of Binding Thrombin

After having established that the RBM log-likelihoods and the fitnesses of the aptamers in our dataset are strongly interrelated, we now use the RBM model to generate new sequences *in silico* (see Methods Sec. 4.6.2). Note that the number of available sequences at any round, $< 10^6$, is much smaller than the number of possible sequences over 20 nucleotides, $4^{20} \simeq 10^{12}$. Hence, it is a non trivial problem to reconstruct the full likelihood landscape from such undersampled data, and use it to generate new binders.

Sampling RBM-SC trained on round-8 data reveals a lack of diversity in the sampled sequences: all the generated sequences with high log-likelihoods are already present in the

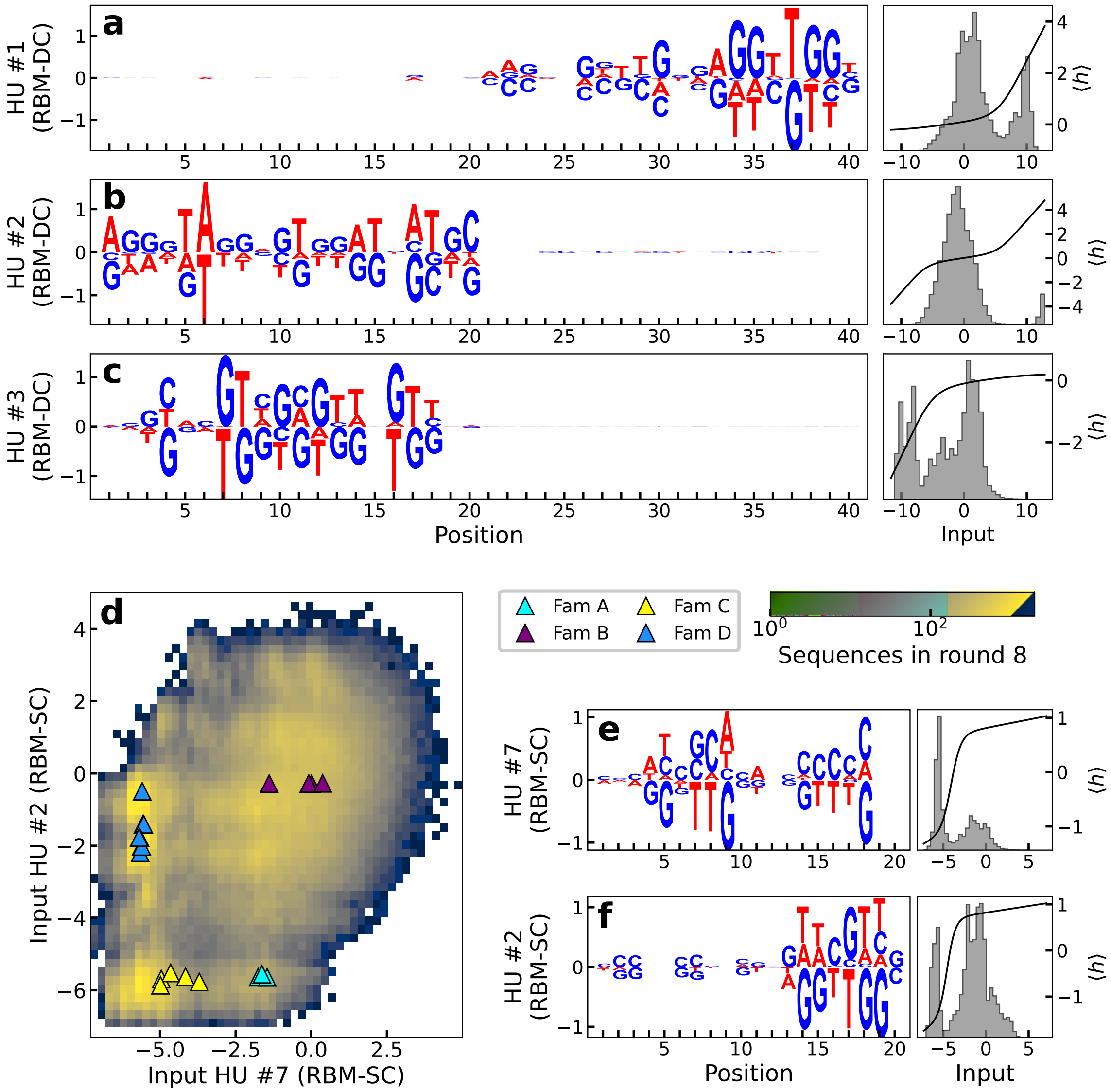


Figure 22. **Weights learned by the RBMs have biological interpretations.**
a-c: Left: logos of three weights $\mu = 1, 2, 3$ with largest Frobenius norms for RBM-DC trained on round 8 aptamer data; Right: histograms of the inputs $I_\mu = \sum_i w_{\mu i}(s_i)$ (where $w_{\mu i}$ is the weight of the connection between hidden unit μ and visible unit i for nucleotide s_i) to the corresponding hidden units for the sequences **s** in the dataset (gray) and average activity (black).
d: The four families identified in [210] are separated in different clusters in the two-dimensional subspace spanned by the inputs to hidden units 2 and 7 of RBM-SC (trained on loop subsequences at round 8).
e, f: Logo, distribution of inputs and average activity of the same hidden units as in panel d.

dataset (Fig. 23a). RBM-SC rightly assigns high scores to the strong binders present at the end of SELEX procedure, but is unable to generate diverse sequences with high scores (Fig. 23a).

We then train another model, called RBM-SU, by maximizing the sum of the log-likelihoods of unique sequences in round 8 dataset (composed of 382094 unique single-loop sequences), see Eq. (4.1). Details of the training procedure are given in Sec. B.3. The rationale for this approach is two fold. First, 8th round data are expected to include better binders and much less parasite sequences than earlier rounds. Second, discarding the sequence counts prevents the model from being dominated by few very good binders to thrombin.

The effective diversity of training data is reflected in the generated sequences from RBM-SU model. A large fraction of sequences generated by RBM-SU with top log-likelihoods are not present in the dataset, contrary to what found with RBM-SC, see Fig. 23a. In addition, about 30% of generated sequences are 4 or more nucleotides away from the dataset, as is the case for the majority of randomly generated sequences of length 20 nucleotides. Furthermore, we show in Fig. 23b that RBM-SU exhibits excellent generalization properties. The log-likelihood of test data (unique sequences present at round 8 but not used for training) is very close to the one of the training data. On the contrary, RBM-SC essentially assigns high scores to high-count sequences in the training data, and shows poor generalization.

We have next experimentally tested the binding to thrombin of some aptamer sequences to validate the ability of the RBM-SU to predict binding and to generate *de novo* binders. The 20-nucleotide DNA sequences are first inserted into the loop of a hairpin with fixed 18 base-pair-long stem. To estimate the binding affinity to thrombin we use native gel shift assay, where we incubate the thrombin protein with the hairpin aptamer, see Methods Sec. 4.6.4 and Supp. Inf. Sec. B.1.

A set of 16 sequences listed in Table 27 (excluding the control sequences listed in the Table), together with 4 binders, experimentally validated in [210] and named ThA, ThB, ThC and ThD, is first used to estimate the log-likelihood threshold above which a sequence is predicted to bind thrombin, see Fig. 23d, where the log-likelihoods of tested sequences are represented as vertical red and green lines, for verified non-binders and binders respectively.

We then propose a set of 27 sequences to test (r1-r27 in Table 4): 2/3 of them are *de novo* designed sequences generated from the RBM-SU model, and the remaining 1/3 are present in round 8. *De novo* sequences are chosen to test the power of the RBM model to produce good thrombin binders, or to predict critical mutations transforming binders into non binders. Sequences already present in the round-8 data are chosen to test non trivial RBM predictions, *e.g.* sequences with low or high counts having, respectively, high or low log-likelihoods. The detailed description of these sequences and of the design criteria is found in Method Sec. 4.6.3.

Over the 27 sequences to test, 21 sequences were above threshold, and therefore predicted as binders and 6 sequences below threshold, predicted as non binders. The experimental gel assays are shown in Fig. 24. Overall, 93% of the RBM predictions (binder or non binder) are confirmed by experiments. The log-likelihoods of the tested sequences, along with the RBM predictions and the experimental findings are reported in Table 4 and represented with the experimental results in Fig. 23e.

These results show that the log-likelihood provided by RBM-SU is an accurate predictor of the capability to bind thrombin. We show in the inset of Fig. 23c the receiver operating characteristic (ROC) curve and the corresponding area under the curve (AUC=0.99) for RBM-SU-generated sequences. Let us stress that RBM-SC, shows poor performance in discriminating good from bad binders among these sequences, see Fig. 79. This failure is expected from the poor generalization abilities of RBM-SC for sequences with low counts (Fig. 23b).

4.3.7 Competition Assay for Exosite Binding Site and Binding Strength Measurements

Thrombin has two exosites, referred to as I and II, which can be bound by aptamers, *e.g.* ThA is known to bind exosite II, while ThD binds exosite I [210]. We first identify the

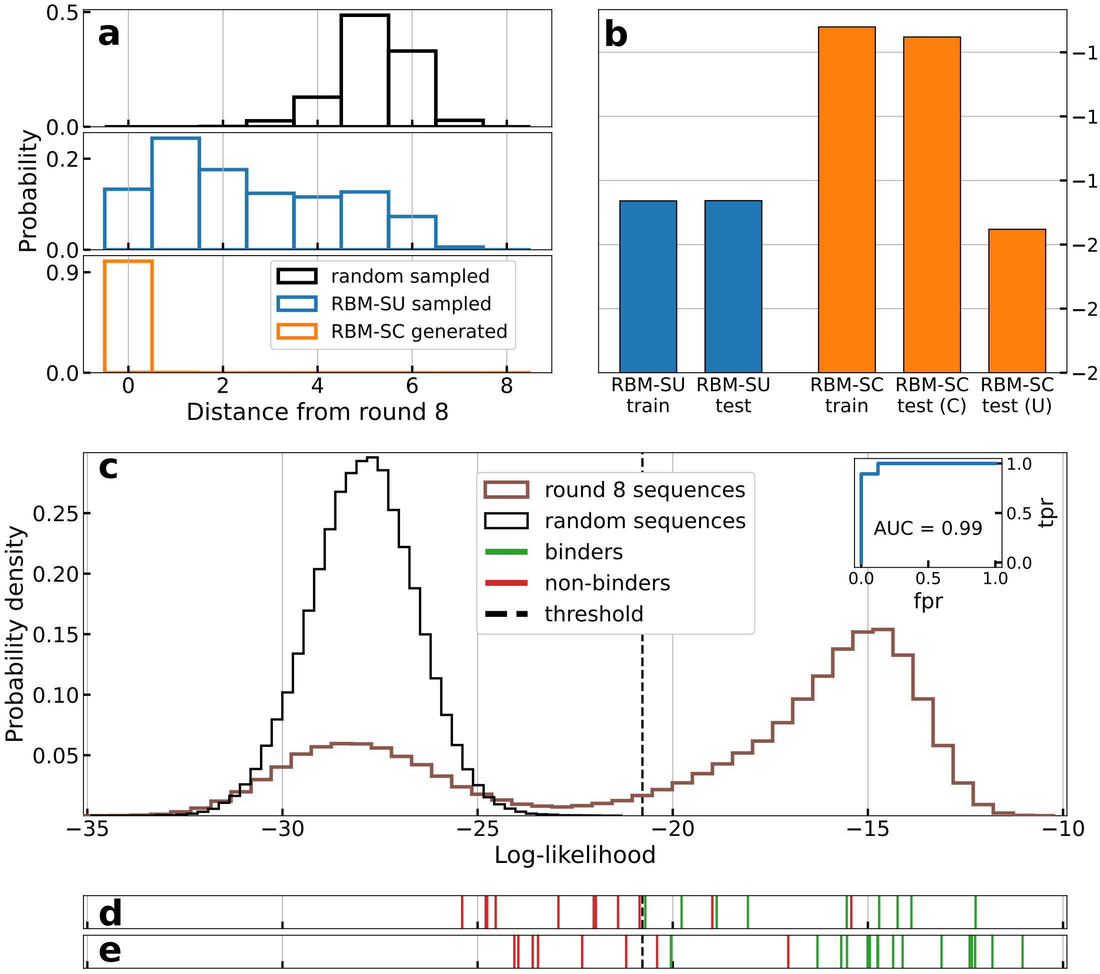


Figure 23. **RBM can be used to design new aptamers binding thrombin.** **a**: Histograms of distances to dataset of the best (top 5% in terms of log-likelihoods) sequences generated by RBM-SC (orange) and RBM-SU (blue) trained on round 8 data. The black histograms show the distribution obtained with sequences generated uniformly at random. **b**: Average log-likelihoods of training (90% of the unique sequences observed at round 8, chosen at random) and test (remaining 10% of unique sequences) sets for RBM-SU and RBM-SC (after re-introduction of counts). For RBM-SC, the average log-likelihood of the test set is shown either when weighing each sequence with (C) or without (U) its counts. For the RBM-SU model we show the unweighted average-log likelihood on the training and testing sets respectively. **c**: Histogram of the log-likelihoods of all unique aptamers observed in the last round (blue line) and of uniformly random sequences (orange line), computed with RBM-SU trained on the same data (blue line). Inset: AUC computed on the sequences generated by the RBM-SC model (panel e). **d**: Vertical lines locate the log-likelihoods of sequences experimentally validated to be binders (green) or non binders (red). Sequences taken from a preliminary set described in Table 27. Results allow us to determine the binding/non binding threshold, located with the black dashed line. **e**: Same as panel d, for sequences designed with the RBM-SU model trained on round-8 sequences, as described in Sec. 4.3.6, see Table 4.

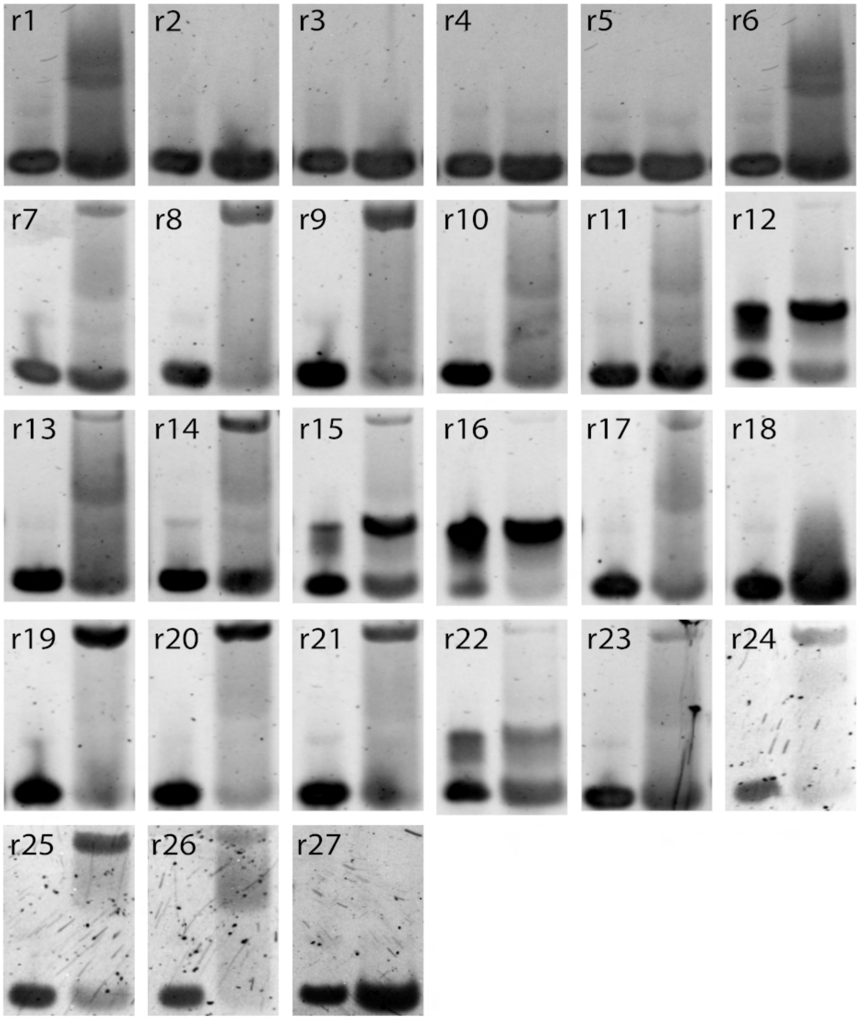


Figure 24. Experimental measurements of binding of respective designed sequences (r1 to r27) to thrombin. 5% native gel assay at 15 °C of stem loops (1-27) alone in the presence of Mg^{2+}/K^{+} (lane 1) and allowed to mix with α-thrombin for 30 minutes at 25 °C on the bench (lane 2). r12, r15, r16, and r22 aptamers were forming dimers with themselves but upon using samples without K^{+}, they were found to bind thrombin. Their entries above display the successful attempt(see Methods for further details). Aptamers r1 and r6 did not show a clear upper band that is indicative of thrombin-aptamer dimer, but the observed smear might indicate weak interactions with the thrombin.

target exosite for all the binding aptamers among the r1-r27 by testing each of them (aside from those which were found to form dimer states, see Table 4) against ThA and ThD, see Methods Sec. 4.6.5. In such a competition assay, the designed aptamers are preincubated with thrombin and are put in competition with a small amount of fluorescently labelled ThD or ThA [210]. If the pre-incubated and fluorescent strand bind the same exosite a thrombin/fluorescent strand complex is observed in the same position as in the thrombin binding assay. However, if the pre-incubated and fluorescent strands target different exosites thrombin is bound twice, causing a downward shift in the observed band (Fig. 55). As shown in Fig. 25 and Table 4 we find that all thrombin-binding aptamers among sequences r1-r27 bind exosite I, except one.

As we noticed that sequence r9, which is an exosite-I binder, is only 3 mutations away from ThA, which binds exosite II, we decided to test all six intermediate sequences, labelled as p1-p6 in Table 4. One mutation (Adenine vs. Thymine on site 17) seems to control the exosite binding preference along the mutational path, see Table 4 and Fig. 78. Analysis of the RBM-SU weights confirms that position 17 is particularly relevant on the aptamer sequence: many weights have non-zero values on this site (Fig. 74). To understand if the presence of A on site 17 (rarely encountered in round 8 sequences) is sufficient to guarantee binding to exosite II we specifically design four sequences (r24 to r27) with this feature and log-likelihoods above threshold, see Methods 4.6.3 and Table 4. As reported above none of these sequence turns out to bind exosite II (while 3 out of 4 bind exosite I), showing that binding specificity is generally controlled by multiple-nucleotide motifs along the sequence.

Next we test if any of the *de novo* generated aptamer sequences with high RBM-SU log-likelihoods are stronger binders than previously identified ThD and ThA aptamers, the binders with the largest number of counts at the end of SELEX [210]. To determine the strongest binder using competition assays, thrombin is mixed with a mixture of the control and the test aptamers at equal ratios, with the control strand being fluorophore labelled (details in Supp. Mat. Sec. B.1.4). The stronger binder is considered to be the control or the test aptamer when fluorescence is observed, respectively, in the thrombin-aptamer gel band or in the stem loop band (the unbound aptamer), see Figs. 57 and 56. We observe that none of the designed aptamers binds thrombin more strongly than ThA to exosite II binders, or than ThD to exosite I binders. This result is expected: given the size of the original library ($\sim 10^{15}$) virtually all possible sequences of 20-nucleotide aptamer are initially present, so it is unlikely that SELEX misses stronger binders than ThA and ThD.

We then ask whether the outcomes of competition assays for the best binders could be predicted from the comparisons of their log-likelihoods. RBM-SC-based predictions have 100% success with respect to the above competition assays, always assigning larger scores to ThA and ThD than to the competing aptamers. Conversely, RBM-SU underestimates ThA and ThD binding strength, assigning, in particular, low log-likelihood to ThA and having a global performance of 38% on performance of RBM-generated sequences in the competition assays with ThA and ThD. However, for competitive assays between sequences r1-r27, RBM-SU scores are slightly more predictive than their RBM-SC counterparts, with fractions of successful predictions equal to, respectively, 67% and 59%. Interestingly RBM-SU and RBM-SC also depart from one another in their estimates of the log-likelihoods of exosite I and II binders. We observe in Fig. 26 that aptamers binding exosite I have higher scores than their exosite II counterparts, explaining the overwhelming presence of exosite I binders among RBM-SU generated sequences. On the opposite, RBM-SC generally assigns higher log-likelihoods to exosite-II binders. The differences in the behaviours of these models are further examined in Discussion.

4.3.8 Supervised Learning Approach

We also explored supervised learning approaches to train from the aptamer datasets. We considered several DNN architectures (ResNet, Siamese Network and variational autoencoder) as well as traditional methods (random forest and gradient boosted tree) that

		Log-likelihood		Binding	Binding		Distance
Label	Sequence	RBM-SC	RBM-SU	Pred.	Result	Exosite	round 8
r1	AGTGATGATGTGTGGTAGGC	-11.5	-23.4	NB	NB*	NA	0
r2	AGTGTAGGTGTGGATGATGC	-11.4	-24.0	NB	NB	NA	0
r3	TAGGTTTTGGGTAGCGTGGT	-13.0	-22.3	NB	NB	NA	1
r4	AGGGATGATGTGTGGCAGGA	-17.3	-23.6	NB	NB	NA	1
r5	CTAGGACGGGTAGGGCGGTG	-15.9	-21.2	NB	NB	NA	1
r6	AGGGATGTGTGTGGTAGGCT	-14.1	-23.9	NB	NB*	NA	0
r7	AGGGATGCTGCGTGGTAGGC	-10.2	-20.0	B	B	II	0
r8	GAGGGTTGGTGTGGTTGGCA	-10.6	-11.0	B	B	I	0
r9	AGGGTTGGTGTGTGGTTGGC	- 9.8	-11.8	B	B	I	0
r10	ATGGTTGGTTTATGGTTGGC	-15.2	-14.7	B	B	I	1
r11	GAAGGGTGGTCAGGGTGGGA	-16.5	-15.7	B	B	I	2
r12	GGAGGGTGGGTCGGGTGGGA	-15.2	-15.0	B	B	NA	1
r13	GGGGTTGGTACAGGGTTGGC	-16.3	-14.9	B	B	I	2
r14	AGATGGGCAGGTTGGTGCGG	-16.3	-16.3	B	B	I	2
r15	AGATGGGTGGGTAGGGTGGG	-13.9	-14.3	B	B	NA	2
r16	ATAGGGTGGGTGGGTGGGTA	-13.1	-15.0	B	B	NA	1
r17	TGGTGGTTGGGTTGGGTTGG	-12.8	-12.3	B	B	I	1
r18	TGGGATGGGATTGGTAGGCG	-12.2	-20.4	B	NB	NA	0
r19	AGGGTTGGTTATGTGGTTGG	-19.3	-20.0	B	B	I	0
r20	ATTGGTTGGGTAGGGTGGTT	-10.4	-12.2	B	B	I	0
r21	AAACGGTTGGTGAGGTTGGT	-11.2	-12.4	B	B	I	0
r22	CGGGGTGGTGTGGGTGGGAG	-15.1	-14.7	B	B	NA	2
r23	TATTGGTTGGATAGGTTGGT	-13.8	-13.1	B	B	I	1
r24	AGGGTTGGGTGGTTGGATGA	-14.9	-14.1	B	B	I	1
r25	CGGGTTGGGGGGTTGGATTC	-17.0	-15.0	B	B	I	1
r26	CGGTTGGGGGGGTTGGATAC	-18.8	-15.5	B	B	I	1
r27	TGTGGGTTGGTGAGGTAGGT	-18.0	-17.0	B	NB	NA	1
ThA	AGGGATGATGTGTGGTAGGC	-6.0	-19.8	B	B	II	0
ThB	AGGGTAGGTGTGGATGATGC	-5.7	-20.7	NB	NA	II	0
ThC	TAGGTTTTGGGTAGGGTGGT	-6.8	-18.1	B	NA	I	0
ThD	GTAGGATGGGTAGGGTGGTC	-5.7	-13.9	B	B	I	0
p1	AGGGATGATGTGTGGTTGGC	-10.3	-17.1	B	B	I	0
p2	AGGGATGGTGTGTGGTAGGC	- 9.2	-16.2	B	B	II	0
p3	AGGGTTGATGTGTGGTAGGC	- 7.2	-19.1	B	B	II	0
p4	AGGGATGGTGTGTGGTTGGC	- 9.3	-13.1	B	B	I	0
p5	AGGGTTGATGTGTGGTTGGC	-11.1	-16.2	B	B	I	0
p6	AGGGTTGGTGTGTGGTAGGC	- 9.7	-15.2	B	B	II	0

Table 4. Sequences generated from RBM-SU, log-likelihoods, binding predictions (based on the comparison of the RBM-SU log-likelihood and the threshold in Fig. 23d), and results from gel shift assay (B for binders, NB for non binders) and exosite binding assays. For comparison, data for ThA, ThB, ThC and ThD sequences from Ref. [210] are shown. ThB and ThC have not been tested for binding with our experimental setup (so NA is used in the corresponding column), although they are expected to bind thrombin given the results obtained in Ref. [210]. *Aptamers and r1 and r6 did not show thrombin binding gel band, but their pattern indicates a possible weak interaction with thrombin.

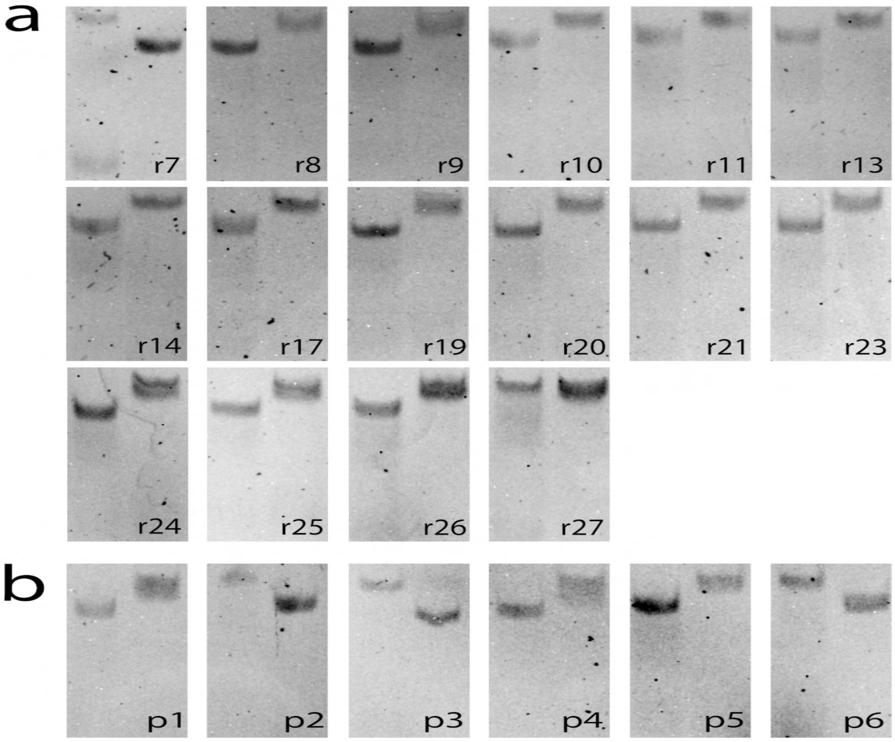


Figure 25. **a**: Binding site assay of all binding sequences and r27 (a nonbinder control) in the RBM generated dataset. **b**: Binding site assay of the 6 sequences that make up the sequence space between ThA and test sequence r9. For all gels, Lane 1 shows addition of ThA and lane 2 shows addition of ThD to the thrombin pre-incubated strand (labeled in black). Results are reported in Table 4.

we trained to classify sequences as binders or non-binders (see Sec. B.4). Training was complicated by the fact that the aptamer dataset only contained positive examples (binders from different selection rounds with their respective counts obtained from the sequencing step). Hence, we either classified sequences with low counts as non-binders, or we generated random sequences not present in the dataset and treated them as non-binders. The first approach achieved between 70% to 84% accuracy on the validation dataset. The second approach had at least 99% accuracy on the validation dataset for all models. However, when evaluating models against the test set (sequences from Table 4), we observed 30% to 74% accuracy for the first approach, and 70% to 89% for the second approach, as the test set is heavily biased to binding sequences, and methods with high accuracy classified most

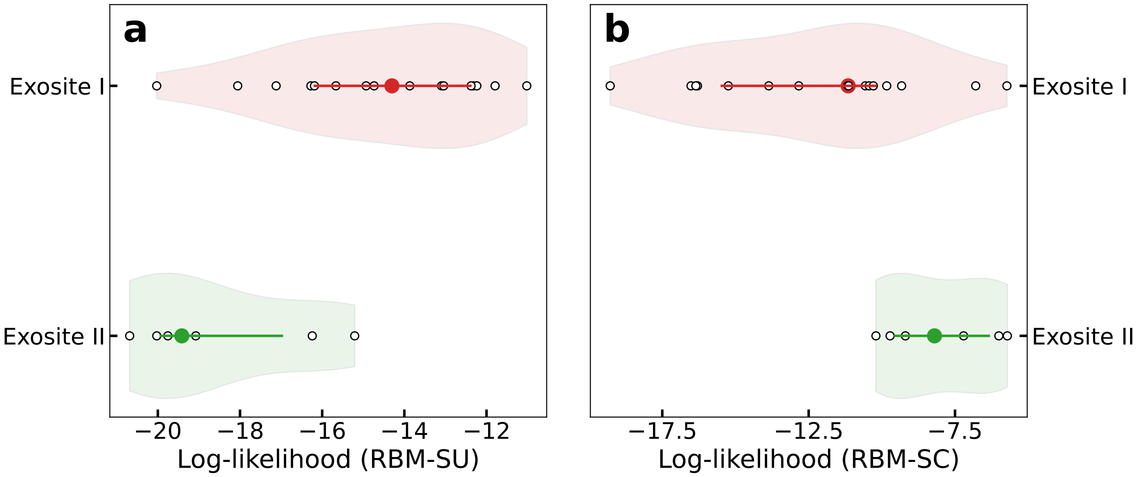


Figure 26. **Aptamers binding to exosite I have larger log-likelihoods with RBM-SU, lower log-likelihood with RBM-SC.** Violin plots showing the log-likelihoods of exosite I (light orange violin) and exosite II (light green violin) binders. Circles in darker colors denote the average log-likelihood over the class, lines denotes 25- and 75-percentiles, and white points corresponds to the log-likelihoods of the generated sequences. In panel a RBM-SU is used, while in panel b RBM-SC is used.

non-binders as false positives. These results indicate that SELEX datasets are challenging for the commonly used supervised learning methods.

4.4 Discussion

In this work we proposed data-driven models of aptamer sequences obtained at different stages of directed evolution for thrombin binding. Our models are based on Restricted Boltzmann Machines (RBM), the simplest neural network architecture embedding the notion of representation (or latent factors) of sequence data.

One of our main findings is that the score (log-likelihood) assigned by the model to a sequence $\boldsymbol{s}$ was linearly related to its fitness $F(\boldsymbol{s})$ in the SELEX experiment. More precisely, repeated applications of Eq. (4.1) at previous rounds of selection imply that the likelihood of a sequence $\boldsymbol{s}$ at round r is related to its fitness through

$$p_r(\mathbf{s}) \propto e^{\beta_r F(\mathbf{s})} \quad , \qquad \beta_r = \alpha_0 + \alpha_1 + ... + \alpha_{r-1} \ , \tag{4.3}$$

where α_{k-1} is the intensity of selection from round $k-1$ to k, see Eq. (4.1), and the initial library is assumed to be roughly uniform over the sequence space ($\beta_0 = 0$). This equation can be conveniently rephrased in the language of statistical physics. The rounds of SELEX selection shape a Boltzmann-like distribution over the aptamer sequences, corresponding to an effective energy equal to minus the fitness, $-F$. The effective inverse temperature β_r at round r is the sum of the intensities of selection at the previous rounds, and measures the cumulative effect of these previous selections. As more rounds are carried out, the effective temperature $1/\beta_r$ diminishes, and the distribution of sequences concentrates around the fittest aptamers, *i.e.* the sequences $\boldsymbol{s}$ maximizing $F(\boldsymbol{s})$, see Fig. 27. As more and more rounds r of SELEX are applied to the aptamer population the cumulative selection strength β_r seem to saturate, a phenomenon compatible with previous theoretical works [213] and observed in other SELEX experiments [198].

The values of the selection strengths α_r and of the cumulative selection strengths β_r can be extracted from our analysis; for definiteness we arbitrarily choose $\beta_6 = 1$ to fix the scale of the fitness F, as Eqs. (4.2) and (4.3) are obviously unchanged under the rescaling $\alpha_r, \beta_r \to \lambda\, \alpha_r, \beta_r, F \to F/\lambda$. First, we report in Fig. 77 the scatter plots of the log-likelihoods of the sequence data with models trained at different rounds, say, r and r'; the slopes of these scatter plots give access to the ratios $\beta_r/\beta_{r'}$ according to Eq. (4.3). Second the linear fits of the log-likelihood (estimated with the RBM trained on round-6 data) vs. log. enrichment ratios, as well as the Fisher ratios shown in Fig. 20c provide estimates of the ratios α_r/β_6.

The double-loop nature of the aptamer sequences studied here is at the origin of two interesting phenomena. First, we find that $\log p_r(\boldsymbol{s})$ and, consequently, the fitness $F(\boldsymbol{s})$ are, to a very good accuracy, equal to the sum of two contributions coming from the left and from the right loops. This additivity property suggests a mechanistic picture of the binding of aptamers to thrombin. The enrichment factor of the set of molecules carrying the sequence $\boldsymbol{s}$ is proportional to the probability p_{bind} that they bind thrombin and to their amplification factor through PCR. Hence, $\log p_{\text{bind}}$ is proportional to the fitness and additivity of the latter implies that p_{bind} is the product of the binding probabilities of the left and right loops. The two loops of aptamers are thus progressively required, through successive SELEX rounds, to bind the thrombin target. While double-loop aptamers with one binding loop and one parasite subsequence exist in early rounds, they progressively disappear (Fig. 20a). The bivalence of aptamers in the final rounds likely reflects the strong selection pressure imposed by SELEX.

The RBM model also allows for identification of the nucleotide motifs in the aptamer sequence that contribute most to the sequence likelihood, or, equivalently, to its fitness. Such motifs are indicative of a G-quadruplex group, a known functional motif in the DNA aptamers that bind thrombin [214]. Other RBM motifs could also allow one to help identify clusters of sequences (subfamilies), investigated in prior works through sequence alignments and manual curation.

A second major finding is that the RBM model is capable of generating new sequences, not present in the dataset, with good binding properties. We have generated 27 aptamer sequences from the RBM that were either predicted to bind or not bind to thrombin. Out of 21 sequences that were thought to be binders, 19 were confirmed to bind thrombin, and all 6 sequences generated as non-binders were rightly predicted so. These non-binder sequences were generated under the non-trivial constraint to differ as little as possible (in terms of mutated nucleotides) from known good binders.

We stress that the capability of RBM models to generate diverse aptamers crucially depends on how they are trained. Standard training, where the counts of sequences are taken into account result in models giving very high scores to the very best binders in the dataset, but unable to generalize beyond these few sequences (Fig. 23b). On the contrary, discarding the count information and maximizing the log-likelihood of the set of unique sequences produces models with very good generalization properties, and able to design new and diverse binders, as confirmed in the experiments reported above. The choice of considering unique sequence is partially reminiscent of the reweighting procedure used in sequence-based modeling of proteins [206, 97], and allows the inferred log-likelihoods to reflect more accurately the probabilities for sequences with low number of counts, see Fig. 27. Notice that, while

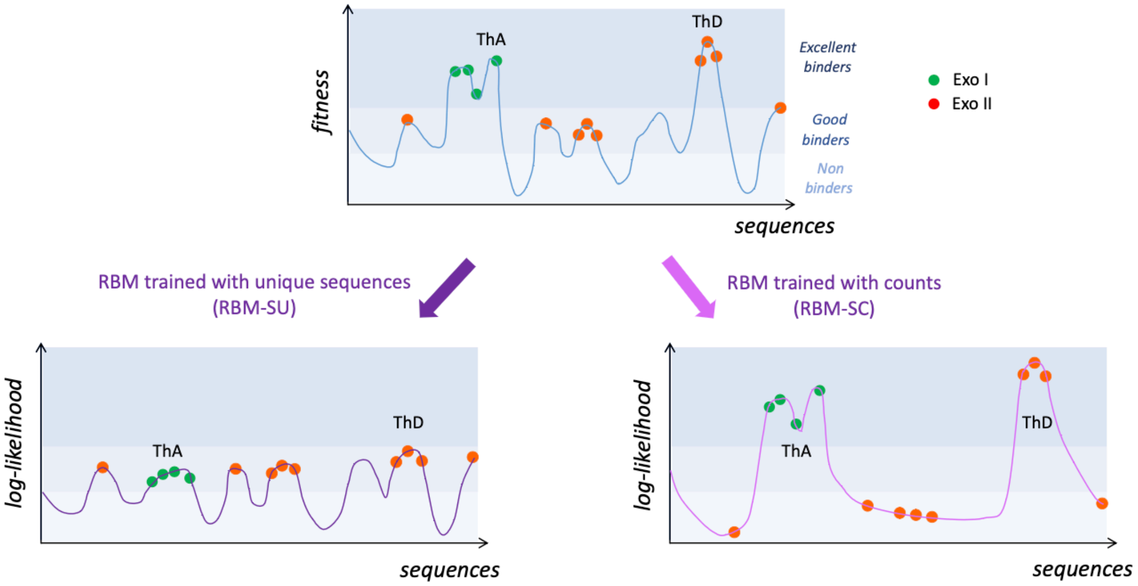


Figure 27. **Sketches of the fitness and inferred landscapes.**
Top: fitness of the aptamer sequences as estimated by the SELEX experiment. After some rounds of selection, most sequences are good binders to thrombin and have low counts (very often, $C = 1$), while some are excellent binders and have large counts. Two excellent binders, ThA and ThD, are schematically shown. Bottom: log-likelihood landscapes defined by the RBM models, trained from unique sequences (RBM-SU, left) or taking into counts (RBM-SC, right). RBM-SU is able to capture the statistical features of the many good binders, but does not reproduce well the few high-fitness peaks. It can be used to generate new sequences (empty peak in the landscape). Conversely, RBM-SC accurately models the high peaks in the fitness landscape, but is unable to reproduce the detailed structure of the landscape at lower levels. It cannot be used to generate new binders.

unique-sequence-based training could *a priori* be sensitive to sequencing errors we estimate that the probability ϵ of misreading a nucleotide is $< 10^{-3}$ (see Methods Sec. 4.6.1 and Sec. B.2), in agreement with error rates with next generation sequencing methods [215]. As a result spurious sequences are $< 0.5\%$ of all unique sequences in the dataset, and have only marginal impact on the trained model. However, ensembles in other SELEX experiments using modified bases might experience higher sequencing error rates, which our approach would allow to identify and correct for (Methods Sec. 4.6.1).

The properties of the two models are graphically summarized in Fig. 27. RBM-SC, which takes into account counts, accurately models the high peaks of the fitness landscape, but discards the smaller peaks. It rightly assigns very high log-likelihoods to the excellent binders, such as ThA or ThD. However, at this level of fitness, the diversity of the sequences that can be generated is very poor. Conversely, RBM-SU, captures the statistical features of sequences at a much lower level of fitness. Many varied sequences can then be generated, the majority of which are good binders. RBM-SU is therefore able to generate more diverse and less strong binders, which makes it particularly appropriate for the design of evolvable aptamers [216]. In principle, RBM-SC inferred from sequences collected in an early round would have had similar properties to RBM-SU inferred from round-8 data. However, in the specific problem of double-loop aptamers we consider here, the presence of a large number of parasite single-loop sequences at the beginning of SELEX evolution could also affect the generative power of models trained at early rounds.

We next used a competitive binding assay both to first classify the binding site of the generated sequences and, in a second step, to assess the strength of binding to a given exosite. We find that the majority of sequences generated with RBM-SU preferentially bind to exosite I. In addition, sequences binding exosite I have on average higher log-likelihoods than the few exosite-II binding sequences. In particular, ThA, an exosite-II binder with a large number of counts in the SELEX experiment is not among the sequences with highest RBM-SU log-likelihoods. Furthermore direct competition experiments between the highest log-likelihood sequences and ThA or ThD (binding exosite I and having a large number of counts) showed that the latter aptamers outperform the former in terms of binding affinity. These apparently paradoxical results can be explained in two ways. First, the log-likelihoods were estimated with the model used for generating sequences, that is, RBM-SU. This model is very good at generate diverse binders, but is not trained to reproduce counts. The absence of correlation between RBM-SU log-likelihoods and counts or binding affinities is therefore not surprising, whereas RBM-SC high scores show a good correlation with large counts as expected (Fig. 76,). Second, these results are compatible with a selection mechanism involving binding to the two sites of thrombin. Binding to exosite II has been shown to facilitate binding to exosite I, presumably through allosteric structural change [210]. Due to this allostery mechanism, when exosite II is loaded (even with a different molecule), hairpin with a low-affinity loop to exosite I could be selected. This mechanism could produce a rather subtle parasitism, where only the best exosite II binders in a quasi-monoclonal population (few sequences with largest counts) are under strong selection, and allow for the presence of a more diverse exosite-I binder population. Further experimental investigations combined with theoretical analysis, *e.g.* using concepts developed in ecosystems dynamics in presence of parasite populations, could help to further investigate the selection dynamics.

We note that our RBM represents a higher level of complexity than the direct contact analysis methods (DCA) that have also been recently applied to protein ensemble selection experiments [217]. While the DCA method trained using the pseudo-likelihood method was not able to correctly predict binders and non-binders for our dataset, when we used contrastive divergence training for DCA, the assigned scores from the trained DCA model showed correlation with our trained RBM (see Supp. Inf. Sec. B.5). As opposed to DCA, which infers pairwise interactions, the RBM model's hidden units can be used for clustering of sequences or identification of multi-nucleotide motifs, such as G-quadruplexes, making them more readily interpretable. We have explored using supervised learning models, including DNNs, on our datasets predicting binders and non-binders, but as further detailed in Supp. Inf. Sec.B.4, we did not obtain good prediction accuracy for the outcomes of our experiments with designed sequences.

4.5 Conclusion

In this work, we presented an unsupervised learning approach for modeling sequence ensembles obtained from selection experiments based on Restricted Boltzmann Machines (RBM). The approach was applied to previously obtained data from SELEX experiment to find thrombin bivalent aptamers nanostructures that bind two different exosites. More precisely, our approach consisted of the following steps: 1) developing a method that estimated sequencing error rates, which could be used for curation of the sequence data, 2) showing that the log-likelihood of the trained RBM accurately predicted aptamer fitness in terms of its propensity to be enriched in later rounds of the experimental selection protocol, 3) using RBM to identify contributions of the two aptamer loops to exosite binding, 4) showing that inspection of the parameters of the trained RBM identified functional features (such as G-quadruplex) of the selected sequences, 5) using the trained model to generate novel sequences, whose ability to bind thrombin was verified experimentally, and 6) comparing RBMs with different supervised learning models trained on the same dataset, with the result that RBM generalized better.

We emphasize that the calculation of log-likelihood and hence of the fitness of any designed sequence by RBMs is very efficient, making them faster than other approaches based on e.g. docking or free-energy estimation from molecular simulation. Furthermore, the structure of the model allows us to capture and identify complex features that could include covarying residues or motifs. We showed that RBM training can be flexibly adapted depending on the scope, e.g. taking into account sequence counts or not allows one to design stronger or more diverse binders. We anticipate that RBMs will be also useful for the modeling of other aptamer datasets with more complex selection protocol, such as competition assays where aptamers are selected to bind to a desired target, e.g. cancerous tissues, and at the same time not to bind to the control, e.g. healthy tissue. We believe our approach has the potential to generate alternative or better binders for these complex targets, as well as to unveil the sequence motifs that are enriched or avoided in these high-quality aptamers. The same approach can be also useful to model RNA and DNA regulatory sequences and their interaction with proteins in the key processes such as transcription regulation [209, 218, 203, 204]. Lastly, our modeling and design methods are also readily applicable to other selection-amplification protocols, such as phage display for antibody discovery [219, 220] or directed protein evolution studies [217, 221], which have much larger space of possible sequences (20^L for length L) compared to aptamers (4^L).

4.6 Methods

4.6.1 Estimation of Sequencing Error Probability

Sequencing errors are potentially harmful, as they could lead to more unique sequences in the dataset and possible biases in the RBM models. We introduce an inference approach to estimate the sequencing error rate, based on the presence of spurious single-site mutations of sequences with high number of counts. In practice the method consists in selecting a subset of sequences with high number of counts, referred to as "peak" sequences, and in comparing the expected number of sequences one mutation away from these peaks due to sequencing errors to the actual number in the data. Our analysis, detailed in Sec. B.2, indicates that the error rate (per nucleotide) is smaller than $\epsilon^* \sim 10^{-3}$.

We use this bound to estimate the expected number of spurious sequences present in the dataset. We obtain $N_{\text{spurious}} \sim 1000$ unique sequences (see SupplementarySec. B.2), corresponding to $\sim 0.5\%$ of the total number of unique sequences present in the data.

4.6.2 Restricted Boltzmann Machine: Definition, Training, Sampling

The probability of a visible and hidden units state in an RBM model is defined by

$$p(\boldsymbol{s}, \boldsymbol{h}) = \frac{1}{Z} \exp\left(\sum_{i=1}^{L} g_i(s_i) - \sum_{\mu=1}^{M} \mathcal{U}_\mu(h_\mu) + \sum_{\mu,i} h_\mu w_{\mu i}(s_i) \right), \tag{4.4}$$

where Z is the normalization, g_i, and $w_{\mu i}$ are parameters to be inferred from the data during training, and

$$\mathcal{U}_\mu(h) = \frac{1}{2}\gamma_{\mu+}(h_+)^2 + \frac{1}{2}\gamma_{\mu-}(h_-)^2 + \theta_{\mu+}h_+ + \theta_{\mu-}h_-, \tag{4.5}$$

where $h_+ = \max(h, 0)$, $h_- = \min(h, 0)$ and $\gamma_{\mu+}, \gamma_{\mu-}, \theta_{\mu+}, \theta_{\mu-}$ are again model parameters to be inferred from the data during training. This specific form of the function $\mathcal{U}_\mu$, which is called "double Rectified Linear Unit" combines the usage of a relatively low number of parameters with the possibility of learning high-order correlations in the data [97]. An advantage of choosing Double ReLU potentials is that the likelihood $\log p(\boldsymbol{s})$ of a sequence $\boldsymbol{s}$, obtained by marginalizing $p(\boldsymbol{s}, \boldsymbol{h})$ over $\boldsymbol{h}$, has an explicit analytical expression in terms of error functions.

It has been suggested that, for RBMs, sparsity of the weight parameters, together with a high number of hidden units, can improve the generative properties of the machine and its interpretability [222, 97]. To prevent the model from overfitting, we hence enforce sparsity of weights and we empirically set M to value above which the model's log-likelihood on validation dataset does not further increase. We resort to a L_1^2 regularization scheme, which consists in adding to the log-likelihood of the data, $\mathcal{L}$ in Eq. (4.1), a term of the form [97, 99]

$$-\lambda \sum_{\mu=1}^{M} \left(\sum_{i=1}^{L} \sum_{s_i} |w_{\mu i}(s_i)| \right)^2 , \tag{4.6}$$

hence enhancing sparsity homogeneously across hidden units. The value of the hyperparameter λ must be, in general, chosen carefully to balance model interpretability (obtained for sparse weights, *i.e.* large λ) and expressivity (to learn data features). We observed little effects of changes in hyperparameters (see also Fig. 75), provided that they are not too different from the one given in Sec. B.3. This is also the case for the number M of hidden units chosen: we used $M \simeq 70$ for RBMs with $L = 20$ visible units, and $M \simeq 90$ for $L = 40$. Precise values of M are given, for each RBM used, in Sec. B.3, but we noticed that using different numbers have little effects on the results discussed in this work (see also Fig. 75).

Once the parameters in Eq. (4.4) are obtained, we can sample from the marginal distribution $p(\boldsymbol{s})$ to generate new sequences. Sampling can be done in several ways [223]. Here we use alternate Gibbs sampling (AGS), which consists in sampling the RBM's visible units while keeping the hidden units fixed and vice-versa, in an alternate manner, until the Monte Carlo Markov Chain equilibrates. To increase the probability of sampling high log-likelihood sequences we can sample from $p(\boldsymbol{s})^2$ instead of $p(\boldsymbol{s})$ using the so-called duplication trick [97]. We write

$$p(\boldsymbol{s})^2 = \left(\int d\boldsymbol{h}\, p(\boldsymbol{s}, \boldsymbol{h}) \right)^2 = \int d\boldsymbol{h}_1 \int d\boldsymbol{h}_2\, p(\boldsymbol{s}, \boldsymbol{h}_1)\, p(\boldsymbol{s}, \boldsymbol{h}_2) \,. \tag{4.7}$$

This squared likelihood distribution can therefore be sampled with standard AGS after duplication of the hidden layer of the trained RBM model.

The average hidden unit μ's activity for a given sequence $\mathbf{s}$ is defined as $\langle h_\mu \rangle = \int d\mathbf{h}\, h_\mu \, p(\mathbf{h}|\mathbf{s})$. Note that $\langle h_\mu \rangle$ only depends on the sequence $\mathbf{s}$ through the input $I_\mu = \sum_i w_{\mu i}(\mathbf{s}_i)$. When the average activity is close to 0, the corresponding hidden unit has vanishing contribution to the sequence log-likelihood, while for both large negative or positive values of average activity the contribution of the hidden unit to the log-likelihood is positive. Therefore the sign of the weights $w_{i\mu}$ assigned to a particular sequence motif is not indicative itself of the presence or absence of a given pattern, as the contribution in $p(\mathbf{h}|\mathbf{s})$ depends on the product $h_\mu I_\mu$ and can only be null or positive.

4.6.3 Design of single-loop aptamers with RBM

The RBM-SU distribution $p(\boldsymbol{s})$ can be sampled to generate sequences $\boldsymbol{s}$ of interest, and test the validity of the model. We describe below how we generated sequences in Table 4.

4.6.3.0.1 Determination of Threshold.

We fix the threshold, which allows us to distinguish good from bad binders based on their log-likelihoods to minimize the number of misclassified sequences among the preliminary set of sequences given in Table 27. As a range of possible values are possible, we actually take the median of this interval.

4.6.3.0.2 Sequences with High Likelihoods.

We first sample through AGS (see Sec. 4.6.2) 4000 sequences from $p(\boldsymbol{s})$ and from $p(\boldsymbol{s})^2$. We then choose 10 among these sequences (named r9 to r17 and r22, r23 in Table 4), which have both high log-likelihood and large distances (numbers of different nucleotides) to round

8 data. In practice these sequences are at Hamming distance 1 or 2 from the closest sequences in the original dataset, since further away sequences have substantially lower log-likelihoods. All 10 generated sequences are experimentally confirmed to be good binders (Table 4), and are indicated as green lines in Fig. 23e.

4.6.3.0.3 Sequences with Critical Mutations for Binding/Non-Binding Status.

We next use our RBM to predict critical mutations capable of changing the binder/non-binder status of aptamers. First we exhaustively look for the smallest possible number of mutations leading to a substantial decrease of the log-likelihood of known good binders. In particular, sequence r1 has 1 mutation with respect to a control sequence that we tested for binding (named d10 in Table 27), r2 and r3 are both 1 mutation away from, respectively ThB and ThC, both identified as good binders in Ref. [210]. All these generated sequences are confirmed to be unable to stably bind thrombin after this single-point mutation (Table 4 and Fig. 24) and they correspond to red vertical lines to the left of the threshold in Fig. 23e. All these mutations removed a G from the sequence, and G nucleotides are necessary to form G-quadruplex motifs, known to be important for thrombin aptamers. To show that our RBM can also identify other positions in the aptamer that are key to thrombin binding, we also design two more sequences, r4 and r5, which have 2 mutations with respect to aptamers found in the SELEX dataset and validated as good binders (respectively, d10 and d18, see Table 27). The mutations are again chosen so that the log-likelihood is decreased as much as possible, but without removing G nucleotides from the original sequences. We find the sequences lost their ability to bind thrombin after the 2 mutations, as predicted by the RBM (Fig. 24), so they correspond to two vertical red lines to the left of the threshold in Fig. 23e.

4.6.3.0.4 Sequences in Dataset with Mismatches between Counts and Log-Likelihoods.

We further test the performance of the RBM model by searching for sequences with (1) relatively low log-likelihoods but with large numbers of counts (139 or more, see Tab. 29) in the SELEX experimental data from Ref. [210], of for sequences with high log-likelihood but with few counts (11 or less, see Tab. 29). The sequences chosen in case (1) are r6, r7, r18, r19 (see Table 4); one of them (r6) is below, and the other 3 are slightly above the identified log-likelihood threshold. Sequences chosen in case (2) are r8, r9, r20, r21 (Table 4). The RBM predictions are confirmed in all cases but one (r18), which corresponds to a red vertical line at the right of the threshold in Fig. 23e.

4.6.3.0.5 Sequences Sharing a Rare Mutation with ThA, a Strong Exosite-II binder.

Last of all we design *de novo* sequences (r24 to r27 in Table 4) under the following two-fold criterion. First these sequences are required to have Adenine in position 17, which is uncommon in the training dataset (A is the second least common nucleotide in that position, being present in about 13% of the sequences in round 8; it is found in ThA, which strongly binds exosite II). Second, the sequences are required to have large log-likelihoods, exceeding the threshold value. Remarkably, the only non-binder among r24-r27 is the one with lowest log-likelihood, r27. However, while mutating away from A in ThA change the binding specificity from exosite II to I (Fig. 26) sequences r24 to r27 are all exosite-I binders, showing that the presence of A17 is not sufficient for exosite-II specificity.

4.6.4 Thrombin Binding Assay

All RBM designed sequences were first assessed for their ability to bind either of the cationic exosites of human alpha-thrombin. Each sequence was placed as the loop of a 18 bp stem loop with the full sequences reported in the Table 23. As done previously [210], we used a 5% native gel shift assay to qualitatively assess the binding of each stem loop to thrombin. Each sequence was tested with two gel lanes, the first lane always corresponding to the stem loop without thrombin and the second lane consisting of equimolar amounts (500 nM) of thrombin and the stem loop. The presence of an upper band, consisting of a stem loop bound to thrombin complex, in the second lane indicates a binding sequence. Sequences without the upper band (nonbinding sequences) either very weakly interact with thrombin, characterized by a smear but no band in the second lane, or do not interact with thrombin at all matching their negative control lane. Sequences ThA and ThD were selected from the previous study as positive controls for their high affinity for thrombin and known binding sites [210]. Results for all RBM generated sequences are shown in Figure 24 and summarized in Table 4. Results for all DCA generated sequences are shown in Fig. 65 and summarized in Table 27. To quantify the interaction of the stem loop and thrombin, we tested both control sequences independently and together in varying concentrations of thrombin (Fig. 55). The results clearly indicate the stem loop/thrombin band occurs from a 1:1 interaction of thrombin and each stem loop, and the simultaneous binding of two stem loops on opposite exosites of thrombin downshifts the stem loops/ thrombin band from the singular case.

A secondary band prominently appeared among four of the sequences during the binding assay, (r12, r15, r16, and r22). These sequences showed no binding to Thrombin at first. Upon further investigation, the secondary band was found to most likely be a dimer state of the DNA loop from interaction of the G-quartet motifs. The four sequences have a higher G-content than all other RBM-generated sequences. Additionally, a G-quadruplex dimer would require K^+ cations to form, indicating a testable transition from the single loop to dimer state. The sequences' Thrombin binding ability was re-assessed by the same experiment, with two small changes. The first was remaking the DNA samples without K^+ in their buffer, so their transition from single stem loop to a dimer state could be observed[224]. The second change was the heating the DNA samples to 90 □ for 5 minutes before immediately chilling them in ice. Samples (r12, r15, r16, r22) in Figure 24 show the results of this final experiment, with all dimer-susceptible sequences showing an ability to bind Thrombin.

Accordingly, we classify these sequences as binders and suggest their absence from the original dataset is due to G-quadruplex dimer formation during the original SELEX procedure. A clear shift from the monomer state in 1x TAE Mg^{2+} (no K^{+}) buffer (lane 1) to the dimer state upon addition of buffer with K+ (lane 2) is also observed for all dimer-susceptible sequences. Note this transition still contains some fraction of the dimer state in lane 1 where the sample contains no K+. This is due to presence of K+ in the gel matrix itself as well as the running buffer.

4.6.5 Exosite Binding Assay

RBM-generated sequences that were able to bind to thrombin were tested to determine which exosite (I or II) of thrombin they bind to. Each aptamer was pre-incubated with thrombin for 30 minutes at 25 □at an equimolar ratio in two separate samples. Small amounts (1/10th the pre-incubated strand) of fluorescent labeled exosite II binder ThA [210] was added to the first sample and fluorescent labeled exosite I binder ThD to the second. Using the same strategy as our thrombin binding assay, our samples were run in a 5% native gel with 5 mM K^{+} for proper DNA/thrombin binding. If the pre-incubated strand bound the same exosite as the fluorescent strand, the thrombin/fluorescent strand complex band would be observed in the same position as seen in our thrombin binding assay. However, if the pre-incubated strand bound the opposite exosite as the fluorescent strand, both strands bind thrombin causing the same downward shift as observed for our exosite verified control strands mixing (Fig. 55). Accordingly, sequences with no binding affinity to thrombin matched control samples with no test strand. By comparing the outcome of both lanes for a sample we are able to firmly assign the binding site of our test sequences. The gel results are shown in Fig. 25 and summarized in Table 4.

4.7 Acknowledgement

We thank G. Mayer and M. Famulok for helpful discussions and comments. A.D.G. would like to thank G. Isacchini and J. Fernandez-de-Cossio-Diaz for helpful discussions and comments. This material is based upon work supported by the National Science Foundation under Grant No. 2155095. We acknowledge use of the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number TG-BIO210009. We also acknowledge Research Computing at Arizona State University for providing HPC resources that have contributed to the research results reported within this paper. S.C., A.D.G. and R.M. are partially supported by the ANR-17 RBMPro CE30-0021-01 and ANR-19 Decrypted CE30-0021-01 grants.

References

[18] Anton Schmitz et al. “A SARS-CoV-2 spike binding DNA Aptamer that inhibits Pseudovirus infection by an RBD-independent mechanism”. In: *Angewandte Chemie International Edition* 60.18 (2021), pp. 10279–10285.

[83] John Jumper et al. “Highly accurate protein structure prediction with AlphaFold”. In: *Nature* 596.7873 (2021), pp. 583–589.

[91] Geoffrey E. Hinton. “Training Products of Experts by Minimizing Contrastive Divergence”. In: *Neural Computation* 14.8 (2002), pp. 1771–1800.

[94] Faruck Morcos et al. “Direct-coupling analysis of residue coevolution captures native contacts across many protein families”. In: *Proceedings of the National Academy of Sciences* 108.49 (2011), E1293–E1301.

[97] Jérôme Tubiana, Simona Cocco, and Rémi Monasson. “Learning protein constitutive motifs from sequence data”. In: *eLife* 8 (2019), e39397.

[99] Barbara Bravi et al. “RBM-MHC: A Semi-Supervised Machine-Learning Method for Sample-Specific Prediction of Antigen Presentation by HLA-I Alleles”. In: *Cell Systems* 12.2 (2021), pp. 195–202.

[175] Günter Mayer et al. “From selection to caged aptamers: identification of light-dependent ssDNA aptamers targeting cytohesin”. In: *Bioorganic & Medicinal Chemistry Letters* 19.23 (2009), pp. 6561–6564.

[176] Sabine Lennarz et al. “Selective Aptamer-Based Control of Intraneuronal Signaling”. In: *Angewandte Chemie* 127.18 (2015), pp. 5459–5463.

[177] Anna Schüller et al. “Activation of the glmS ribozyme confers bacterial growth inhibition”. In: *Chembiochem* 18.5 (2017), pp. 435–440.

[178] Malte Rosenthal, Franziska Pfeiffer, and Günter Mayer. “A Receptor-Guided Design Strategy for Ligand Identification”. In: *Angewandte Chemie International Edition* 58.31 (2019), pp. 10752–10755.

[179] Alvaro Dar□ıo Ortega et al. “A synthetic RNA-based biosensor for fructose-1, 6-bisphosphate that reports glycolytic flux”. In: *Cell Chemical Biology* (2021).

[180] Christian Renzl, Ankana Kakoti, and Günter Mayer. “Aptamer-Mediated Reversible Transactivation of Gene Expression by Light”. In: *Angewandte Chemie* 132.50 (2020), pp. 22600–22604.

[181] Valeriy Domenyuk et al. “Poly-ligand profiling differentiates trastuzumab-treated breast cancer patients according to their outcomes”. In: *Nature Communications* 9.1 (2018), pp. 1–9.

[182] Laia Civit et al. “Systematic evaluation of cell-SELEX enriched aptamers binding to breast cancer cells”. In: *Biochimie* 145 (2018), pp. 53–62.

[183] Tassilo Hornung et al. “ADAPT identifies an ESCRT complex composition that discriminates VCaP from LNCaP prostate cancer cell exosomes”. In: *Nucleic Acids Research* 48.8 (2020), pp. 4013–4027.

[184] Jiehua Zhou and John J Rossi. “Cell-type-specific, aptamer-functionalized agents for targeted disease therapy”. In: *Molecular Therapy-Nucleic Acids* 3 (2014), e169.

[185] Craig Tuerk and Larry Gold. “Systematic evolution of ligands by exponential enrichment: RNA ligands to bacteriophage T4 DNA polymerase”. In: *Science* 249.4968 (1990), pp. 505–510.

[186] Andrew D Ellington and Jack W Szostak. “In vitro selection of RNA molecules that bind specific ligands”. In: *Nature* 346.6287 (1990), pp. 818–822.

[187] Mayte Sola et al. “Aptamers against live targets: is in vivo SELEX finally coming to the edge?” In: *Molecular Therapy-Nucleic Acids* 21 (2020), pp. 192–204.

[188] Daniela Proske et al. “Aptamers—basic research, drug development, and clinical applications”. In: *Applied Microbiology and Biotechnology* 69.4 (2005), pp. 367–374.

[189] Jan P Elskens, Joke M Elskens, and Annemieke Madder. “Chemical modification of aptamers for increased binding affinity in diagnostic applications: Current status and future prospects”. In: *International Journal of Molecular Sciences* 21.12 (2020), p. 4522.

[190] Sofia D’Souza, KV Prema, and Seetharaman Balaji. “Machine learning models for drug–target interactions: current knowledge and future directions”. In: *Drug Discovery Today* 25.4 (2020), pp. 748–756.

[191] Raphael JL Townshend et al. “Geometric deep learning of RNA structure”. In: *Science* 373.6558 (2021), pp. 1047–1051.

[192] Pauric Bannigan et al. “Machine learning directed drug formulation development”. In: *Advanced Drug Delivery Reviews* (2021).

[193] Jan Hoinka et al. “Identification of sequence–structure RNA binding motifs for SELEX-derived aptamers”. In: *Bioinformatics* 28.12 (2012), pp. i215–i223.

[194] Jia Song et al. “A sequential multidimensional analysis algorithm for aptamer identification based on structure analysis and machine learning”. In: *Analytical Chemistry* 92.4 (2019), pp. 3307–3314.

[195] Khalid K Alam, Jonathan L Chang, and Donald H Burke. “FASTAptamer: a bioinformatic toolkit for high-throughput sequence analysis of combinatorial selections”. In: *Molecular Therapy-Nucleic Acids* 4 (2015), e230.

[196] Timothy L Bailey et al. “The MEME suite”. In: *Nucleic Acids Research* 43.W1 (2015), W39–W49.

[197] Peng Jiang et al. “MPBind: a Meta-motif-based statistical framework and pipeline to Predict Binding potential of SELEX-derived aptamers”. In: *Bioinformatics* 30.18 (2014), pp. 2665–2667.

[198] Qingtong Zhou et al. “Searching the Sequence Space for Potent Aptamers Using SELEX in Silico”. In: *Journal of Chemical Theory and Computation* 11.12 (2015), pp. 5939–5946.

[199] Qingtong Zhou et al. “Exploring the Mutational Robustness of Nucleic Acids by Searching Genotype Neighborhoods in Sequence Space”. In: *The Journal of Physical Chemistry Letters* 8.2 (2017), pp. 407–414.

[200] Abe Pressman et al. “Analysis of in vitro evolution reveals the underlying distribution of catalytic activity among random sequences”. In: *Nucleic Acids Research* 45.14 (2017), pp. 8167–8179.

[201] Abe D. Pressman et al. “Mapping a Systematic Ribozyme Fitness Landscape Reveals a Frustrated Evolutionary Network for Self-Aminoacylating RNA”. In: *Journal of the American Chemical Society* 141.15 (2019), pp. 6213–6223.

[202] Peter K Koo et al. “Inferring sequence-structure preferences of RNA-binding proteins with convolutional residual networks”. In: *BioRxiv* (2018), p. 418459.

[203] Jan Zrimec et al. “Learning the regulatory code of gene expression”. In: *Frontiers in Molecular Biosciences* 8 (2021).

[204] Peter K Koo and Matt Ploenzke. “Deep learning for inferring transcription factor binding sites”. In: *Current Opinion in Systems Biology* 19 (2020), pp. 16–23.

[205] Drew H Bryant et al. “Deep diversification of an AAV capsid protein by machine learning”. In: *Nature Biotechnology* 39.6 (2021), pp. 691–696.

[206] Simona Cocco et al. “Inverse statistical physics of protein sequences: a key issues review”. In: *Reports on Progress in Physics* 81.3 (2018), p. 032601.

[207] Eleonora De Leonardis et al. “Direct-Coupling Analysis of nucleotide coevolution facilitates RNA secondary and tertiary structure prediction”. In: *Nucleic Acids Research* 43.21 (2015), pp. 10444–10455.

[208] William P Russ et al. “An evolution-based model for designing chorismate mutase enzymes”. In: *Science* 369.6502 (2020), pp. 440–445.

[209] Qin Zhou et al. “Global pairwise RNA interaction landscapes reveal core features of protein recognition”. In: *Nature Communications* 9.1 (2018), pp. 1–10.

[210] Yu Zhou et al. “DNA-Nanoscaffold-Assisted Selection of Femtomolar Bivalent Human alpha-Thrombin Aptamers with Potent Anticoagulant Activity”. In: *ChemBioChem* 20.19 (2019), pp. 2494–2503.

[211] Tijmen Tieleman. “Training Restricted Boltzmann Machines Using Approximations to the Likelihood Gradient”. In: *Proceedings of the 25th International Conference on Machine Learning*. ICML ’08. Helsinki, Finland: Association for Computing Machinery, 2008, pp. 1064–1071.

[212] Richard A. Neher and Boris I. Shraiman. “Statistical genetics and evolution of quantitative traits”. In: *Reviews of Modern Physics* 83.4 (2011), pp. 1283–1300.

[213] Daniel L Hartl, Daniel E Dykhuizen, and Antony M Dean. “Limits of Adaptation: The Evolution of Selective Neutrality”. In: *Genetics* 111.3 (1985), pp. 655–674.

[214] KAILLATHE Padmanabhan et al. “The structure of alpha-thrombin inhibited by a 15-mer single-stranded DNA aptamer.” In: *Journal of Biological Chemistry* 268.24 (1993), pp. 17651–17654.

[215] Franziska Pfeiffer et al. “Systematic evaluation of error rates and causes in short samples in next-generation sequencing”. In: *Scientific Reports* 8.1 (2018), pp. 1–14.

[216] Andreas Wagner. *Robustness and evolvability in living systems*. Princeton university press, 2013.

[217] Matteo Bisardi et al. “Modeling sequence-space exploration and emergence of epistatic signals in protein evolution”. In: *Molecular Biology and Evolution* 39.1 (2022), msab321.

[218] Tzu-Fang Lou et al. “Integrated analysis of RNA-binding protein complexes using in vitro selection and high-throughput sequencing and sequence specificity landscapes (SEQRS)”. In: *Methods* 118 (2017), pp. 171–181.

[219] Christoph M Hammers and John R Stanley. “Antibody phage display: technique and applications”. In: *The Journal of Investigative Dermatology* 134.2 (2014), e17.

[220] Titus Kretzschmar and Thomas Von Rüden. “Antibody discovery: phage display”. In: *Current Opinion in Biotechnology* 13.6 (2002), pp. 598–602.

[221] Luca Sesta et al. “AMaLa: Analysis of Directed Evolution Experiments via Annealed Mutational Approximated Landscape”. In: *International Journal of Molecular Sciences* 22.20 (2021), p. 10908.

[222] J. Tubiana and R. Monasson. “Emergence of Compositional Representations in Restricted Boltzmann Machines”. In: *Phys. Rev. Lett.* 118 (13 2017), p. 138301.

[223] Clément Roussel, Simona Cocco, and Rémi Monasson. “Barriers and dynamical paths in alternating Gibbs sampling of restricted Boltzmann machines”. In: *Physical Review E* 104.3 (2021), p. 034109.

[224] Mateusz Kogut, Cyprian Kleist, and Jacek Czub. “Why do G-quadruplexes dimerize through the 5’-ends? Driving forces for G4 DNA dimerization examined in atomic detail”. In: *PLoS Computational Biology* 15.9 (2019), e1007383.

Chapter 5

FUTURE WORK AND CONCLUSION

This chapter focuses on two projects that are nearing their eventual realization. I have been working on them for the last ten months and hope to see them published in the next couple of months before I leave for (real or imagined) greener pastures.

5.1 Ongoing Coarse-Grained MD Research

DNA crystallization for the purpose of protein structure determination was the original motivation for the development of DNA nanotechnology [29]. Today, DNA crystals and other DNA lattices are seeing increased attention because of the promise of creating ordered nanoscale materials [225]. Possible developments of the technology can bring advances in energy storage, optics, catalysis, metamaterials, information storage devices, and other electronics [226, 227]. DNA lattices demonstrate the ability to have defined geometry and incorporate guest molecules [228]. The end result being a macroscopic 3D material with unprecedented 3D control. Here, we develop a multiscale model capable of studying the assembly of a large amount of DNA origami for exploring DNA origami lattice systems.

5.1.1 A Coarser Representation

Despite the coarse-grained nature of oxDNA, there are many DNA Nanotechnology processes that cannot be modeled using this software, due to the large number of DNA origami in the system. For example, self-assembly of DNA origami into larger constructs cannot be achieved in a timely manner in oxDNA without supercomputer-level resources. Systems of interest include DNA crystal lattices such as the tetrastack lattice seen in Figure 28. The tetrastack lattice is highly valued because of its omnidirectional photonic bandgap which would allow for optoelectronic devices including semiconductor lasers and nonlinear optical switches[229].

Current approaches for modeling assembly of DNA origami represent the DNA origami as a single particle, with the sequence-specific handles for binding represented as “patches”. This is the “patchy particle” model that has seen widespread use in modeling large assemblies, including theoretical DNA nanocrystals and can be seen in Figure 29c.

However, many details of the system are entirely ignored using the patchy particle model, particularly the individual fluctuations of the DNA origami, which can have a major effect on the availability of the patches during binding. As an intermediate representation, groups of nucleotides can be represented as single particles; in essence, coarse-graining the oxDNA representation of DNA. Making use of a network model we can approximately capture the dynamics of the structure as simulated by oxDNA. DNA strands expected to hybridize with one another over the course of the simulation will be represented at the oxDNA nucleotide level to retain the thermodynamics of the oxDNA model.

To give this representation as much flexibility as possible, the user will be able to define how many particles represent each origami allowing for a scale of representations. Different coarse-grained representations of the same icosahedral DNA origami can be seen in Figure 29b. The trade-off is that the representation cannot have any chemically defined interactions, since their sizes and topology can be different across representations of the same system. Instead, each representation will need to be fit to reproduce the average fluctuations of the target oxDNA system.

5.1.1.1 Model Topology

The decision of where to place particles to represent a group of other smaller particles is nontrivial. In our use case we must be able to capture the overall shape of the structure to avoid having complexes that can overlap due to too low of a resolution. The choice of each

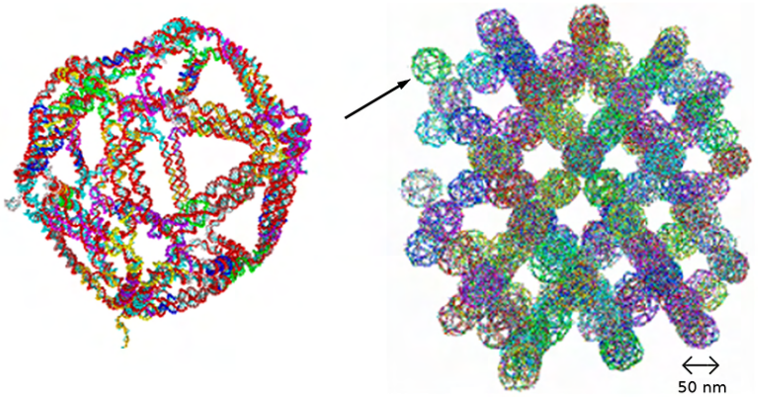


Figure 28. Assembly of icosahedral DNA origami into a tetrastack DNA crystal lattice

particle location is further hampered by the necessity of having locations to attach the handles responsible for binding. Each nucleotide in the original system can only be assigned to one of the larger coarse-grained particles. With this condition in mind, it is a natural choice to use the k-means algorithm to partition our oxDNA system into larger particles. The coordinates of each nucleotide in the oxDNA system are fed into the algorithm, which then finds a user-specified number of "mean" positions that reduces the variance of the coordinate data [230]. The algorithm then follows an iterative procedure known as expectation-maximization where at each step the nucleotides are assigned to a cluster based off their distance to the "mean" positions and then the "mean" positions are recalculated from the positions of the nucleotides belonging to the cluster. The process ends when nucleotides are no longer being re-assigned to different clusters, i.e. the process converges. To bias the algorithim towards placing means in positions where handles can be easily attached, we can add multiple copies of coordinates where our handles should be attached to our coordinate list we give as input to the algorithm. Using this method results in usable coarse-grain topologies made up of as many particles as specified by the user.

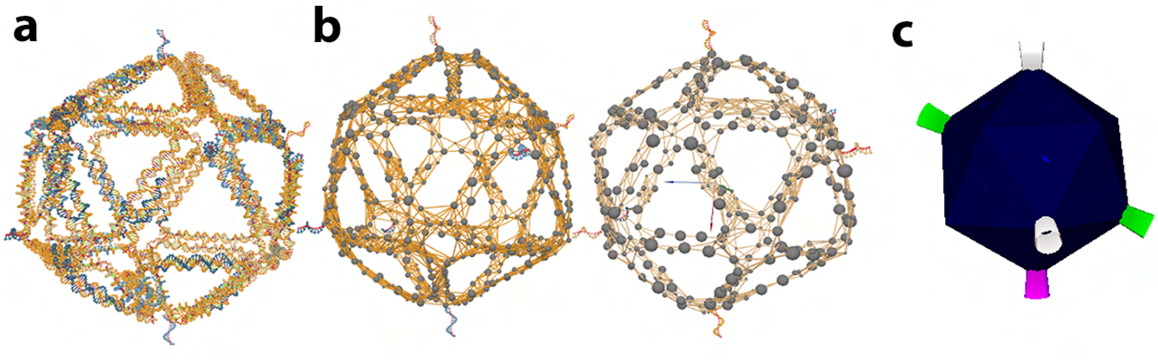


Figure 29. (a) oxDNA visualization of icosahedral DNA origami. (b) Two different representations using the proposed network model of an icosahedral DNA. (c) The patchy particle model of a DNA origami.

5.1.1.2 Model Parameterization

The goal of this model is to approximately capture the dynamics of DNA origami simulated in the oxDNA model, at a fraction of the computational cost. Starting from an oxDNA trajectory of our target system we can calculate the target dynamics of our coarse grain system by calculating the deviations of each cluster's group of nucleotides at each timestep. In practice, the center of mass position of the list of nucleotides that represent a coarse-grain particle is calculated at each time step and used to calculate a deviation from the mean structure of the coarse-grained system. This nets a fluctuation profile known as the B factor. We can then use these calculated B factors as the target dynamics for the parameterization of the coarse-grained system. The potential function used to drive the dynamics of the system is just a simple spring potential attached to between each pair of coarse-grained particles within a certain cutoff distance from one another. If the spring constants were homogenous among all particle pairs, this model would simply be an Anisotropic Network Model. However, in our parameterization scheme we use a modified version of the Heterogenous Anisotropic Network Model introduced by Lu *et al.* [137]. The parameterization procedure starts from the ANM best fit of the target B factors and iteratively changes the spring constant values to reproduce those of our target B factors as close as possible. This process is implemented using a modified version of the original script by Lu et al. [137].

5.1.1.3 Attachment of Handles

Once our coarse-grained model's topology and spring connections are defined, the only step left before simulation of the system is to attach any handles used to bind other DNA origami (in a coarse-grained representation or not). In our model, handles are represented by the nucleotide level oxDNA model and then attached to a coarse-grained particle using external forces, in particular another spring potential. Using this method we simplify the computational cost, but retain the thermodynamics and kinetics of the DNA handles.

5.1.1.4 Current Usage

This model has yet to be published but is in the final rounds of testing. We are testing it on the icosahedral DNA origami structures that which were designed to form a tetrastack lattice. However, the experiments have to date not been successful. Using this model we plan on investigating probable causes for this.

5.2 Aptamer Analysis with Unsupervised Models

5.2.1 Aptamer Dataset Topology

An aptamer is defined as any molecule that tightly binds a target molecule. Aptamers can have both therapeutic and diagnostic potential depending on the target they bind. Of particular interest are aptamers made of DNA, RNA, peptides, or antibodies due to their lack of immunogenicity [45, 231].

In order to generate novel sequences that tightly bind a target, popular methods use large libraries of random sequences which are then exposed to the target. DNA and RNA aptamers are typically generated using a method known as Systematic Evolution of Ligands by EXponential enrichment (SELEX) which is discussed extensively in 4. The data produced by high throughput SELEX (HT-SELEX) consists of sequencing data of enriched sequences.

In vivo selection of antibodies can be done using phage display. Phage display uses a bacteriophage with a modified coat protein to display a scaffold protein with a variable region. In the production of antibodies, the most commonly used phage is the M13 bacteriophage. Variants of M13 add the genes of antibody fragments to different coat proteins with the most common being the pIII due to its flexibility and ability to display large proteins [232]. Typically, the CDR3 region of the displayed antibody is the primary contributor in determining antibody/ target compatibility [232, 233]. Similar to SELEX, *in vivo* phage display is performed in rounds and yields sequencing data of enriched antibody fragments [234].

5.2.1.1 Challenges with Aptamer Datasets

Analysis of aptamer sequencing datasets is difficult due to a few primary challenges

- **Datasets can be rife with sequencing errors**. The sequencing methods, biopolymer, and enrichment methods are key contributors making each dataset's error unique.
- **Relatively few sequences in the datasets tightly bind the target of interest**. It is common for the majority of data to consist of weak or nonspecific binders [235].
- **Large populations of a single sequence does not correlate to a sequence's binding ability**. Experimental artifacts (ex. PCR bias) or sticky sequences can result in large populations of particular sequences that do not tightly bind the target [236].

Existing analysis methods for aptamers broadly categorize into three categories: motif-finding, cluster-finding, and machine learning[235]. Motif-finding methods including MEME [196] and AptaMotif [193] search for common motifs (a short repeated pattern) in the aptamer dataset. Cluster-finding methods including AptaCluster [237] and FASTAptamer

[195] find groups of related sequences in the aptamer dataset based off their distance to one another. Machine learning approaches include DNN supervised methods [238] such as Variational AutoEncoders [96] that train a model for classification or regression based off labels derived from the sequence data.

All of the previously mentioned analysis tools are sensitive to one of more challenges listed above. For high quality analysis, preprocessing of the dataset is absolutely necessary.

5.2.1.2 Dataset Preprocessing

Raw high-throughput sequencing data contains reads that have been truncated and contain sequencing errors. Truncations comprise a small percentage of the total dataset (2%) and are discarded with little to no effect on the resulting data [215]. Sequencing errors can be systematic, random, or sequence-specific with different sequencing machines and experimental conditions returning distinct compositions of error [239, 215]. Estimation and correction of error on high-throughput sequencing is of considerable interest for single nucleotide polymorphism, halotype, and other genomic studies [239].

However, most error correction methods are not applicable to aptamer datasets due to the short sequence length and over-represented motifs [215]. Instead of attempting to find and fix errors in the dataset, selecting a subset of the data that we can most likely trust is our preferred strategy. Our key assumptions can be summarized as follows. Sequences are less likely to be sequencing errors if they are present across multiple rounds of sequencing and are read more than once in a round. These assumptions are based off the fact that single base substitutions are unlikely to occur consistently in the same place for many reads across consecutive rounds.

Following our assumptions, we select all sequences that are present in at least two rounds of selection with more than one read in each round as our dataset. Reads are then normalized across rounds by dividing the read by the total reads per million (as done in [195]). An enrichment score for each pairs of rounds in the dataset is calculated for each sequence. A distance average is then calculated from all enrichment scores made up of rounds a specified distance away from each other. A final average is taken over the distance averages to return a final enrichment average. Taking an enrichment average has the benefit of prioritizing sequences that were enriched throughout the entire selection process, not just between two selected rounds.

Thus far, we have applied our dataset preprocessing methods to SARS-CoV-2 DNA aptamers, cancer exosomes, and phage display datasets. We are currently using the models discussed in the following section to learn about the functional features of these sequences and generate new sequences via sampling and rational design.

5.2.2 New Models for Aptamer Analysis

5.2.2.1 Extensions to the RBM

Due to their stellar performance, the RBM has been used extensively. The RBM has been modified extensively in different regards: the model's topology, learning algorithm, and sampling methods being the main focuses of these alterations. The Conditional RBM worked by adding connections between the current and previous states of both the visible and hidden units to efficiently model time dependent data[240, 241, 242]. The Convolutional RBM[243] worked by changing the connecting weight matrix to be convolutional filters which removes some of the positional dependence of the hidden layer in learning different features. It has been used for human behavior recognition on video data [244] and for sound classification [245]. Adding a classification layer to an RBM resulted in the Discriminative and Hybrid Discrimnative RBM models[246]. These models achieved very good performance on classification datasets such as MNIST and changed the objective function of the model to be exactly differentiable. Combining the two previous mentioned approaches, a Convolutional Classification RBM has been used in the analysis of lung CT data [247]. As a generalization of the DRBM and HDRBM, the Supervised RBM adds a third layer to the model and is able to perform regression as well as classification on labeled datasets. The downside is that typical methods for sampling the hidden layer can no longer be used because the hidden layer would need to be sampled sequentially which is not ideal, especially for large models. They instead trained the RBM using a variational method which minimizes the evidence lower bound (ELBO) [248]. Other changes to the RBM's architecture have included different activation functions on the hidden units [249] as well as different visible and hidden unit types such as gaussian, multinomial, and rectified linear units [250].

Improved Markov Chain Monte Carlo sampling methods have been a focus for many groups. One of the most popular methods, persistent contrastive divergence (PCD) saves the chain at each epoch and initializes the next round of Gibbs sampling from the saved chain[211]. A proposed improvent, parallel tempering (PT) increases the mixing of the chain by simulating many copies of the chain at different temperatures and allowing them to exchange states[251]. The lowest temperature chain is used as the generated sample for the free energy calculation.

The above list of modifications is far from exhaustive. A good review on many of the different forms of RBMs can be found here [252].

5.2.2.2 Pooling Convolutional Restricted Boltzmann Machine

Here we present a Convolutional Restricted Boltzmann Machine, first introduced in [243], that uses the same hidden and visible layers as defined by Tubiana *et al.* [97]. In order to completely destroy the location dependence of each feature for applications on unaligned and different sized data, we use a modified max pool layer on the outputs of the convolutional layer. The modification of the max pool layer is slight, where it returns either the most positive value or most negative value by comparison of the absolute values. By returning the max positive or max negative value, we are able to use the dReLU potential which applies different nonlinearities to positive and negative inputs. During sampling and

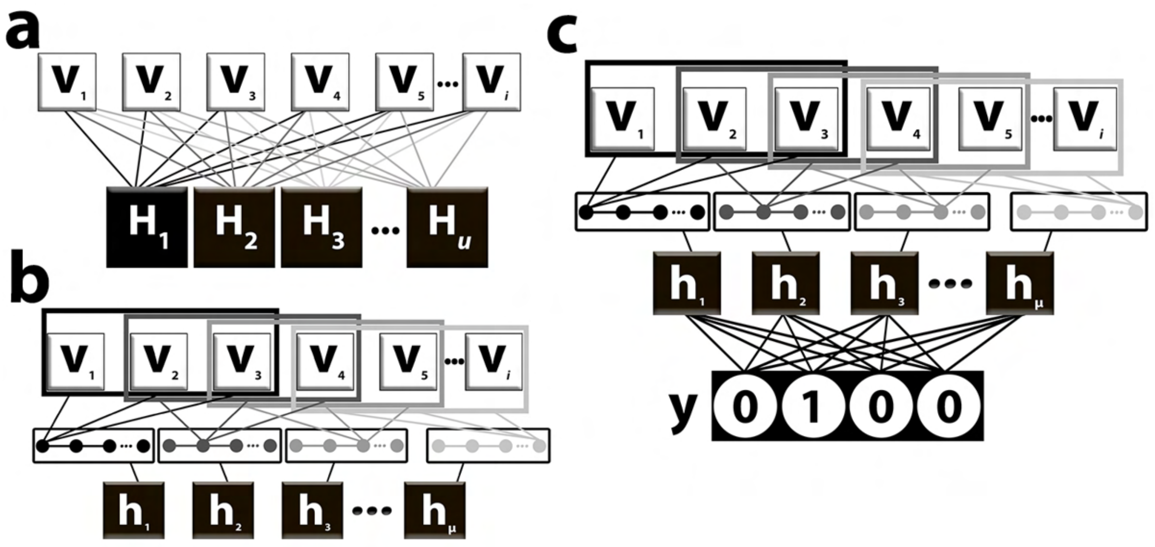


Figure 30. a) Model Topology of Restricted Boltzmann Machine (RBM). b) Model Topology of Convolutional RBM with pooling layer. c) Model topology of pooling Convolutional RBM with classification layer.

training, the indices of the max values are stored for computing the transpose pool operation and used to sample the visible layer. We also introduce the ability to use different convolution sizes for different hidden units.

We term this model the Pooling Convolutional Restricted Boltzmann Machine (PCRBM). A depiction of the model's topology can be seen in Figure 30b. There are i visible units and μ hidden units. The energy function of the model can be defined as in Equation 5.1 where $\mathcal{U}$ represents the dReLU potential, $*$ represents a convolution operation, $\otimes$ represents the transpose convolution operation, single square brackets [ ] denote a modified max pool operation, and double square bracket [[ ]] denotes the transpose pool operation.

$$E(v,h) = -\sum_i g_i(v_i) + \sum_\mu \mathcal{U}_\mu(h_\mu) - \sum_\mu h_\mu \left[W_\mu * v\right] \tag{5.1}$$

Conditional probability distributions can be derived using Bayes Theorem. The probability of a hidden node configuration is given by Equation 5.3 with the visible input given by equation 5.4 while the probability of a visible node configuration is given by Equation 5.5

with input from the hidden layer given by equation 5.2. Updates of hidden units and visible layers can be performed with Gibbs sampling using these equations. Due to the conditional independence of the visible and hidden units, updates of individual nodes can be performed in parallel within their layer.

$$I^h_\mu(v) = [W_\mu * v] \tag{5.2}$$

$$P(h_\mu|v) \propto \exp\left(-\mathcal{U}_\mu(h_\mu) + h_\mu I^h_\mu(v)\right) \tag{5.3}$$

$$I^v_i(h) = \left(\sum_\mu W_\mu \, [[h_\mu]]\right)_i \tag{5.4}$$

$$P(v_i|h) \propto \exp\left(g_i(v_i) + v_i I^v_i(h)\right) \tag{5.5}$$

Likewise the marginal distribution of the visible layer can be defined by equation 5.7 where Γ is defined as the cumulant generating function of the hidden unit probability distribution function which is defined by equation 5.8.

$$P(v) = \int \prod_\mu dh_\mu P(v,h) \tag{5.6}$$

$$P(v) \propto \exp\left(\sum_i g_i(v_i) + \sum_\mu \Gamma_\mu(I_\mu)\right) \tag{5.7}$$

$$\Gamma_\mu = \log\left[\int dh_\mu \exp\left(-\mathcal{U}_\mu(h_\mu) + h_\mu I^h_\mu\right)\right] \tag{5.8}$$

Minimizing the free energy of the data ($-\log(P(v))$) is the learning objective of the RBM (discussed in 1.2.3.1) and can be performed with a few different methods. In our implementation, contrastive divergence (CD), persistent contrastive divergence (PCD), and parallel tempering (PT) are implemented for sampling the positive phase of the free energy equation.

Cost Function and Regularization

For our CRBM we maximize the following function:

$$\langle log\left(P(v)\right)\rangle_{\text{MSA}} = \frac{\sum_b w_b \log\left(P(v_b)\right)}{\sum_b w_b} \tag{5.9}$$

Sequences can be weighted (w_b) by sequence identity to avoid over-fitting [98] or by their copy number to better represent the fitness landscape of aptamers generated via SELEX [253]. The sequence weights provide a mechanism to tweak the gradients of the data. Regularization terms on the weights is given by equation 5.10

$$W_{\text{reg}} = \lambda_1^2 \sum_{\mu} \frac{\left(\sum_{ic,kx,ky} |W_{\mu,ic,kx,ky}|\right)^2}{2ic\ kx\ ky} \tag{5.10}$$

where ic, kx, and ky are the input channels (most purposes should be 1), the kernel size on the visible node dimension of the convolution, and the kernel size on the category dimension of the convolution.

Additionally a distance regularization term was implemented to promote each filter to learn a unique feature. This regularization term D_{reg} is calculated as the mean of the pairwise distances of each weight that are the same size. The formula is given by equation 5.11.

$$D_{\text{reg}} = \sum_{l} \frac{\lambda_d}{1 + \frac{1}{\mu^2} \sum_{\mu,\breve{\mu}} |W_\mu| - |W_{\breve{\mu}|}} \tag{5.11}$$

The regularization terms on the visible biases (g_i) is the same form as Tubiana's RBM [97] which is given by equation 5.12.

$$F_{\text{reg}} = \frac{\lambda_f}{2} \sum_{i,v} g_i(v)^2 \tag{5.12}$$

Optionally two other regularization terms can be used. One attempts to minimize the effect of gaps in the weights by summing the absolute value of the gap contributions (equation 5.13). The other promotes an even distribution between the positive and negative contributions of each weight as seen in equation 5.14.

$$G_{\text{reg}} = \lambda_g \sum_{\mu,ic} |W_{\mu,ic,gap}| \tag{5.13}$$

$$B_{\text{reg}} = \lambda_b \sum_{\mu} | \frac{\sum_{ic,kx,ky} max(W_{\mu,ic,kx,ky}, 0)}{\sum_{ic,kx,ky} |W_{\mu,ic,kx,ky}|} - \frac{\sum_{ic,kx,ky} |min(W_{\mu,ic,kx,ky}, 0)|}{\sum_{ic,kx,ky} |W_{\mu,ic,kx,ky}|} | \tag{5.14}$$

In total our cost function can be expressed as:

$$C(v) = - < logP(v) >_{\text{MSA}} - W_{\text{reg}} - F_{\text{reg}} - D_{\text{reg}} - G_{\text{reg}} - B_{\text{reg}} \tag{5.15}$$

5.2.2.3 Classification PCRBM

As an extension to the model, we have also implemented the Hybrid Discriminative RBM model [246], where the CRBM takes binary inputs and produces a classification label through a linearly connected tertiary layer. An illustration of the model's topology can be seen in Figure 30c. A weight matrix (symbol) connecting the hidden units to the label layer and biases for each class are introduced.

Accordingly the energy function is modified to include the new layer and its parameters:

$$E(v, h, y) = - \sum_i g_i(v_i) + \sum_{\mu} \mathcal{U}_{\mu}(h_{\mu}) - \sum_{\mu} h_{\mu} \left[W_{\mu} * v \right] - dy^T - hMy^T \tag{5.16}$$

The conditional probability distributions must be modified as well yielding:

$$I_i^v(h) = \left(\sum_{\mu} W_{\mu} \left[[h_{\mu}] \right] \right)_i \tag{5.17}$$

$$P(v_i|h) \propto \exp\left(g_i(v_i) + v_i I_i^v(h)\right) \tag{5.18}$$

$$I_{\mu}^{v,y}(v, y) = [W_{\mu} * v] + M_{\mu} y^T \tag{5.19}$$

$$P(h_{\mu}|v, y) \propto \exp\left(-\mathcal{U}_{\mu}(h_{\mu}) + h_{\mu} I_{\mu}^{v,y}(v, y)\right) \tag{5.20}$$

$$P(y|h) = \frac{\exp\left(d_y + hM_y\right)}{\sum_y \exp\left(d_y + hM_y\right)} \tag{5.21}$$

$$P(y|v) = \frac{\exp\left(d_y + \sum_{\mu} \Gamma_{\mu}(I_{\mu}^{v,y})\right)}{\sum_y \exp\left(d_y + \sum_{\mu} \Gamma_{\mu}(I_{\mu}^{v,y})\right)} \tag{5.22}$$

Our free energy becomes the joint conditional of the visible and label layers. The free energy can be decomposed into discriminative and generative components as shown below and discussed in [246].

$$F(v,y) = -\log\left(P(v,y)\right) = -\log\left(P(y|v)\right) - \log\left(P(v)\right) \tag{5.23}$$

We train the model using the hybrid learning objective defined in [246] making our model a Hybrid Discriminative Pooling Convolutional Restricted Boltzmann Machine (HDPCRBM).

MNIST Performance

As proof of concept we evaluated our Hybrid Discriminative PCRBM on the MNIST Dataset with 60000 training images and 10000 test images. Without a thorough hyperparameter search and unenhanced data, we were able to achieve a 98.93% accuracy on the validation set and 99.28% accuracy on the training dataset using just 200 hidden units. Our test set error (1.07%) is better than the binary Hybrid Discriminative RBM model with 1500 hidden units (1.28%) and the Discriminative Infinite RBM with 621 hidden units (1.41%) [254].

The performance is comparable to many other DNN methods. CNN and FCN architectures containing 343,0730 and 74,362 parameters, respectively, achieved better performance at 0.72% and 0.55% in [255]. Comparatively our model had 200 weights of shape 15x15, a 28x28 visible bias layer, a 10 parameter label bias, a 200x10 hidden to label weight matrix, and 200 four parameter activation functions yielding a total of 49,394 parameters.

5.3 Conclusion

In this work, I presented computational approaches to outstanding problems in the field of biopolymer nanotechnology. First, was the development and application of a coarse-grained model for the characterization of hybrid nucleic acid-protein structures. Not mentioned was a current application of the model, where we are assessing the ability of certain hybrid nucleic acid-protein structures to bind a therapeutic target. Future research includes the network model for DNA origami which was reviewed in the beginning of this chapter. I hope to see it's use in designing DNA crystals and other large DNA assemblies. Then we discussed aptamer design using generative and interpretable machine learning models. Analysis yielded insight into the binding properties of Thrombin aptamers and the generation of new Thrombin aptamers. Building off the model developed by Tubiana *et al.* [98], I showed a new model capable of being used on unaligned and different length data. Application of the model on SARS-CoV-2 DNA aptamers, cancer exosomes, and phage display data are ongoing projects.

Beyond my personal contributions to the field, I am excited by all of the research efforts going into this field. The field of biopolymer nanotechnology is incredibly vast and varied. Predicting even the next five years of developments is a fool's errand, but I hope to see an increased focus on applications centered around renewable energy, carbon capture, and overthrowing the corporate oligarchy.

REFERENCES

[1] William Charlton et al. *Aristotle's physics: Books I and II*. Oxford University Press, 1983.

[2] Clive Ruggles, B Cunliffe, and C Renfrew. "Astronomy and Stonehenge". In: *Proceedings-British Academy*. Vol. 92. Oxford University Press. 1997, pp. 203–230.

[3] Paolo Mazzarello. "A unifying concept: the history of cell theory". In: *Nature Cell Biology* 1.1 (1999), E13–E15.

[4] William A Fedak and Jeffrey J Prentis. "The 1925 Born and Jordan paper "On quantum mechanics"". In: *American Journal of Physics* 77.2 (2009), pp. 128–139.

[5] Werner Heisenberg. "Über quantentheoretische Umdeutung kinematischer und mechanischer Beziehungen". In: *Original Scientific Papers Wissenschaftliche Originalarbeiten*. Springer, 1985, pp. 382–396.

[6] Radostin Danev, Haruaki Yanagisawa, and Masahide Kikkawa. "Cryo-electron microscopy methodology: current aspects and future directions". In: *Trends in Biochemical Sciences* 44.10 (2019), pp. 837–848.

[7] Daniel J Nieves, Katharina Gaus, and Matthew AB Baker. "DNA-based super-resolution microscopy: DNA-PAINT". In: *Genes* 9.12 (2018), p. 621.

[8] Franz J Giessibl. "Advances in atomic force microscopy". In: *Reviews of Modern Physics* 75.3 (2003), p. 949.

[9] Lothar Schermelleh et al. "Super-resolution microscopy demystified". In: *Nature Cell Biology* 21.1 (2019), pp. 72–84.

[10] Mark Winey et al. "Conventional transmission electron microscopy". In: *Molecular Biology of the Cell* 25.3 (2014), pp. 319–323.

[11] KD Vernon-Parry. "Scanning electron microscopy: an introduction". In: *III-Vs Review* 13.4 (2000), pp. 40–44.

[12] Yunfei Hu et al. "NMR-based methods for protein analysis". In: *Analytical Chemistry* 93.4 (2021), pp. 1866–1879.

[13] Ryoichi Arai et al. "Conformations of variably linked chimeric proteins evaluated by synchrotron X-ray small-angle scattering". In: *PROTEINS: Structure, Function, and Bioinformatics* 57.4 (2004), pp. 829–838.

[14] WHt Zachariasen. "A general theory of X-ray diffraction in crystals". In: *Acta Crystallographica* 23.4 (1967), pp. 558–564.

[15] Mizuho Fushitani. “Applications of pump-probe spectroscopy”. In: *Annual Reports Section ”C” (Physical Chemistry)* 104 (2008), pp. 272–297.

[16] Claudio Pellegrini and Joachim Stöhr. “X-ray free-electron lasers—principles, properties and applications”. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 500.1-3 (2003), pp. 33–40.

[17] Shangguo Hou, Jack Exell, and Kevin Welsher. “Real-time 3D single molecule tracking”. In: *Nature Communications* 11.1 (2020), pp. 1–10.

[18] Anton Schmitz et al. “A SARS-CoV-2 spike binding DNA Aptamer that inhibits Pseudovirus infection by an RBD-independent mechanism”. In: *Angewandte Chemie International Edition* 60.18 (2021), pp. 10279–10285.

[19] Priyabrata Pattnaik. “Surface plasmon resonance”. In: *Applied Biochemistry and Biotechnology* 126.2 (2005), pp. 79–92.

[20] Christopher Maffeo, Jejoong Yoo, and Aleksei Aksimentiev. “De novo reconstruction of DNA origami structures through atomistic molecular dynamics simulation”. In: *Nucleic Acids Research* 44.7 (2016), pp. 3013–3019.

[21] Bastiaan C Buddingh’ and Jan C M Van Hest. “Artificial Cells: Synthetic Compartments with Life-like Functionality and Adaptivity”. In: (2017).

[22] Frederick C Neidhardt, John L Ingraham, Moselio Schaechter, et al. *Physiology of the bacterial cell*. Sinauer associates, 1990.

[23] Teruyuki Nagamune. “Biomolecular engineering for nanobio/bionanotechnology”. In: *Nano Convergence* 4.1 (2017), pp. 1–56.

[24] Akira Hiyoshi et al. “Does a DNA-less cellular organism exist on Earth?” In: *Genes to Cells* 16.12 (2011), pp. 1146–1158.

[25] Chris P. Ponting and Ross C. Hardison. “What fraction of the human genome is functional?” In: *Genome Research* 21.11 (2011), pp. 1769–1776.

[26] Anandakumar Shanmugam, Arumugam Nagarajan, and Shanmughavel Pramanayagam. “Non-coding DNA – a brief review”. In: *Journal of Applied Biology Biotechnology* 5.05 (2017), pp. 42–47.

[27] Eric J Strobel et al. “RNA systems biology: uniting functional discoveries and structural tools to understand global roles of RNAs”. In: *Current Opinion in Biotechnology* 39 (2016), pp. 182–191.

[28] John L Rinn and Howard Y Chang. “Genome regulation by long noncoding RNAs”. In: *Annual Review of Biochemistry* 81 (2012).

[29] Nadrian C. Seeman. “Nucleic acid junctions and lattices”. In: *Journal of Theoretical Biology* 99.2 (1982), pp. 237–247.

[30] Nadrian C. Seeman. “An Overview of Structural DNA Nanotechnology”. In: *Molecular Biotechnology* 37.3 (2007), pp. 246–257.

[31] Paul W.K. Rothemund. “Folding DNA to create nanoscale shapes and patterns”. In: *Nature* 440.7082 (2006), pp. 297–302.

[32] Lulu Qian and Erik Winfree. “Scaling up digital circuit computation with DNA strand displacement cascades”. In: *Science* 332.6034 (2011), pp. 1196–1201.

[33] Iuliia Zarubiieva et al. “Automated Leak Analysis of Nucleic Acid Circuits”. In: *ACS Synthetic Biology* (2022).

[34] Somnath Tagore et al. “DNA computation: application and perspectives”. In: *J. Proteomics Bioinform* 3.07 (2010).

[35] Suping Li and Yuliang Jiang. “A DNA nanorobot functions as a cancer therapeutic in response to a molecular trigger in vivo”. In: *Nature Biotechnology* 36.3 (2018), pp. 258–264.

[36] Sanchita Bhadra et al. “High-surety isothermal amplification and detection of SARS-CoV-2”. In: *MSphere* 6.3 (2021), e00911–20.

[37] Douglas Carmean et al. “DNA data storage and hybrid molecular–electronic computing”. In: *Proceedings of the IEEE* 107.1 (2018), pp. 63–72.

[38] A.-K. Pumm et al. “A DNA origami rotary ratchet motor”. In: *Nature* 607.7919 (2022).

[39] Peter Yakovchuk, Ekaterina Protozanova, and Maxim D. Frank-Kamenetskii. “Base-stacking and base-pairing contributions into thermal stability of the DNA double helix”. In: *Nucleic Acids Research* 34.2 (2006), pp. 564–574.

[40] Daniela Rhodes and Hans J Lipps. “G-quadruplexes and their regulatory roles in biology”. In: *Nucleic Acids Research* 43.18 (2015), pp. 8627–8637.

[41] Anthony K. Henras et al. “An overview of pre-ribosomal RNA processing in eukaryotes”. In: *Wiley Interdisciplinary Reviews: RNA* 6.2 (2015), pp. 225–242.

[42] Fabrizio Pucci and Alexander Schug. “Shedding light on the dark matter of the biomolecular structural universe: Progress in RNA 3D structure prediction”. In: *Methods* 162 (2019), pp. 68–73.

[43] Eric J Strobel, Angela M Yu, and Julius B Lucks. “High-throughput determination of RNA structures”. In: *Nature Reviews Genetics* 19.10 (2018), pp. 615–634.

[44] Peixuan Guo. "The emerging field of RNA nanotechnology". In: *Nature Nanotechnology* 5.12 (2010), pp. 833–842.

[45] Xiaohua Ni et al. "Nucleic acid aptamers: clinical applications and promising new horizons". In: *Current Medicinal Chemistry* 18.27 (2011), pp. 4206–4214.

[46] Peter K. Robinson. "Enzymes: principles and biotechnological applications". In: *Essays in Biochemistry* 59 (2015), pp. 1–41.

[47] H. Lee Sweeney and Erika L.F. Holzbaur. "Motor proteins". In: *Cold Spring Harbor Perspectives in Biology* 10.5 (2018).

[48] John H Golbeck. "Structure and function of photosystem I". In: *Annual Review of Plant Biology* 43.1 (1992), pp. 293–324.

[49] Nikolaus Pfanner, Bettina Warscheid, and Nils Wiedemann. "Mitochondrial proteins: from biogenesis to functional networks". In: *Nature Reviews Molecular Cell Biology* 20.5 (2019), pp. 267–284.

[50] Jie Zhu et al. "Protein assembly by design". In: *Chemical Reviews* 121.22 (2021), pp. 13701–13796.

[51] Priya Katyal, Michael Meleties, and Jin K Montclare. "Self-assembled protein-and peptide-based nanomaterials". In: *ACS Biomaterials Science & Engineering* 5.9 (2019), pp. 4132–4147.

[52] Yusuke Azuma, Thomas G.W. Edwardson, and Donald Hilvert. "Tailoring lumazine synthase assemblies for bionanotechnology". In: *Chemical Society Reviews* 47.10 (2018), pp. 3543–3557.

[53] Fabio Lapenta et al. "Coiled coil protein origami: From modular design principles towards biotechnological applications". In: *Chemical Society Reviews* 47.10 (2018), pp. 3530–3542.

[54] Gabriella Collu et al. "Chimeric single α-helical domains as rigid fusion protein connections for protein nanotechnology and structural biology". In: *Structure* 30.1 (2022), pp. 95–106.

[55] Brandan Dunham and Madhavi K. Ganapathiraju. "Benchmark evaluation of protein–protein interaction prediction algorithms". In: *Molecules* 27.1 (2022).

[56] Ivan Anishchenko et al. "De novo protein design by deep network hallucination". In: *Nature* 600.7889 (2021), pp. 547–552.

[57] Aaron Chevalier et al. "Massively parallel de novo protein design for targeted therapeutics". In: *Nature* 550.7674 (2017), pp. 74–79.

[58] Xingjie Pan and Tanja Kortemme. “Recent advances in de novo protein design: Principles, methods, and applications”. In: *Journal of Biological Chemistry* 296 (2021).

[59] Robert A Langan et al. “De novo design of bioactive protein switches”. In: *Nature* 572.7768 (2019), pp. 205–210.

[60] Jonathan Kyle Lassila et al. “Combinatorial methods for small-molecule placement in computational enzyme design”. In: *Proceedings of the National Academy of Sciences* 103.45 (2006), pp. 16710–16715.

[61] Aron Broom et al. “Ensemble-based enzyme design can recapitulate the effects of laboratory directed evolution in silico”. In: *Nature Communications* 11.1 (2020), pp. 1–10.

[62] Richard A Norman et al. “Computational approaches to therapeutic antibody design: established methods and emerging trends”. In: *Briefings in Bioinformatics* 21.5 (2020), pp. 1549–1567.

[63] Kathryn E Tiller and Peter M Tessier. “Advances in antibody design”. In: *Annual review of biomedical engineering* 17 (2015), p. 191.

[64] Nicholas Stephanopoulos. “Hybrid Nanostructures from the Self-Assembly of Proteins and DNA”. In: *Chem* 6.2 (2020), pp. 364–405.

[65] Yang Xu et al. “Tunable Nanoscale Cages from Self-Assembling DNA and Protein Building Blocks”. In: *ACS Nano* 13.3 (2019), pp. 3545–3554.

[66] Elisa de Llano et al. “Adenita: interactive 3D modelling and visualization of DNA nanostructures”. In: *Nucleic Acids Research* 1 (2020).

[67] Chao Min Huang et al. “Integrating computer-aided engineering and computer-aided design for DNA assemblies”. In: *bioRxiv* (2020).

[68] Shawn M. Douglas et al. “Rapid prototyping of 3D DNA-origami shapes with caDNAno”. In: *Nucleic Acids Research* 37.15 (2009), pp. 5001–5006.

[69] Sean Williams et al. “Tiamat: a three-dimensional editing tool for complex DNA structures”. In: *International Workshop on DNA-Based Computers*. Springer. 2008, pp. 90–101.

[70] David Doty, Benjamin L Lee, and Tristan Stérin. “scadnano: a browser-based, scriptable tool for designing DNA nanostructures”. In: *arXiv* (2020).

[71] Joakim Bohlin et al. “Design and simulation of DNA, RNA and hybrid protein–nucleic acid nanostructures with oxView”. In: *Nature Protocols* (2022), pp. 1–27.

[72] Nicolas Levy and Nicolas Schabanel. “ENSnano: a 3D modeling software for DNA nanostructures”. In: *DNA27-27th International Conference on DNA Computing and Molecular Programming*. 2021.

[73] Jacob B. Bale et al. “Accurate design of megadalton-scale two-component icosahedral protein complexes”. In: *Science* 353.6297 (2016), pp. 389–394.

[74] Longxing Cao et al. “De novo design of picomolar SARS-CoV-2 miniprotein inhibitors”. In: *Science* 370.6515 (2020).

[75] Erik Poppleton et al. “Nanobase. org: a repository for DNA and RNA nanostructures”. In: *Nucleic Acids Research* 50.D1 (2022), pp. D246–D252.

[76] Fan Hong et al. “Layered-crossover tiles with precisely tunable angles for 2D and 3D DNA crystal engineering”. In: *Journal of the American Chemical Society* 140.44 (2018), pp. 14670–14676.

[77] Thomas E. Ouldridge. “Coarse-grained modelling of dna and dna self-assembly”. PhD thesis. 2011, p. 1.

[78] Guangbao Yao et al. “Meta-DNA structures”. In: *Nature Chemistry* 12.11 (2020), pp. 1067–1075.

[79] Niranjan Srinivas et al. “On the biophysics and kinetics of toehold-mediated DNA strand displacement”. In: *Nucleic Acids Research* 41.22 (2013), pp. 10641–10658.

[80] Rahul Sharma et al. “Characterizing the Motion of Jointed DNA Nanostructures Using a Coarse-Grained Model”. In: *ACS Nano* 11.12 (2017), pp. 12426–12435.

[81] Benedict EK Snodin et al. “Direct simulation of the self-assembly of a small DNA origami”. In: *ACS nNno* 10.2 (2016), pp. 1724–1737.

[82] Thomas E Ouldridge et al. “DNA hybridization kinetics: zippering, internal displacement and sequence dependence”. In: *Nucleic Acids Research* 41.19 (2013), pp. 8886–8895.

[83] John Jumper et al. “Highly accurate protein structure prediction with AlphaFold”. In: *Nature* 596.7873 (2021), pp. 583–589.

[84] Kathryn Tunyasuvunakool et al. “Highly accurate protein structure prediction for the human proteome”. In: *Nature* 596.7873 (2021), pp. 590–596.

[85] Wujie Wang, Simon Axelrod, and Rafael Gómez-Bombarelli. “Differentiable Molecular Simulations for Control and Learning”. In: (2020). arXiv: 2003.00868.

[86] Justin S Smith, Olexandr Isayev, and Adrian E Roitberg. “ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost”. In: *Chemical Science* 8.4 (2017), pp. 3192–3203.

[87] Hao Wang and Weitao Yang. “Force field for water based on neural network”. In: *The Journal of Physical Chemistry Letters* 9.12 (2018), pp. 3232–3240.

[88] Pilsun Yoo et al. “Neural network reactive force field for C, H, N, and O systems”. In: *NPJ Computational Materials* 7.1 (2021), pp. 1–10.

[89] Oliver T Unke et al. “Machine learning force fields”. In: *Chemical Reviews* 121.16 (2021), pp. 10142–10186.

[90] Paul Smolensky. “Chapter 6: information processing in dynamical systems: foundations of harmony theory”. In: *Parallel Distributed Processing: Explorations in the Microstructure of Cognition* 1 ().

[91] Geoffrey E. Hinton. “Training Products of Experts by Minimizing Contrastive Divergence”. In: *Neural Computation* 14.8 (2002), pp. 1771–1800.

[92] Yann LeCun et al. “A tutorial on energy-based learning”. In: *Predicting Structured Data* 1.0 (2006).

[93] H-A Loeliger. “An introduction to factor graphs”. In: *IEEE Signal Processing Magazine* 21.1 (2004), pp. 28–41.

[94] Faruck Morcos et al. “Direct-coupling analysis of residue coevolution captures native contacts across many protein families”. In: *Proceedings of the National Academy of Sciences* 108.49 (2011), E1293–E1301.

[95] Martin Weigt et al. “Identification of direct residue contacts in protein–protein interaction by message passing”. In: *Proceedings of the National Academy of Sciences* 106.1 (2009), pp. 67–72.

[96] Jonathan C Chen et al. “Generating experimentally unrelated target molecule-binding highly functionalized nucleic-acid polymers using machine learning”. In: *Nature Communications* 13.1 (2022), pp. 1–17.

[97] Jérôme Tubiana, Simona Cocco, and Rémi Monasson. “Learning protein constitutive motifs from sequence data”. In: *eLife* 8 (2019), e39397.

[98] Jérôme Tubiana. “Restricted Boltzmann machines: from compositional representations to protein sequence analysis”. PhD thesis. Université Paris sciences et lettres, 2018.

[99] Barbara Bravi et al. "RBM-MHC: A Semi-Supervised Machine-Learning Method for Sample-Specific Prediction of Antigen Presentation by HLA-I Alleles". In: *Cell Systems* 12.2 (2021), pp. 195–202.

[100] Wenyan Liu et al. "Diamond family of nanoparticle superlattices". In: *Science* 351.6273 (2016), pp. 582–586.

[101] Chun Geng and Paul J. Paukstelis. "DNA crystals as vehicles for biocatalysis". In: *Journal of the American Chemical Society* 136.22 (2014), pp. 7817–7820.

[102] Suping Li et al. "A DNA nanorobot functions as a cancer therapeutic in response to a molecular trigger in vivo". In: *Nature biotechnology* 36.3 (2018), p. 258.

[103] Fei Zhang et al. "Structural DNA nanotechnology: State of the art and future perspective". In: *Journal of the American Chemical Society* 136.32 (2014), pp. 11198–11211.

[104] Rein V. Ulijn and Roman Jerala. "Peptide and protein nanotechnology into the 2020s: Beyond biology". In: *Chemical Society Reviews* 47.10 (2018), pp. 3391–3394.

[105] Neil P King et al. "Accurate design of co-assembling multi-component protein nanomaterials". In: *Nature* 510.7503 (2014), pp. 103–108.

[106] Mikael Madsen and Kurt V Gothelf. "Chemistries for DNA nanotechnology". In: *Chemical reviews* 119.10 (2019), pp. 6384–6458.

[107] Juan Jin et al. "Peptide assembly directed and quantified using megadalton DNA nanostructures". In: *ACS Nano* 13.9 (2019), pp. 9927–9935.

[108] Chao-Min Huang et al. "Integrated computer-aided engineering and design for DNA assemblies". In: *Nature Materials* 20.9 (2021), pp. 1264–1271.

[109] Daniel M. Hinckley et al. "An experimentally-informed coarse-grained 3-site-per-nucleotide model of DNA: Structure, thermodynamics, and dynamics of hybridization". In: *Journal of Chemical Physics* 139.14 (2013), p. 144903.

[110] Debayan Chakraborty, Naoto Hori, and D. Thirumalai. "Sequence-Dependent Three Interaction Site Model for Single- and Double-Stranded DNA". In: *Journal of Chemical Theory and Computation* 14.7 (2018), pp. 3763–3779.

[111] Natalia A. Denesyuk and D. Thirumalai. "Coarse-grained model for predicting RNA folding thermodynamics". In: *Journal of Physical Chemistry B* 117.17 (2013), pp. 4901–4911.

[112] Samuela Pasquali and Philippe Derreumaux. "HiRE-RNA: A high resolution coarse-grained energy model for RNA". In: *Journal of Physical Chemistry B* 114.37 (2010), pp. 11957–11966.

[113] Thomas E Ouldridge, Ard A Louis, and Jonathan PK Doye. “Structural, mechanical, and thermodynamic properties of a coarse-grained DNA model”. In: *The Journal of chemical physics* 134.8 (2011), 02B627.

[114] Benedict EK Snodin et al. “Introducing improved structural properties and salt dependence into a coarse-grained model of DNA”. In: *The Journal of chemical physics* 142.23 (2015), 06B613_1.

[115] Petr Šulc et al. “A nucleotide-level coarse-grained model of RNA”. In: *The Journal of chemical physics* 140.23 (2014), p. 235102.

[116] Petr Šulc et al. “Sequence-dependent thermodynamics of a coarse-grained DNA model”. In: *Journal of Chemical Physics* 137.13 (2012), p. 5101.

[117] Megan C. Engel et al. “Force-Induced Unravelling of DNA Origami”. In: *ACS Nano* 12.7 (2018), pp. 6734–6747.

[118] Antonio Suma et al. “TacoxDNA: A user-friendly web server for simulations of complex DNA structures, from single strands to origami”. In: *Journal of Computational Chemistry* 40.29 (2019), pp. 2586–2595.

[119] Jonathan P.K. Doye et al. “Coarse-graining DNA for simulations of DNA nanotechnology”. In: *Physical Chemistry Chemical Physics* 15.47 (2013), pp. 20395–20414.

[120] Michael Matthies et al. “Triangulated Wireframe Structures Assembled Using Single-Stranded DNA Tiles”. In: *ACS Nano* 13.2 (2019), pp. 1839–1848.

[121] Adam K. Sieradzan et al. “A new protein nucleic-acid coarse-grained force field based on the UNRES and NARES-2P force fields”. In: *Journal of Computational Chemistry* 39.28 (2018), pp. 2360–2370.

[122] Garima Mishra and Yaakov Levy. “Molecular determinants of the interactions between proteins and ssDNA Molecular determinants of the interactions between proteins and ssDNA”. In: *Proceedings of the National Academy of Sciences of the United States of America* 112.16 (2015), pp. 5033–5038.

[123] Cheng Tan, Tsuyoshi Terakawa, and Shoji Takada. “Dynamic Coupling among Protein Binding, Sliding, and DNA Bending Revealed by Molecular Dynamics”. In: *Journal of the American Chemical Society* 138.27 (2016), pp. 8512–8522.

[124] Cheng Tan and Shoji Takada. “Dynamic and Structural Modeling of the Specificity in Protein-DNA Interactions Guided by Binding Assay and Structure Data”. In: *Journal of Chemical Theory and Computation* 14.7 (2018), pp. 3877–3889.

[125] Bin Zhang et al. “Exploring the free energy landscape of nucleosomes”. In: *Journal of the American Chemical Society* 138.26 (2016), pp. 8126–8133.

[126] Rodrigo V Honorato, Jorge Roel-Touris, and Alexandre MJJ Bonvin. "MARTINI-based protein-DNA coarse-grained HADDOCKing". In: *Frontiers in molecular biosciences* 6 (2019), p. 102.

[127] Aram Davtyan et al. "AWSEM-MD: protein structure prediction using coarse-grained physical potentials and bioinformatically based local structure biasing". In: *The Journal of Physical Chemistry B* 116.29 (2012), pp. 8494–8503.

[128] Ali Rana Atilgan et al. "Anisotropy of fluctuation dynamics of proteins with an elastic network model". In: *Biophysical journal* 80.1 (2001), pp. 505–515.

[129] Sambit Kumar Mishra and Robert L. Jernigan. "Protein dynamic communities from elastic network models align closely to the communities defined by molecular dynamics". In: *PLoS ONE* 13.6 (2018).

[130] M. Gur, E. Zomot, and I. Bahar. "Global motions exhibited by proteins in micro- to milliseconds simulations concur with anisotropic network model predictions". In: *Journal of Chemical Physics* 139.12 (2013), p. 121912.

[131] Lei Yang et al. "Close Correspondence between the Motions from Principal Component Analysis of Multiple HIV-1 Protease Structures and Elastic Network Modes". In: *Structure* 16.2 (2008), pp. 321–330.

[132] Lorenzo Rovigatti et al. "A comparison between parallelization approaches in molecular dynamics simulations on GPUs". In: *Journal of computational chemistry* 36.1 (2015), pp. 1–8.

[133] Zhoutong Sun et al. "Utility of B-Factors in Protein Science: Interpreting Rigidity, Flexibility, and Internal Motion and Engineering Thermostability". In: *Chemical Reviews* (2019).

[134] Edvin Fuglebakk, Nathalie Reuter, and Konrad Hinsen. "Evaluation of protein elastic network models based on an analysis of collective motions". In: *Journal of Chemical Theory and Computation* 9.12 (2013), pp. 5618–5628.

[135] R Elber and M Karplus. "Low-frequency modes in proteins: Use of the effective-medium approximation to interpret the fractal dimension observed in electron-spin relaxation measurements". In: *Physical Review Letters* 56.4 (1986), pp. 394–397.

[136] Turkan Haliloglu, Ivet Bahar, and Burak Erman. "Gaussian dynamics of folded proteins". In: *Physical Review Letters* 79.16 (1997), pp. 3090–3093.

[137] Fei Xia, Dudu Tong, and Lanyuan Lu. "Robust heterogeneous anisotropic elastic network model precisely reproduces the experimental b-factors of biomolecules". In: *Journal of Chemical Theory and Computation* 9.8 (2013), pp. 3704–3714.

[138] Kelin Xia. “Multiscale virtual particle based elastic network model (MVP-ENM) for normal mode analysis of large-sized biomolecules”. In: *Physical Chemistry Chemical Physics* 20.1 (2017), pp. 658–669.

[139] Mingyang Lu, Billy Poon, and Jianpeng Ma. “A new method for coarse-grained elastic normal-mode analysis”. In: *Journal of Chemical Theory and Computation* 2.3 (2006), pp. 464–471.

[140] Min Yeh Tsai et al. “Electrostatics, structure prediction, and the energy landscapes for protein folding and binding”. In: *Protein Science* 25.1 (2016), pp. 255–269.

[141] Vinod K. Misra et al. “Electrostatic contributions to the binding free energy of the λcl repressor to DNA”. In: 75.5 (1998), pp. 2262–2273.

[142] Amir Marcovitz and Yaakov Levy. “Weak frustration regulates sliding and binding kinetics on rugged protein-DNA landscapes”. In: *Journal of Physical Chemistry B* 117.42 (2013), pp. 13005–13014.

[143] Alex Buchberger et al. “Hierarchical assembly of nucleic acid/coiled-coil peptide nanostructures”. In: *Journal of the American Chemical Society* 142.3 (2019), pp. 1406–1416.

[144] Leela S. Dodda et al. “LigParGen web server: An automatic OPLS-AA parameter generator for organic ligands”. In: *Nucleic Acids Research* 45.W1 (2017), W331–W336.

[145] Leela S. Dodda et al. “1.14CM1A-LBCC: Localized Bond-Charge Corrected CM1A Charges for Condensed-Phase Simulations”. In: *Journal of Physical Chemistry B* 121.15 (2017), pp. 3864–3870.

[146] William L. Jorgensen and Julian Tirado-Rives. “Potential energy functions for atomic-level simulations of water and organic and biomolecular systems”. In: 102.19 (2005), pp. 6665–6670.

[147] H. J.C. Berendsen, D. van der Spoel, and R. van Drunen. “GROMACS: A message-passing parallel molecular dynamics implementation”. In: *Computer Physics Communications* 91.1-3 (1995), pp. 43–56.

[148] Erik Poppleton et al. “Design, optimization and analysis of large DNA and RNA nanostructures through interactive visualization, editing and molecular simulation”. In: *Nucleic Acids Research* 48.12 (2020), e72.

[149] Andrej Šali and Tom L. Blundell. “Comparative protein modelling by satisfaction of spatial restraints”. In: *Journal of Molecular Biology* (1993).

[150] Stephen Albert Johnston et al. “A simple platform for the rapid development of antimicrobials”. In: *Scientific reports* 7.1 (2017), pp. 1–11.

[151] Jianyi Yang and Yang Zhang. "I-TASSER server: new development for protein structure and function predictions". In: *Nucleic acids research* 43.W1 (2015), W174–W181.

[152] Alexey Drozdetskiy et al. "JPred4: a protein secondary structure prediction server". In: *Nucleic acids research* 43.W1 (2015), W389–W394.

[153] Steve Plimpton. "Fast parallel algorithms for short-range molecular dynamics". In: *Journal of computational physics* 117.1 (1995), pp. 1–19.

[154] Yamuna Krishnan and Nadrian C Seeman. "Introduction: nucleic acid nanotechnology". In: *Chemical Reviews* 119.10 (2019), pp. 6271–6272.

[155] Fan Hong et al. "DNA origami: scaffolds for creating higher order structures". In: *Chemical Reviews* 117.20 (2017), pp. 12584–12640.

[156] James D Watson and Francis HC Crick. "Molecular structure of nucleic acids: a structure for deoxyribose nucleic acid". In: *Nature* 171.4356 (1953), pp. 737–738.

[157] Kaikai Chen et al. "Digital data storage using DNA nanostructures and solid-state nanopores". In: *Nano Letters* 19.2 (2018), pp. 1210–1215.

[158] Kaikai Chen et al. "Nanopore-based DNA hard drives for rewritable and secure data storage". In: *Nano Letters* 20.5 (2020), pp. 3754–3760.

[159] Tianqi Song et al. "Fast and compact DNA logic circuits based on single-stranded gates using strand-displacing polymerase". In: *Nature Nanotechnology* 14.11 (2019), pp. 1075–1081.

[160] Georg Seelig et al. "Enzyme-free nucleic acid logic circuits". In: *Science* 314.5805 (2006), pp. 1585–1588.

[161] Anupama J Thubagere et al. "Compiler-aided systematic construction of large-scale DNA strand displacement circuits using unpurified components". In: *Nature Communications* 8.1 (2017), pp. 1–12.

[162] Qiao Jiang et al. "DNA origami as a carrier for circumvention of drug resistance". In: *Journal of the American Chemical Society* 134.32 (2012), pp. 13396–13403.

[163] Po-Ssu Huang, Scott E Boyken, and David Baker. "The coming of age of de novo protein design". In: *Nature* 537.7620 (2016), pp. 320–327.

[164] Rhiju Das and David Baker. "Macromolecular modeling with rosetta". In: *Annual Review of Biochemistry* 77.1 (2008), pp. 363–382.

[165] Qinqin Hu et al. "DNA nanotechnology-enabled drug delivery systems". In: *Chemical Reviews* 119.10 (2018), pp. 6459–6506.

[166] Shawn M Douglas, Ido Bachelet, and George M Church. "A logic-gated nanorobot for targeted transport of molecular payloads". In: *Science* 335.6070 (2012), pp. 831–834.

[167] Wei Lu et al. "OpenAWSEM with Open3SPN2: A fast, flexible, and accessible framework for large-scale coarse-grained biomolecular simulations". In: *PLoS Computational Biology* 17.2 (2021), e1008308.

[168] Jonah Procyk, Erik Poppleton, and Petr Šulc. "Coarse-grained nucleic acid–protein model for hybrid nanotechnology". In: *Soft Matter* 17.13 (2021), pp. 3586–3593.

[169] Dmitry Lyumkis. "Challenges and opportunities in cryo-EM single-particle analysis". In: *Journal of Biological Chemistry* 294.13 (2019), pp. 5181–5197.

[170] Eric F Pettersen et al. "UCSF Chimera—a visualization system for exploratory research and analysis". In: *Journal of Computational Chemistry* 25.13 (2004), pp. 1605–1612.

[171] Alexander Wlodawer, Mi Li, and Zbigniew Dauter. "High-resolution cryo-EM maps and models: a crystallographer's perspective". In: *Structure* 25.10 (2017), pp. 1589–1597.

[172] Sunhwan Jo et al. "CHARMM-GUI: a web-based graphical user interface for CHARMM". In: *Journal of Computational Chemistry* 29.11 (2008), pp. 1859–1865.

[173] Jing Huang and Alexander D MacKerell Jr. "CHARMM36 all-atom additive protein force field: Validation based on comparison to NMR data". In: *Journal of Computational Chemistry* 34.25 (2013), pp. 2135–2145.

[174] Mark James Abraham et al. "GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers". In: *SoftwareX* 1 (2015), pp. 19–25.

[175] Günter Mayer et al. "From selection to caged aptamers: identification of light-dependent ssDNA aptamers targeting cytohesin". In: *Bioorganic & Medicinal Chemistry Letters* 19.23 (2009), pp. 6561–6564.

[176] Sabine Lennarz et al. "Selective Aptamer-Based Control of Intraneuronal Signaling". In: *Angewandte Chemie* 127.18 (2015), pp. 5459–5463.

[177] Anna Schüller et al. "Activation of the glmS ribozyme confers bacterial growth inhibition". In: *Chembiochem* 18.5 (2017), pp. 435–440.

[178] Malte Rosenthal, Franziska Pfeiffer, and Günter Mayer. "A Receptor-Guided Design Strategy for Ligand Identification". In: *Angewandte Chemie International Edition* 58.31 (2019), pp. 10752–10755.

[179] Alvaro Dario Ortega et al. “A synthetic RNA-based biosensor for fructose-1, 6-bisphosphate that reports glycolytic flux”. In: *Cell Chemical Biology* (2021).

[180] Christian Renzl, Ankana Kakoti, and Günter Mayer. “Aptamer-Mediated Reversible Transactivation of Gene Expression by Light”. In: *Angewandte Chemie* 132.50 (2020), pp. 22600–22604.

[181] Valeriy Domenyuk et al. “Poly-ligand profiling differentiates trastuzumab-treated breast cancer patients according to their outcomes”. In: *Nature Communications* 9.1 (2018), pp. 1–9.

[182] Laia Civit et al. “Systematic evaluation of cell-SELEX enriched aptamers binding to breast cancer cells”. In: *Biochimie* 145 (2018), pp. 53–62.

[183] Tassilo Hornung et al. “ADAPT identifies an ESCRT complex composition that discriminates VCaP from LNCaP prostate cancer cell exosomes”. In: *Nucleic Acids Research* 48.8 (2020), pp. 4013–4027.

[184] Jiehua Zhou and John J Rossi. “Cell-type-specific, aptamer-functionalized agents for targeted disease therapy”. In: *Molecular Therapy-Nucleic Acids* 3 (2014), e169.

[185] Craig Tuerk and Larry Gold. “Systematic evolution of ligands by exponential enrichment: RNA ligands to bacteriophage T4 DNA polymerase”. In: *Science* 249.4968 (1990), pp. 505–510.

[186] Andrew D Ellington and Jack W Szostak. “In vitro selection of RNA molecules that bind specific ligands”. In: *Nature* 346.6287 (1990), pp. 818–822.

[187] Mayte Sola et al. “Aptamers against live targets: is in vivo SELEX finally coming to the edge?” In: *Molecular Therapy-Nucleic Acids* 21 (2020), pp. 192–204.

[188] Daniela Proske et al. “Aptamers—basic research, drug development, and clinical applications”. In: *Applied Microbiology and Biotechnology* 69.4 (2005), pp. 367–374.

[189] Jan P Elskens, Joke M Elskens, and Annemieke Madder. “Chemical modification of aptamers for increased binding affinity in diagnostic applications: Current status and future prospects”. In: *International Journal of Molecular Sciences* 21.12 (2020), p. 4522.

[190] Sofia D’Souza, KV Prema, and Seetharaman Balaji. “Machine learning models for drug–target interactions: current knowledge and future directions”. In: *Drug Discovery Today* 25.4 (2020), pp. 748–756.

[191] Raphael JL Townshend et al. “Geometric deep learning of RNA structure”. In: *Science* 373.6558 (2021), pp. 1047–1051.

[192] Pauric Bannigan et al. “Machine learning directed drug formulation development”. In: *Advanced Drug Delivery Reviews* (2021).

[193] Jan Hoinka et al. “Identification of sequence–structure RNA binding motifs for SELEX-derived aptamers”. In: *Bioinformatics* 28.12 (2012), pp. i215–i223.

[194] Jia Song et al. “A sequential multidimensional analysis algorithm for aptamer identification based on structure analysis and machine learning”. In: *Analytical Chemistry* 92.4 (2019), pp. 3307–3314.

[195] Khalid K Alam, Jonathan L Chang, and Donald H Burke. “FASTAptamer: a bioinformatic toolkit for high-throughput sequence analysis of combinatorial selections”. In: *Molecular Therapy-Nucleic Acids* 4 (2015), e230.

[196] Timothy L Bailey et al. “The MEME suite”. In: *Nucleic Acids Research* 43.W1 (2015), W39–W49.

[197] Peng Jiang et al. “MPBind: a Meta-motif-based statistical framework and pipeline to Predict Binding potential of SELEX-derived aptamers”. In: *Bioinformatics* 30.18 (2014), pp. 2665–2667.

[198] Qingtong Zhou et al. “Searching the Sequence Space for Potent Aptamers Using SELEX in Silico”. In: *Journal of Chemical Theory and Computation* 11.12 (2015), pp. 5939–5946.

[199] Qingtong Zhou et al. “Exploring the Mutational Robustness of Nucleic Acids by Searching Genotype Neighborhoods in Sequence Space”. In: *The Journal of Physical Chemistry Letters* 8.2 (2017), pp. 407–414.

[200] Abe Pressman et al. “Analysis of in vitro evolution reveals the underlying distribution of catalytic activity among random sequences”. In: *Nucleic Acids Research* 45.14 (2017), pp. 8167–8179.

[201] Abe D. Pressman et al. “Mapping a Systematic Ribozyme Fitness Landscape Reveals a Frustrated Evolutionary Network for Self-Aminoacylating RNA”. In: *Journal of the American Chemical Society* 141.15 (2019), pp. 6213–6223.

[202] Peter K Koo et al. “Inferring sequence-structure preferences of RNA-binding proteins with convolutional residual networks”. In: *BioRxiv* (2018), p. 418459.

[203] Jan Zrimec et al. “Learning the regulatory code of gene expression”. In: *Frontiers in Molecular Biosciences* 8 (2021).

[204] Peter K Koo and Matt Ploenzke. “Deep learning for inferring transcription factor binding sites”. In: *Current Opinion in Systems Biology* 19 (2020), pp. 16–23.

[205] Drew H Bryant et al. “Deep diversification of an AAV capsid protein by machine learning”. In: *Nature Biotechnology* 39.6 (2021), pp. 691–696.

[206] Simona Cocco et al. “Inverse statistical physics of protein sequences: a key issues review”. In: *Reports on Progress in Physics* 81.3 (2018), p. 032601.

[207] Eleonora De Leonardis et al. “Direct-Coupling Analysis of nucleotide coevolution facilitates RNA secondary and tertiary structure prediction”. In: *Nucleic Acids Research* 43.21 (2015), pp. 10444–10455.

[208] William P Russ et al. “An evolution-based model for designing chorismate mutase enzymes”. In: *Science* 369.6502 (2020), pp. 440–445.

[209] Qin Zhou et al. “Global pairwise RNA interaction landscapes reveal core features of protein recognition”. In: *Nature Communications* 9.1 (2018), pp. 1–10.

[210] Yu Zhou et al. “DNA-Nanoscaffold-Assisted Selection of Femtomolar Bivalent Human alpha-Thrombin Aptamers with Potent Anticoagulant Activity”. In: *ChemBioChem* 20.19 (2019), pp. 2494–2503.

[211] Tijmen Tieleman. “Training Restricted Boltzmann Machines Using Approximations to the Likelihood Gradient”. In: *Proceedings of the 25th International Conference on Machine Learning*. ICML ’08. Helsinki, Finland: Association for Computing Machinery, 2008, pp. 1064–1071.

[212] Richard A. Neher and Boris I. Shraiman. “Statistical genetics and evolution of quantitative traits”. In: *Reviews of Modern Physics* 83.4 (2011), pp. 1283–1300.

[213] Daniel L Hartl, Daniel E Dykhuizen, and Antony M Dean. “Limits of Adaptation: The Evolution of Selective Neutrality”. In: *Genetics* 111.3 (1985), pp. 655–674.

[214] KAILLATHE Padmanabhan et al. “The structure of alpha-thrombin inhibited by a 15-mer single-stranded DNA aptamer.” In: *Journal of Biological Chemistry* 268.24 (1993), pp. 17651–17654.

[215] Franziska Pfeiffer et al. “Systematic evaluation of error rates and causes in short samples in next-generation sequencing”. In: *Scientific Reports* 8.1 (2018), pp. 1–14.

[216] Andreas Wagner. *Robustness and evolvability in living systems*. Princeton university press, 2013.

[217] Matteo Bisardi et al. “Modeling sequence-space exploration and emergence of epistatic signals in protein evolution”. In: *Molecular Biology and Evolution* 39.1 (2022), msab321.

[218] Tzu-Fang Lou et al. "Integrated analysis of RNA-binding protein complexes using in vitro selection and high-throughput sequencing and sequence specificity landscapes (SEQRS)". In: *Methods* 118 (2017), pp. 171–181.

[219] Christoph M Hammers and John R Stanley. "Antibody phage display: technique and applications". In: *The Journal of Investigative Dermatology* 134.2 (2014), e17.

[220] Titus Kretzschmar and Thomas Von Rüden. "Antibody discovery: phage display". In: *Current Opinion in Biotechnology* 13.6 (2002), pp. 598–602.

[221] Luca Sesta et al. "AMaLa: Analysis of Directed Evolution Experiments via Annealed Mutational Approximated Landscape". In: *International Journal of Molecular Sciences* 22.20 (2021), p. 10908.

[222] J. Tubiana and R. Monasson. "Emergence of Compositional Representations in Restricted Boltzmann Machines". In: *Phys. Rev. Lett.* 118 (13 2017), p. 138301.

[223] Clément Roussel, Simona Cocco, and Rémi Monasson. "Barriers and dynamical paths in alternating Gibbs sampling of restricted Boltzmann machines". In: *Physical Review E* 104.3 (2021), p. 034109.

[224] Mateusz Kogut, Cyprian Kleist, and Jacek Czub. "Why do G-quadruplexes dimerize through the 5'-ends? Driving forces for G4 DNA dimerization examined in atomic detail". In: *PLoS Computational Biology* 15.9 (2019), e1007383.

[225] Paul J Paukstelis and Nadrian C Seeman. "3D DNA crystals and nanotechnology". In: *Crystals* 6.8 (2016), p. 97.

[226] Jason S Kahn and Oleg Gang. "Designer Nanomaterials through Programmable Assembly". In: *Angewandte Chemie* 134.3 (2022), e202105678.

[227] Ye Tian et al. "Ordered three-dimensional nanomaterials using DNA-prescribed and valence-controlled material voxels". In: *Nature Materials* 19.7 (2020), pp. 789–796.

[228] Tao Zhang et al. "3D DNA origami crystals". In: *Advanced Materials* 30.28 (2018), p. 1800273.

[229] TT Ngo et al. "Tetrastack: Colloidal diamond-inspired structure with omnidirectional photonic band gap for low refractive index contrast". In: *Applied Physics Letters* 88.24 (2006), p. 241920.

[230] Mohiuddin Ahmed, Raihan Seraj, and Syed Mohammed Shamsul Islam. "The k-means algorithm: A comprehensive survey and performance evaluation". In: *Electronics* 9.8 (2020), p. 1295.

[231] Monica Colombo et al. “Peptide aptamers: The versatile role of specific protein function inhibitors in plant biotechnology”. In: *Journal of Integrative Plant Biology* 57.11 (2015), pp. 892–901.

[232] Mohamed A Alfaleh et al. “Phage display derived monoclonal antibodies: from bench to bedside”. In: *Frontiers in Immunology* 11 (2020), p. 1986.

[233] Briana I Martinez et al. “Uncovering temporospatial sensitive TBI targeting strategies via in vivo phage display”. In: *Science Advances* 8.29 (2022), eabo5047.

[234] Line Ledsgaard et al. “Basics of antibody phage display technology”. In: *Toxins* 10.6 (2018), p. 236.

[235] Di Sun et al. “Computational tools for aptamer identification and optimization”. In: *TrAC Trends in Analytical Chemistry* (2022), p. 116767.

[236] Mayumi Takahashi et al. “High throughput sequencing analysis of RNA libraries reveals the influences of initial library and PCR methods on SELEX efficiency”. In: *Scientific Reports* 6.1 (2016), pp. 1–14.

[237] Jan Hoinka et al. “Aptacluster–a method to cluster ht-selex aptamer pools and lessons from its application”. In: *International Conference on Research in Computational Molecular Biology*. Springer. 2014, pp. 115–128.

[238] Neda Emami and Reza Ferdousi. “AptaNet as a deep learning approach for aptamer–protein interaction prediction”. In: *Scientific Reports* 11.1 (2021), pp. 1–19.

[239] David Laehnemann, Arndt Borkhardt, and Alice Carolyn McHardy. “Denoising DNA deep sequencing data—high-throughput sequencing errors and their correction”. In: *Briefings in bioinformatics* 17.1 (2016), pp. 154–179.

[240] Graham W Taylor, Geoffrey E Hinton, and Sam Roweis. “Modeling human motion using binary latent variables”. In: *Advances in Neural Information Processing Systems* 19 (2006).

[241] Ruslan Salakhutdinov, Andriy Mnih, and Geoffrey Hinton. “Restricted Boltzmann machines for collaborative filtering”. In: *Proceedings of the 24th international conference on Machine learning*. 2007, pp. 791–798.

[242] Erkan Karakus and Hatice Kose. “Conditional restricted Boltzmann machine as a generative model for body-worn sensor signals”. In: *IET Signal Processing* 14.10 (2020), pp. 725–736.

[243] Mohammad Norouzi. “Convolutional restricted Boltzmann machines for feature learning”. PhD thesis. School of Computing Science-Simon Fraser University, 2009.

[244] Lukun Wang. “Three-dimensional convolutional restricted Boltzmann machine for human behavior recognition from RGB-D video”. In: *EURASIP Journal on Image and Video Processing* 2018.1 (2018), pp. 1–11.

[245] Hardik B Sailor and Hemant A Patil. “Novel unsupervised auditory filterbank learning using convolutional RBM for speech recognition”. In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 24.12 (2016), pp. 2341–2353.

[246] Hugo Larochelle and Yoshua Bengio. “Classification using discriminative restricted Boltzmann machines”. In: *Proceedings of the 25th international conference on Machine learning*. 2008, pp. 536–543.

[247] Gijs Van Tulder and Marleen De Bruijne. “Combining generative and discriminative representation learning for lung CT analysis with convolutional restricted Boltzmann machines”. In: *IEEE transactions on medical imaging* 35.5 (2016), pp. 1262–1272.

[248] Tu Dinh Nguyen et al. “Supervised restricted boltzmann machines”. In: *Uncertainty in Artificial Intelligence - Proceedings of the 33rd Conference, UAI 2017* (2017).

[249] Vinod Nair and Geoffrey E Hinton. “Rectified linear units improve restricted boltzmann machines”. In: *Icml*. 2010.

[250] Geoffrey E. Hinton. “A practical guide to training restricted boltzmann machines”. In: *Lecture Notes in Computer Science* 7700 LECTURE NO (2012), pp. 599–619. DOI: 10.1007/978-3-642-35289-8_32.

[251] Guillaume Desjardins et al. “Parallel tempering for training of restricted Boltzmann machines”. In: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. MIT Press Cambridge, MA. 2010, pp. 145–152.

[252] Rémi Souriau et al. “A review on generative Boltzmann networks applied to dynamic systems”. In: *Mechanical Systems and Signal Processing* 147 (2021), p. 107072.

[253] A Di Gioacchino et al. “Generative and interpretable machine learning for aptamer design and analysis of in vitro sequence selection”. In: (2022).

[254] Xuan Peng, Xunzhang Gao, and Xiang Li. “On better training the infinite restricted Boltzmann machines”. In: *Machine Learning* 107.6 (2018), pp. 943–968.

[255] Xuefeng Jiang et al. “Capsnet, cnn, fcn: Comparative performance evaluation for image classification”. In: *International Journal of Machine Learning and Computing* 9.6 (2019), pp. 840–848.

[256] David N Mastronarde. “Automated electron microscope tomography using robust prediction of specimen movements”. In: *Journal of Structural Biology* 152.1 (2005), pp. 36–51.

[257] Sjors HW Scheres. “RELION: implementation of a Bayesian approach to cryo-EM structure determination”. In: *Journal of Structural Biology* 180.3 (2012), pp. 519–530.

[258] Jasenko Zivanov et al. “New tools for automated high-resolution cryo-EM structure determination in RELION-3”. In: *eLife* 7 (2018), e42166.

[259] James R Kremer, David N Mastronarde, and J Richard McIntosh. “Computer visualization of three-dimensional image data using IMOD”. In: *Journal of Structural Biology* 116.1 (1996), pp. 71–76.

[260] Shawn Q Zheng et al. “MotionCor2: anisotropic correction of beam-induced motion for improved cryo-electron microscopy”. In: *Nature Methods* 14.4 (2017), pp. 331–332.

[261] Alexis Rohou and Nikolaus Grigorieff. “CTFFIND4: Fast and accurate defocus estimation from electron micrographs”. In: *Journal of Structural Biology* 192.2 (2015), pp. 216–221.

[262] Ali Punjani et al. “cryoSPARC: algorithms for rapid unsupervised cryo-EM structure determination”. In: *Nature Methods* 14.3 (2017), pp. 290–296.

[263] Sjors HW Scheres and Shaoxia Chen. “Prevention of overfitting in cryo-EM structure determination”. In: *Nature Methods* 9.9 (2012), pp. 853–854.

[264] Joseph N. Zadeh et al. “NUPACK: Analysis and design of nucleic acid systems”. In: *Journal of Computational Chemistry* 32.1 (2011), pp. 170–173.

[265] Diederik P. Kingma and M. Welling. “Auto-Encoding Variational Bayes”. In: *CoRR* abs/1312.6114 (2014).

[266] Kaiming He et al. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 2016-December (2016), pp. 770–778. arXiv: 1512.03385.

[267] Evie van der Spoel et al. “Siamese Neural Networks for One-Shot Image Recognition”. In: *ICML - Deep Learning Workshop* 7.11 (2015), pp. 956–963. arXiv: arXiv:1011.1669v3.

[268] Rafael Müller, Simon Kornblith, and Geoffrey Hinton. “When does label smoothing help?” In: *Advances in Neural Information Processing Systems* 32 (2019). arXiv: 1906.02629.

[269] Takeru Miyato et al. “Spectral normalization for generative adversarial networks”. In: *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings* (2018). arXiv: 1802.05957.

[270] Han Zhang et al. “Self-attention generative adversarial networks”. In: *36th International Conference on Machine Learning, ICML 2019* 2019-June (2019), pp. 12744–12753. arXiv: 1805.08318.

[271] Magnus Ekeberg, Tuomo Hartonen, and Erik Aurell. “Fast pseudolikelihood maximization for direct-coupling analysis of protein structure from many homologous amino-acid sequences”. In: *Journal of Computational Physics* 276 (2014), pp. 341–356. arXiv: 1401.4832.

[272] J. P. Barton et al. “ACE: adaptive cluster expansion for maximum entropy graphical model inference”. In: *Bioinformatics* 32.20 (2016), pp. 3089–3097.

APPENDIX A

SUPPLEMENTARY INFORMATION FOR CHAPTER 3

A.1 Materials and supplies

All DNA sequences were purchased from Integrated DNA technologies (IDT). The M13 scaffold strand was amplified and purified in-house. The aldolse protein was expressed in-house.

A.2 Synthesis and characterization of KDPG aldolase protein-DNA building block, tetrahedral origami, 4 turn tetrahedron.

A.2.1 Synthesis of KDPG aldolase protein-DNA building block (PDNA-bb).

The protein was expressed, purified and conjugated as previously reported[65].

A.2.2 Origami formation.

All origami solutions were made to 100 µL volumes with 20 nM of the M13 scaffold and 10 equivalents of staples (200 nM) in 1XTAE-18.5mM $MgCl_2$ buffer. Staples bearing handles were also added at 10x excess. The samples were heated and slowly cooled in a PCR machine using the 'Origami tetrahedron' annealing protocol described below.

A.2.3 Origami Tetrahedron annealing protocol.

Samples were held at 90 °C for 5 min, followed by a gradient from 86-71 °C at a rate of 1 °C/5 min, followed by a gradient from 70-40 °C at a rate of 1°C/15 min, followed by another gradient from 39-20 °C at the rate of 1°C/10 min, and then quickly cooled, and stored at, 10 °C.

A.2.4 Annealing protocol for PDNA incorporation in the tetrahedral origami frame.

Samples were heated to 45°C for 15 min, and then cooled slowly by a gradient from 40-4°C for over 12 hours. Purified PDNA was added in 40x excess (4 sites*10X excess) to the impure tetrahedron origami structures, following which the sample was gel purified as described below.

A.2.5 Characterization of Tetrahedral origami structures.

Samples were run on 1.2% Agarose gels made in 1xTAE with 20 mM $MgCl_2$ buffer, and pre-stained with ethidium bromide. The running buffer was 1xTAE with 12.5 mM $MgCl_2$. To 10 μL of the annealed sample from the thermocycler was added 1 μL of 10x loading dye. The gels were electrophoresed for 1.5 hours at a constant voltage of 90 V at 4 °C.

A.2.6 Purification of tetrahedron origami structures.

20 nM, 200 μL samples were run on a pre-stained 1.2% Agarose gel as before for 1.5-2 hours. After electrophoresis, the band of choice was excised, put into a freeze and squeeze tube and kept in -80 °C for 1 hour, then centrifuged in the cold room at low centrifuge speeds of 1600 rcf for 40 min and characterized by TEM for intactness.

A.2.7 ssDNA purification.

All ssDNA strands forming open tetrahedron DNA (1T, 2T, 3T, & 4T) were obtained from IDT and purified in house using 8 % denaturing polyacrylamide gel electrophoresis (PAGE). The running buffer was 1XTBE buffer. The desired bands were excised from gel and kept in eluting buffer for overnight at room temperature followed by desalting using a 3 kDa Amicon filter. All purified DNA strands were stored at -20 °C for further use. The variation in T nucleotide base sequence in different open tetrahedral has been highlighted in red letters in Figure 36B.

A.2.8 Open DNA 4 Turn-tetrahedron formation.

All open DNA tetrahedron (1T, 2T, 3T & 4T) solutions were made in 60 μL volume with 1 μM concentration of each component strands of tetrahedron in equimolar ratio in 1XTAE-12.5 mM $MgCl_2$ buffer (shown in green, yellow, ash and green in Figure 36). The solutions were annealed in a PCR machine using 'Open DNA 4 turn-tetrahedron annealing protocol' described below.

A.2.9 Open DNA 4 turn-tetrahedron annealing protocol.

Samples were heated at 90 °C for 5 min, followed by a gradient of 88 – 76 °C at a rate of 1°C/5 min, followed by a gradient of 76 – 24 °C at a rate of 1°C/2.5 min and then quickly cooled to 4 °C.

A.2.10 Characterization and purification of open 4 turn-DNA tetrahedron.

Samples were run on 5 % Native PAGE gel made in 1XTAE-12.5 mM $MgCl_2$ buffer. The running buffer was 1XTAE-12.5 mM $MgCl_2$. The gel was electrophoresed for 2.5 h at constant 200 V keeping constant temperature at 10 °C, and post-stained with ethidium bromide. Thereafter band of choice was excised from gel and kept in 1XTAE-12.5 mM $MgCl_2$ buffer at room temperature for overnight. Purity of samples were confirmed via running the eluted samples on 5 % Native PAGE gel made in 1XTAE with 12.5 mM $MgCl_2$.

A.2.11 4 turn-DNA tetrahedron-protein cage formation.

All DNA tetrahedral-protein samples were made to 90 µL volumes with 300 ng open DNA tetrahedron (~21 nM) and 100 nM PDNA in 1X-TAE-12.5 mM $MgCl_2$ buffer. Solutions were annealed in a PCR machine using the '4 turn-DNA tetrahedral-protein cage annealing protocol' described below.

A.2.12 4 turn-DNA tetrahedral-protein cage annealing protocol.

Samples were heated at 56 °C for 2 min, followed by a gradient of 55 – 46 °C at a rate of 1 °C/2 min, followed by a gradient of 45 – 30 °C at a rate of 1 °C/15 min, followed by 29 – 26 °C at a rate of 1 °C/10 min, followed by incubation at 25 °C for 30 minutes and then quickly cooled at 4 °C.

A.2.13 Characterization and purification of 4 turn-DNA tetrahedral-protein cage.

Samples were run on 4 % Native PAGE gel made in 1XTAE-12.5 mM $MgCl_2$ buffer. The gel was electrophoresed for 2 h at constant 200 V at 4 °C. After that, gel was stained with ethidium bromide and the band of choice was excised and kept in 1XTAE-12.5 mM $MgCl_2$ buffer at 4 °C for overnight. The sample concentration was measured using a Nanodrop and further characterized using AFM.

A.3 Experimental protocols for TEM, Cryo-TEM and AFM

A.3.1 Transmission electron microscopy (TEM) characterization.

5 µL of sample was adsorbed on a formvar stabilized carbon type-B, 400 mesh copper grid (Ted Pella, part number 01814-F) that was glow-discharged for 1 minute. The sample was stained using 5 µL of a 2% (w/v) uranyl formate solution with 25 mM sodium hydroxide. The grids were allowed to sit for 5 minutes before applying the samples. Sample was then applied on the grid and incubated for 5 minutes. Grids were allowed to float on a drop of the required sample or stain before wicking excess liquid using a Whatman filter paper.

A.3.2 Plunging freezing for Cryo-EM imaging.

5 µL of sample was absorbed on the carbon side of the ultrathin carbon film on lacey carbon support film, 400 mesh copper grid (Ted Pella, part number 08124) that had been glow discharged for 1 minute. The grids were left to sit idle for 5 minutes before the samples applied onto it. Samples were incubated for 5 minutes. Thereafter, the grids were plunged using an in-house manual plunger after 5-6 seconds into liquid ethane and immediately transferred to grid boxes in liquid nitrogen. The grids were stored in these boxes until imaged in the microscope.

A.3.3 Atomic force microscopy (AFM) characterization.

30 µL of samples were deposited on freshly cleaved mica surface (Ted Pella) and 20 µL of 1XTAE-12.5 mM $MgCl_2$ filtered buffer was added to the samples. After incubating samples at room temperature for 10 min, 10 µL of filtered $NiCl_2$ (0.2 M) solution was added to the samples and kept it at room temperature for 2 min. About 60 µL of filtered 1XTAE-12.5 mM $MgCl_2$ buffer was added to the AFM tips. All the AFM imaging was done in the 'ScanAsyst mode in fluid' with ScanAsyst-Fluid+ tips (Bruker).

A.4 Processing of cryo-EM data

A.4.1 Data acquisition.

All cryo-EM data collections were completed in the Eyring Materials Center (EMC) at Arizona State University (ASU). The grid specimen was imaged using a Thermo Fisher/FEI Titan Krios transmission electron microscope (TEM) (Thermo Fisher/FEI, Hillsborough, OR) at an accelerating voltage of 300 keV. The electron scattering was recorded by a Gatan Summit K2 direct electron detector (DED) camera in super-resolution mode (Gatan, Pleasanton, CA). For the tetrahedron dataset, the nominal magnification was set to 30,487x, corresponding to a physical pixel size of 1.64 Å/pixel at the specimen level. The defocus was varied from -0.8 to -2.5 µm. The camera counted rate was calibrated to 3.24 e^-/pixel/second. The exposure time was 8 seconds, accumulating to a total dosage of 46.1 e^-/Å[256]. The procedure of low-dose imaging was automated using SerialEM software (version 3.8)[256] with customized macros.

For the PDNA-bound tetrahedron dataset, the nominal magnification was set to 37,879X, corresponding to a physical pixel size of 1.32 Å/pixel at the specimen level. The defocus was varied from -0.8 to -2.5 µm. The camera counted rate was calibrated to 4.33 e^-/pixel/second. The exposure time was 8 seconds, accumulating to a total dosage of 39.5 e^-/Å[256].

A.4.2 Image processing

Image processing was generally conducted using the Relion software (version 3.1-beta)[257, 258]. For the tetrahedron dataset, 3,448 cryo-EM movies were unpacked and gain normalized using IMOD software package (version 4.9)[259]. The specimen movements between frames were registered and averaged using MotionCor2 (version 1.2.1)[260], and the CTF (contrast transfer function) parameters of the frame average were estimated using CTFFIND4 (version 4.1.13)[261]. The frame averages were imported into Relion for subsequent processing. 25,949 particles were manually selected from the micrographs using a Gaussian blob with a diameter of 802 Å. Iterative reference-free two-dimensional (2D) classification was performed using Relion to remove false positives and incomplete views. 20,714 selected particle images were used to generate a three-dimensional (3D) initial model using Relion[257, 258, 262]. The cryo-EM density was then refined against the experimental particle images by imposing a tetrahedral symmetry. The final resolution was determined as 26.1 Å using a gold-standard FSC method at the cutoff of 0.143[263].

For the PDNA-bb-bound tetrahedron, 2,619 cryo-EM movies were unpacked and gain normalized using IMOD software package[259]. The specimen movements between frames were registered and averaged using MotionCor2[260]. The CTF parameters of the frame average were estimated using CTFFIND4[261]. The frame averages were imported into Relion for subsequent processing and 10,255 particles were selected from the micrographs. Iterative reference-free 2D classification was performed to remove any false positives and incomplete views. 7,676 particle images were selected to generate a 3D initial model using Relion[257, 258, 262]. The cryo-EM density was then refined against the experimental particle images by imposing a tetrahedral symmetry. The final resolution was determined as 27.6 Å using a gold-standard FSC method at the cutoff of 0.143[263].

A.5 Experimental details of the fluorophore assay

Samples for both the calibration curves, i.e. Cy5 labelled strand and the FAM-DNA1 aldolase protein, were prepared by making double stranded versions of each. This was first done by annealing the corresponding sample with its complementary strand in defined ratios (1X for the Cy5 strand and 3X excess for Protein-conjugate (since there are 3 DNA per protein)). These double stranded versions were then annealed and measured in a Nanolog fluorimeter (Horiba Jobin Yvon) using a quartz cuvette of 3-mm path length having a sample volume of 60 µL at 495 nm for FAM-DNA1 (KDPG aldolase protein) and at 647 nm for the Cy5 labeled strand.

The calibration curves were fit using the equation y =mx + c, where in m is the slope and c is the intercept, where the emission peak values were taken at 520 nm for the FAM sample and 664 nm for the Cy5 sample.

A.6 Anisotropic Network Model fitting and linker parameters

Two separate Anisotropic Network Models (ANMs) were used to simulate KDPG Aldolase at our two different temperatures (113K and 300K). As crystallographic data (B factors) for the KDPG aldolase fluctuations is available at 110K, our low temperature models were parameterized to best match this information. The fitting for our low temperature ANM is shown below in Figure 31 with the ANM parameters listed in the description.

High temperature fluctuation data is not readily available and can significantly differ from crystal data collected at low temperatures. To generate sufficient data, KDPG aldolase was simulated with Charmm36 Forcefield and tip3p water molecules for 10 ns from its crystal structure. The B factors and average coordinates were collected and used to parameterize our high temperature ANM model. The comparison of the B factors from fully atomistic simulation and the ANM B factors are shown below. The fitting for our high temperature ANM is shown below in Figure 32 with the ANM parameters listed in the description.

The linker used was parameterized previously[168]. The parameters in SI units are 10.97 Å for the equilibrium length and 0.031 pN/Å. The force constant was raised slightly from our previous publication to 0.0530 pN/Å in order to limit the maximum linker length possible.

A.7 Additional Simulation Data

All simulations were carried out using the oxDNA2 model with sequence dependent parameters. The below tables contain additional simulation data for the empty and connected cages of the cage design, broken down by the respective pair-wise interaction potentials in

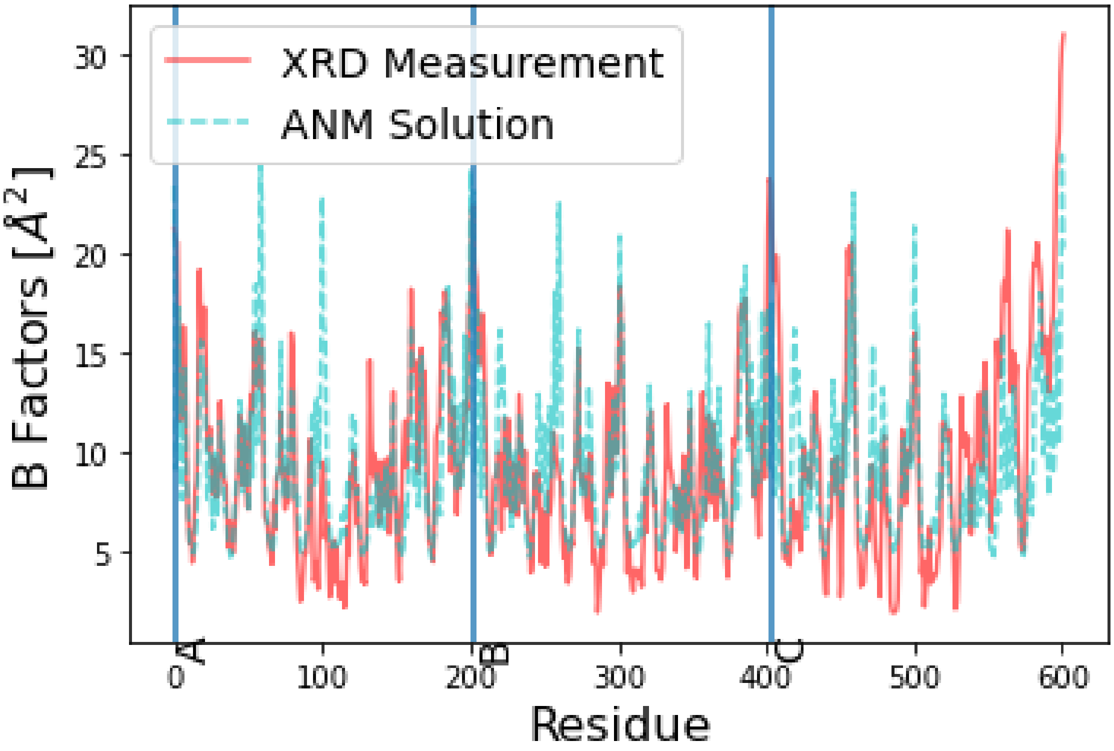


Figure 31. Comparison of the XRD measurement of B Factors from 1WA3.pdb and the predicted B Factors of the ANM at 113K with a cutoff of 13 Å and force constant of 15.039 pN/Å.

the coarse-grained model of DNA, oxDNA: FENE spring backbone-backbone-potential, excluded volume between nearest neighbor nucleotides (BEXC), stacking interaction (STCK), non-nearest neighbor excluded volume (NEXC) base pairing interaction (HB), cross-stacking interaction (CRSTCK), coaxial stacking interaction (CXSTCK), electrostatic repulsion modeled via Debye-Huckel potential.

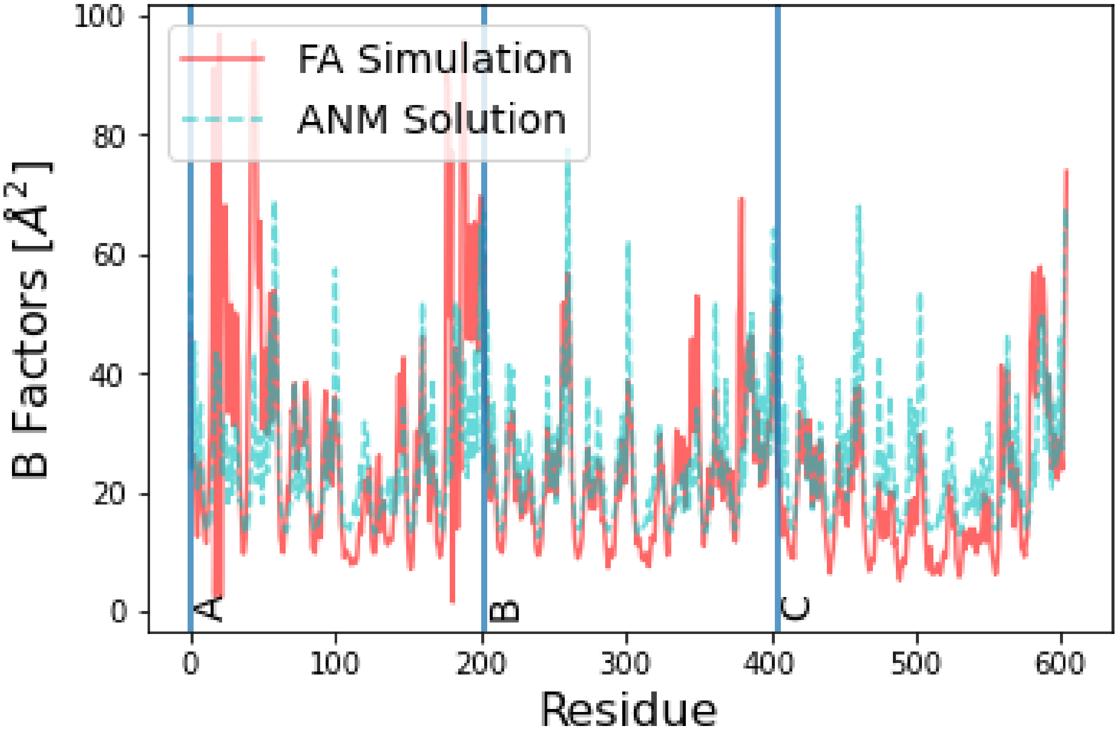


Figure 32. Comparison fully atomistic B Factors from our CHARMM simulation and the predicted B Factors of the ANM at 300K with a cutoff of 13 Å and force constant of 15.982 pN/ Å.

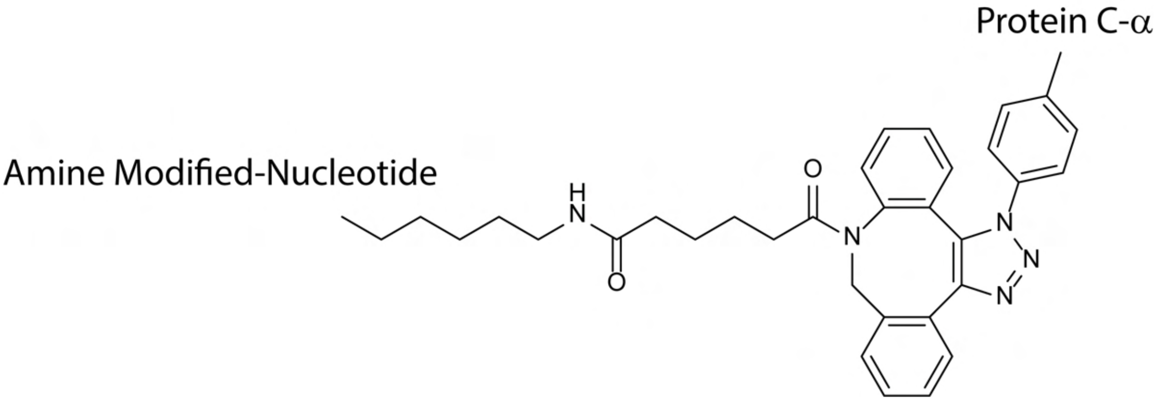


Figure 33. The chemical schematic of the DBCO-NHS ester linker used in this work. Note that this structure represents the linker after reaction with both the amine-modified nucleotide (thus the sulfo-NHS moiety has been displaced) and the azido-Phe on the protein surface (leading to the triazole linkage shown).

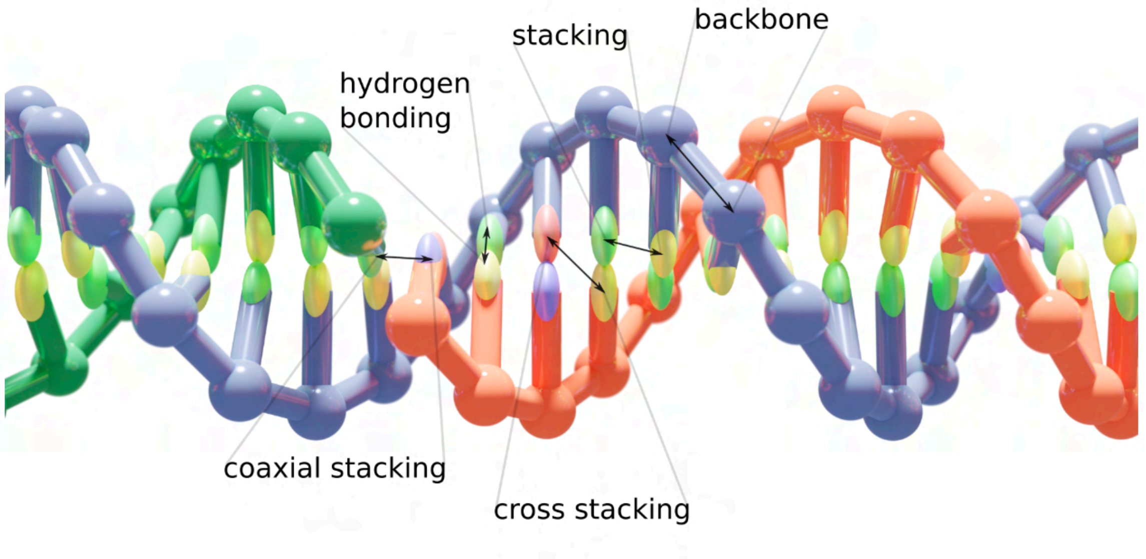


Table 5. Overview of the oxDNA model. Pairwise interactions including stacking, cross stacking, coaxial stacking, hydrogen bonding, and nearest neighbor backbone excluded volume is depicted on a dna duplex.

Contribution:	FENE	BEXC	STCK	NEXC	HB	CRSTCK	CXSTCK	DH	Total
1T Avg	3.896	0.003	-61.441	0.021	-0.043	-4.975	-1.46	0.252	-63.747
1T Std Dev	0.191	0.002	5.098	0.008	0.044	0.692	0.48	0.02	
2T Avg	3.801	0.01	-62.148	0.028	-0.023	-3.687	-0.582	0.242	-62.359
2T Std Dev	0.117	0.012	4.794	0.012	0.04	0.401	0.653	0.034	
3T Avg	3.878	0.01	-66.354	0.021	-0.01	-2.858	-0.134	0.2	-65.247
3T Std Dev	0.131	0.008	7.81	0.014	0.018	1.512	0.131	0.036	
4T Avg	3.813	0.01	-68.287	0.023	-0.003	-2.222	-0.076	0.204	-66.538
4T Std Dev	0.115	0.01	6.341	0.012	0.008	1.706	0.081	0.041	

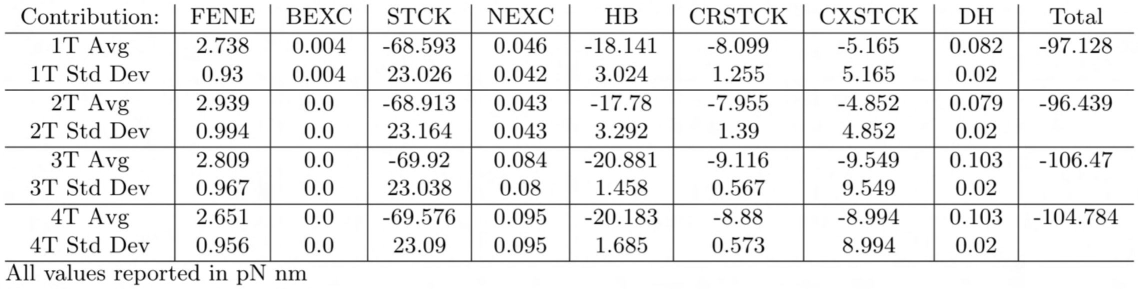
All values reported in pN nm

Table 6. Average energy broken down by oxDNA forcefield term for all T spacer nucleotides in models with all arms attached to KDPG aldolase.

Contribution:	FENE	BEXC	STCK	NEXC	HB	CRSTCK	CXSTCK	DH	Total
1T Avg	3.822	0.004	-70.904	0.038	-0.015	-6.393	-1.558	0.262	-74.744
1T Std Dev	0.078	0.003	20.535	0.022	0.019	1.319	1.42	0.071	
2T Avg	3.873	0.003	-72.886	0.017	-0.012	-4.654	-0.166	0.238	-73.587
2T Std Dev	0.089	0.002	17.673	0.008	0.025	0.877	0.183	0.052	
3T Avg	3.871	0.007	-74.238	0.017	-0.01	-3.711	-0.068	0.219	-73.913
3T Std Dev	0.078	0.004	11.806	0.006	0.018	1.538	0.049	0.034	
4T Avg	3.927	0.008	-73.405	0.015	-0.784	-2.99	-0.069	0.21	-73.088
4T Std Dev	0.082	0.005	8.835	0.009	3.706	2.097	0.045	0.033	

All values reported in pN nm

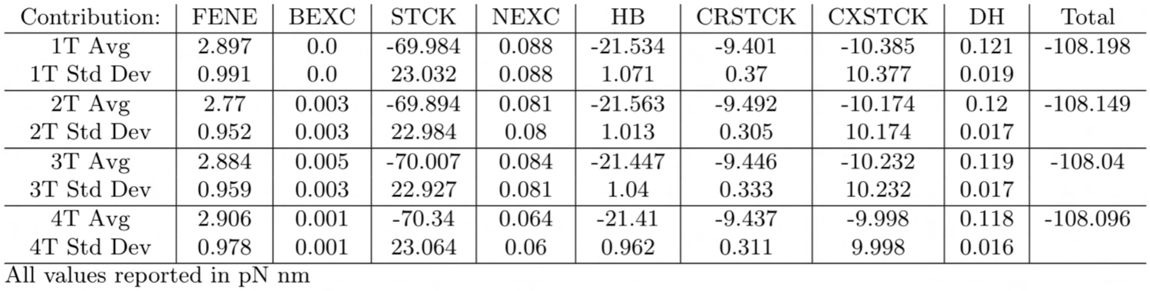
Table 7. Average energy broken down by oxDNA forcefield term for all T spacer nucleotides in empty DNA cage.



Contribution:	FENE	BEXC	STCK	NEXC	HB	CRSTCK	CXSTCK	DH	Total
1T Avg	2.738	0.004	-68.593	0.046	-18.141	-8.099	-5.165	0.082	-97.128
1T Std Dev	0.93	0.004	23.026	0.042	3.024	1.255	5.165	0.02	
2T Avg	2.939	0.0	-68.913	0.043	-17.78	-7.955	-4.852	0.079	-96.439
2T Std Dev	0.994	0.0	23.164	0.043	3.292	1.39	4.852	0.02	
3T Avg	2.809	0.0	-69.92	0.084	-20.881	-9.116	-9.549	0.103	-106.47
3T Std Dev	0.967	0.0	23.038	0.08	1.458	0.567	9.549	0.02	
4T Avg	2.651	0.0	-69.576	0.095	-20.183	-8.88	-8.994	0.103	-104.784
4T Std Dev	0.956	0.0	23.09	0.095	1.685	0.573	8.994	0.02	

All values reported in pN nm

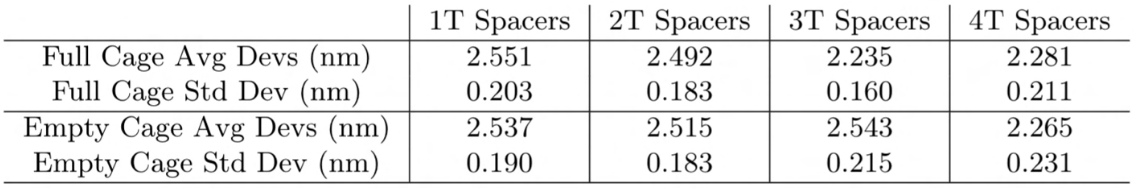
Table 8. Average energy broken down by oxDNA forcefield term for the four nucleotides centered at the nick in the cage's base with KDPG aldolase.



Contribution:	FENE	BEXC	STCK	NEXC	HB	CRSTCK	CXSTCK	DH	Total
1T Avg	2.897	0.0	-69.984	0.088	-21.534	-9.401	-10.385	0.121	-108.198
1T Std Dev	0.991	0.0	23.032	0.088	1.071	0.37	10.377	0.019	
2T Avg	2.77	0.003	-69.894	0.081	-21.563	-9.492	-10.174	0.12	-108.149
2T Std Dev	0.952	0.003	22.984	0.08	1.013	0.305	10.174	0.017	
3T Avg	2.884	0.005	-70.007	0.084	-21.447	-9.446	-10.232	0.119	-108.04
3T Std Dev	0.959	0.003	22.927	0.081	1.04	0.333	10.232	0.017	
4T Avg	2.906	0.001	-70.34	0.064	-21.41	-9.437	-9.998	0.118	-108.096
4T Std Dev	0.978	0.001	23.064	0.06	0.962	0.311	9.998	0.016	

All values reported in pN nm

Table 9. Average energy broken down by oxDNA forcefield term for the four nucleotides centered at the nick in the empty cage's base.



	1T Spacers	2T Spacers	3T Spacers	4T Spacers
Full Cage Avg Devs (nm)	2.551	2.492	2.235	2.281
Full Cage Std Dev (nm)	0.203	0.183	0.160	0.211
Empty Cage Avg Devs (nm)	2.537	2.515	2.543	2.265
Empty Cage Std Dev (nm)	0.190	0.183	0.215	0.231

Table 10. Average and standard deviation of full and empty cages' root mean squared fluctuations from the mean structure.

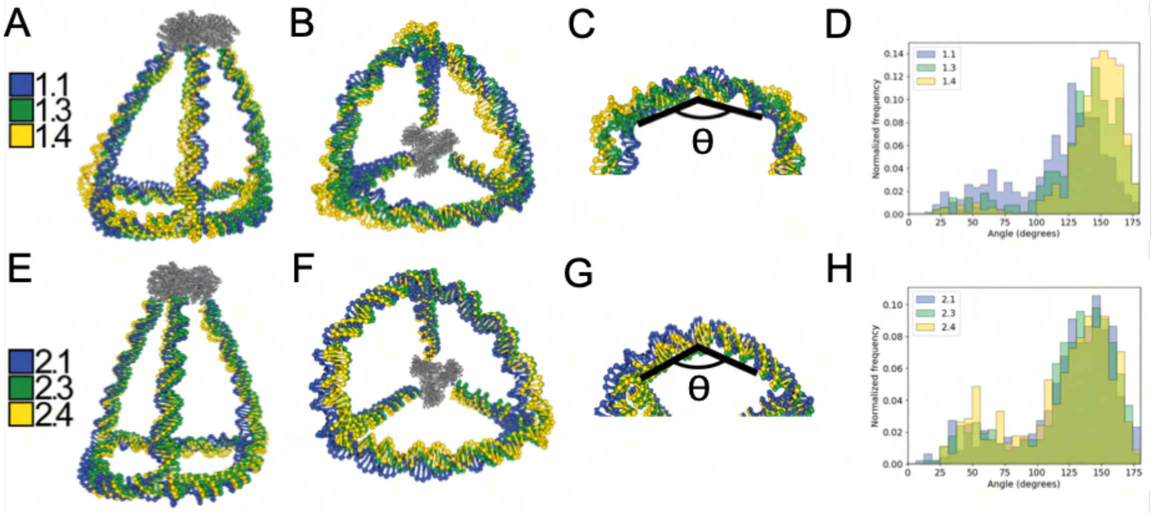


Figure 34. Asymmetric Cage Designs. Panels A-B show the aligned mean structures of the asymmetric designs holding the arm across from the nick point constant at 2T with variable T spacers in the other two arms. Panels C-D show the angle being measured at the nick point and the distribution of that angle across the simulation trajectories. Panels E-F show the aligned mean structures of the asymmetric designs varying the T spacers in the arm across from the nick point and holding the other two arms constant at 2T spacers. Panels G-H show the angle being measured at the nick point and the distribution of that angle across the simulation trajectories.

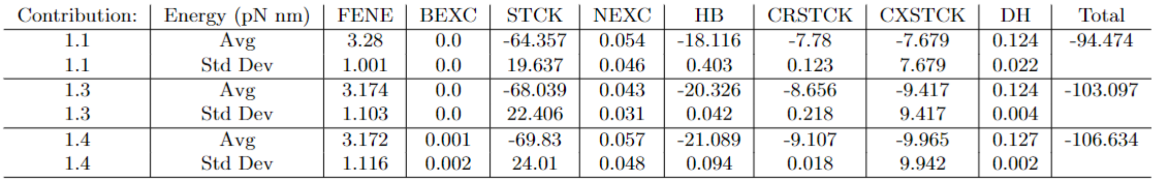
Additionally, two sets of asymmetric cages were designed to explore the differences in adding T spacers in the different arms of the cages.

One set of asymmetric systems was created by holding the arm across from the nick point constant at a 2T spacer, and varying the T spacer amount of the other two arms of the DNA cage with either 1T, 3T, or 4T spacers. Respectively these designs were named 1.1, 1.3, and 1.4.

The second set of asymmetric systems was created by holding the two arms attached to the nick point constant at 2T spacers, and varying the T spacer amount of the one arm across from the nick point with either 1T, 3T, or 4T spacers. Respectively these designs were named 2.1, 2.3, and 2.4.

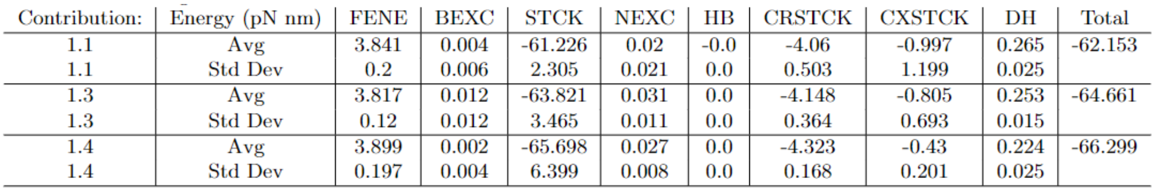
All six asymmetric designs mentioned in the main text were simulated using the same exact methodology as the symmetric cages. Below is a figure summarizing the effect of the asymmetric cages on the nick point and tables summarizing the average energy of each individual arm and the nick point for each individual asymmetric design.

The average energies of the asymmetric cages were also computed and shown below in tables 11, 12, 13, 14, 15, 16, 17, 18.



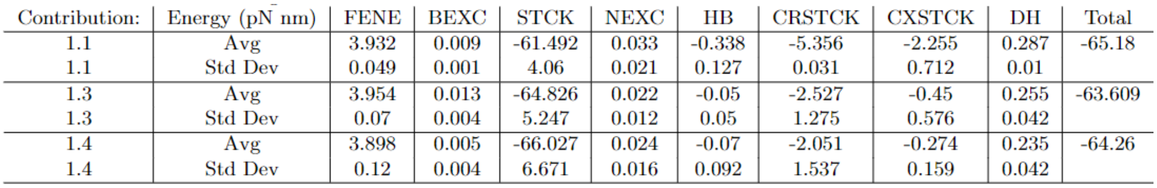
Contribution:	Energy (pN nm)	FENE	BEXC	STCK	NEXC	HB	CRSTCK	CXSTCK	DH	Total
1.1	Avg	3.28	0.0	-64.357	0.054	-18.116	-7.78	-7.679	0.124	-94.474
1.1	Std Dev	1.001	0.0	19.637	0.046	0.403	0.123	7.679	0.022	
1.3	Avg	3.174	0.0	-68.039	0.043	-20.326	-8.656	-9.417	0.124	-103.097
1.3	Std Dev	1.103	0.0	22.406	0.031	0.042	0.218	9.417	0.004	
1.4	Avg	3.172	0.001	-69.83	0.057	-21.089	-9.107	-9.965	0.127	-106.634
1.4	Std Dev	1.116	0.002	24.01	0.048	0.094	0.018	9.942	0.002	

Table 11. Average energy broken down by oxDNA forcefield term for the four nucleotides centered at the nick in the asymmetric cages with the arm across from the nick point staying constant.



Contribution:	Energy (pN nm)	FENE	BEXC	STCK	NEXC	HB	CRSTCK	CXSTCK	DH	Total
1.1	Avg	3.841	0.004	-61.226	0.02	-0.0	-4.06	-0.997	0.265	-62.153
1.1	Std Dev	0.2	0.006	2.305	0.021	0.0	0.503	1.199	0.025	
1.3	Avg	3.817	0.012	-63.821	0.031	0.0	-4.148	-0.805	0.253	-64.661
1.3	Std Dev	0.12	0.012	3.465	0.011	0.0	0.364	0.693	0.015	
1.4	Avg	3.899	0.002	-65.698	0.027	0.0	-4.323	-0.43	0.224	-66.299
1.4	Std Dev	0.197	0.004	6.399	0.008	0.0	0.168	0.201	0.025	

Table 12. Average energy broken down by oxDNA forcefield term for the four T spacers nucleotides (2T spacer) in the arm across from the nick point.

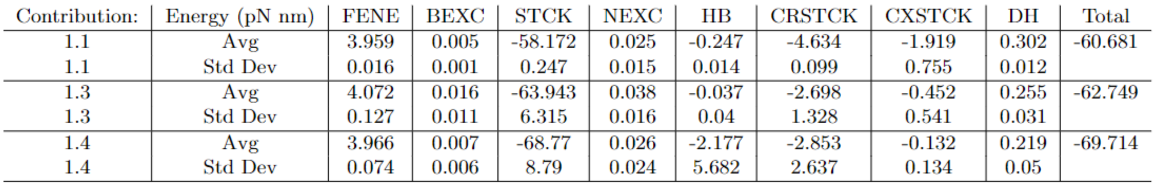


Contribution:	Energy (pN nm)	FENE	BEXC	STCK	NEXC	HB	CRSTCK	CXSTCK	DH	Total
1.1	Avg	3.932	0.009	-61.492	0.033	-0.338	-5.356	-2.255	0.287	-65.18
1.1	Std Dev	0.049	0.001	4.06	0.021	0.127	0.031	0.712	0.01	
1.3	Avg	3.954	0.013	-64.826	0.022	-0.05	-2.527	-0.45	0.255	-63.609
1.3	Std Dev	0.07	0.004	5.247	0.012	0.05	1.275	0.576	0.042	
1.4	Avg	3.898	0.005	-66.027	0.024	-0.07	-2.051	-0.274	0.235	-64.26
1.4	Std Dev	0.12	0.004	6.671	0.016	0.092	1.537	0.159	0.042	

Table 13. Average energy broken down by oxDNA forcefield term for the variable number of T spacers nucleotides in the left arm (when viewed with the nick point in front and arm held constant in the back).

A.8 Cryo Fitting Data

Tables 19, 20, 21, & 22 The below tables, report the Chimera fitting values for all 10 mean simulation models while maximizing the fit for correlation. Each table shows fitting results for the specified cryo map (filled or empty) at the temperature of the simulation models used.



Contribution:	Energy (pN nm)	FENE	BEXC	STCK	NEXC	HB	CRSTCK	CXSTCK	DH	Total
1.1	Avg	3.959	0.005	-58.172	0.025	-0.247	-4.634	-1.919	0.302	-60.681
1.1	Std Dev	0.016	0.001	0.247	0.015	0.014	0.099	0.755	0.012	
1.3	Avg	4.072	0.016	-63.943	0.038	-0.037	-2.698	-0.452	0.255	-62.749
1.3	Std Dev	0.127	0.011	6.315	0.016	0.04	1.328	0.541	0.031	
1.4	Avg	3.966	0.007	-68.77	0.026	-2.177	-2.853	-0.132	0.219	-69.714
1.4	Std Dev	0.074	0.006	8.79	0.024	5.682	2.637	0.134	0.05	

Table 14. Average energy broken down by oxDNA forcefield term for the variable number of T spacers nucleotides in the right arm (when viewed with the nick point in front and arm held constant in the back).

Contribution:	Energy (pN nm)	FENE	BEXC	STCK	NEXC	HB	CRSTCK	CXSTCK	DH	Total
2.1	Avg	3.201	0.001	-66.613	0.039	-19.358	-8.366	-8.66	0.123	-99.633
2.1	Std Dev	1.182	0.001	21.809	0.039	0.472	0.186	8.66	0.015	
2.3	Avg	3.239	0.001	-66.119	0.03	-18.739	-8.168	-8.442	0.124	-98.074
2.3	Std Dev	1.139	0.001	21.376	0.025	0.613	0.156	8.442	0.018	
2.4	Avg	3.156	0.008	-64.948	0.046	-18.743	-7.933	-7.876	0.123	-96.167
2.4	Std Dev	1.051	0.014	20.415	0.034	0.751	0.213	7.876	0.015	

Table 15. Average energy broken down by oxDNA forcefield term for the four nucleotides centered at the nick in the asymmetric cages with the arm across from the nick point being varied in T spacer number.

Contribution:	Energy (pN nm)	FENE	BEXC	STCK	NEXC	HB	CRSTCK	CXSTCK	DH	Total
2.1	Avg	3.982	0.011	-66.405	0.041	0.0	-5.732	-2.312	0.286	-70.129
2.1	Std Dev	0.024	0.008	1.665	0.003	0.0	0.159	0.076	0.009	
2.3	Avg	3.814	0.007	-64.429	0.025	-0.0	-3.155	-0.427	0.234	-63.931
2.3	Std Dev	0.08	0.005	8.086	0.013	0.0	1.56	0.526	0.034	
2.4	Avg	3.923	0.008	-67.764	0.029	-2.795	-3.099	-0.229	0.226	-69.701
2.4	Std Dev	0.158	0.011	11.013	0.014	7.395	2.931	0.336	0.055	

Table 16. Average energy broken down by oxDNA forcefield term for the variable number of T spacer nucleotides in the arm across from the nick point.

Contribution:	Energy (pN nm)	FENE	BEXC	STCK	NEXC	HB	CRSTCK	CXSTCK	DH	Total
2.1	Avg	4.032	0.027	-59.3	0.04	-0.188	-3.195	-1.239	0.288	-59.535
2.1	Std Dev	0.083	0.019	3.716	0.021	0.206	0.206	1.167	0.034	
2.3	Avg	3.953	0.016	-60.754	0.011	-0.042	-3.278	-0.734	0.256	-60.572
2.3	Std Dev	0.231	0.012	4.492	0.01	0.074	0.378	0.608	0.016	
2.4	Avg	3.992	0.016	-63.775	0.01	-0.17	-3.462	-0.691	0.243	-63.837
2.4	Std Dev	0.065	0.012	6.127	0.004	0.181	0.535	0.364	0.017	

Table 17. Average energy broken down by oxDNA forcefield term for the four T spacer nucleotides in the left arm (when viewed with the nick point in front and variable arm in the back).

Contribution:	Energy (pN nm)	FENE	BEXC	STCK	NEXC	HB	CRSTCK	CXSTCK	DH	Total
2.1	Avg	3.904	0.013	-58.857	0.018	-0.036	-3.43	-0.837	0.275	-58.95
2.1	Std Dev	0.21	0.007	3.426	0.01	0.029	0.202	0.73	0.032	
2.3	Avg	3.869	0.001	-59.81	0.05	-0.058	-3.435	-0.96	0.267	-60.076
2.3	Std Dev	0.21	0.002	2.764	0.044	0.059	0.19	0.859	0.026	
2.4	Avg	3.903	0.003	-59.92	0.025	-0.104	-3.486	-0.928	0.273	-60.234
2.4	Std Dev	0.108	0.002	2.301	0.004	0.029	0.161	0.721	0.019	

Table 18. Average energy broken down by oxDNA forcefield term for the four T spacer nucleotides in the right arm (when viewed with the nick point in front and variable arm in the back).

Empty Cage					
113K	0	1	2	3	4
Correlation	0.7875	0.8002	0.8154	0.7847	0.7673
Overlap	64.1	65.31	67.69	63.42	61.26

Table 19. Fitting results of atomic models of the mean structure of the DNA cage fit to the cryo map of the empty cage. The 0-4 indicate the number of PDNA incorporation for the simulation mean structures from their 113K simulation.

Filled Cage					
113K	0	1	2	3	4
Correlation	0.7932	0.7891	0.8076	0.7944	0.755
Overlap	32.36	32.33	33.57	32.77	29.81

Table 20. Fitting results of atomic models of the mean structure of the DNA cage fit to the cryo map of the filled cage. The 0-4 indicate the number of PDNA incorporation for the simulation mean structures from their 113K simulation.

Empty Cage					
300K	0	1	2	3	4
Correlation	0.8054	0.7981	0.8059	0.7974	0.7861
Overlap	66.76	65.72	66.92	66.59	64.59

Table 21. Fitting results of atomic models of the mean structure of the DNA cage fit to the cryo map of the empty cage. The 0-4 indicate the number of PDNA incorporation for the simulation mean structures from their 300K simulation.

Cryo map fittings of the mean structures are at 300K are shown in the main text Figure 15. The fittings of the mean structures at 113K is shown in Figure 34.

Filled Cage					
300K	0	1	2	3	4
Correlation	0.8111	0.795	0.8046	0.804	0.7951
Overlap	34.07	33.07	33.72	33.73	32.92

Table 22. Fitting results of atomic models of the mean structure of the DNA cage fit to the cryo map of the filled cage. The 0-4 indicate the number of PDNA incorporation for the simulation mean structures from their 300K simulation.

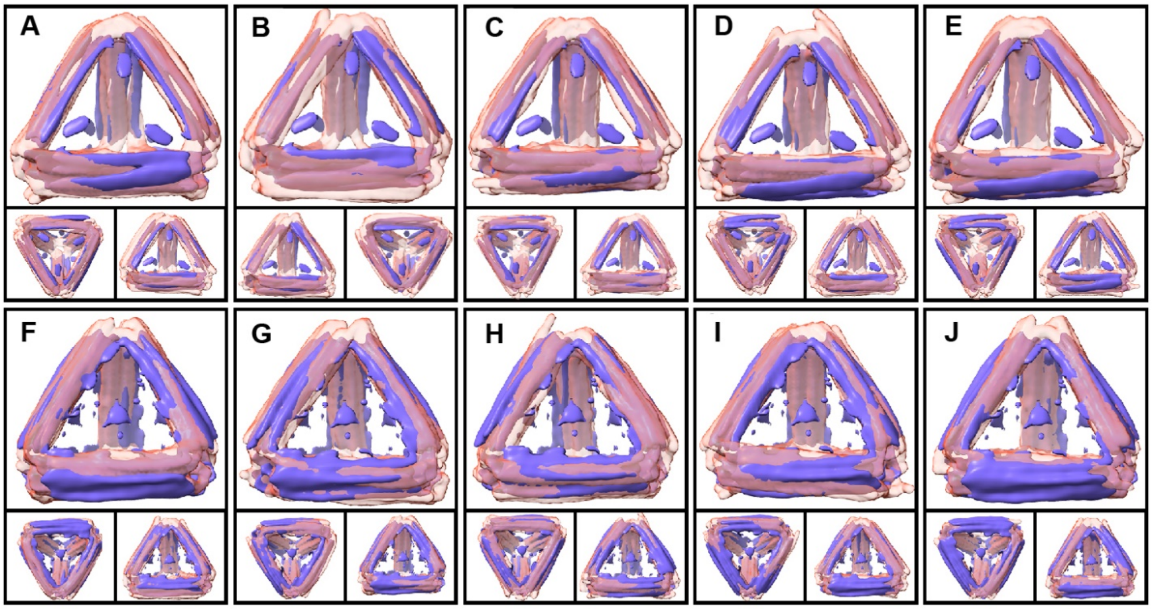


Figure 35. Fitting images of simulation mean structures at 113K. Atomic maps generated from the mean structures at the same resolution as the cryo map are shown in translucent pink and the cryo map itself shown in purple. Each sub-figure depicts three views of the same fitting. **A)** 0 PDNA fit to empty cage. **B)** 1 PDNA fit to empty cage. **C)** 2 PDNA fit to empty cage. **D)** 3 PDNA fit to empty cage. **E)** 4 PDNA fit to empty cage. **F)** 0 PDNA fit to filled cage. **G)** 1 PDNA fit to filled cage. **H)** 2 PDNA fit to filled cage. **I)** 3 PDNA fit to filled cage. **J)** 4 PDNA fit to filled cage.

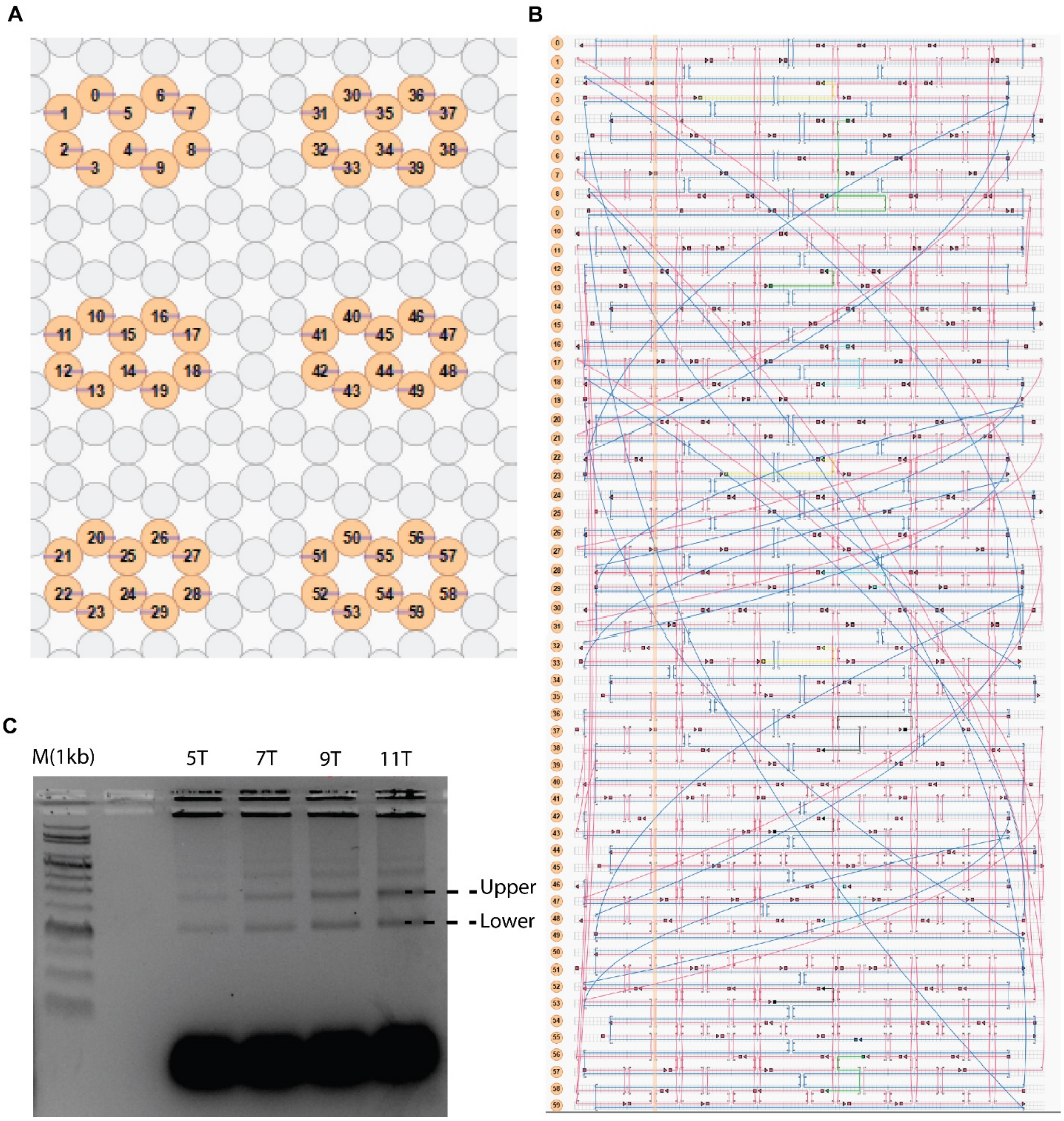


Figure 36. CADNANO design scheme of the tetrahedral origami cage. **A)** Ten helix bundles used for each edge. **B)** Design details of crossovers and connections. Light blue refers to the scaffold routing. Pink refers to the staple strands. Yellow, cyan, black and green represent the handle positions for each of the faces of the tetrahedron used for the incorporation of the PDNA. **C)** Agarose gel characterization of the tetrahedral frame with varying lengths of poly-thymidine linkers between arms. The bands shown as lower and upper were isolated and purified and characterized by negative stain TEM. Lane M(1kb)= 1kb ds ladder, Lanes 5T, 7T, 9T, 11T are origami structures assembled with varying poly-thymidines ranging from 5 to 11 respectively.

A.9 Additional Supplementary Figures and Sequences of DNA origami handles/staples

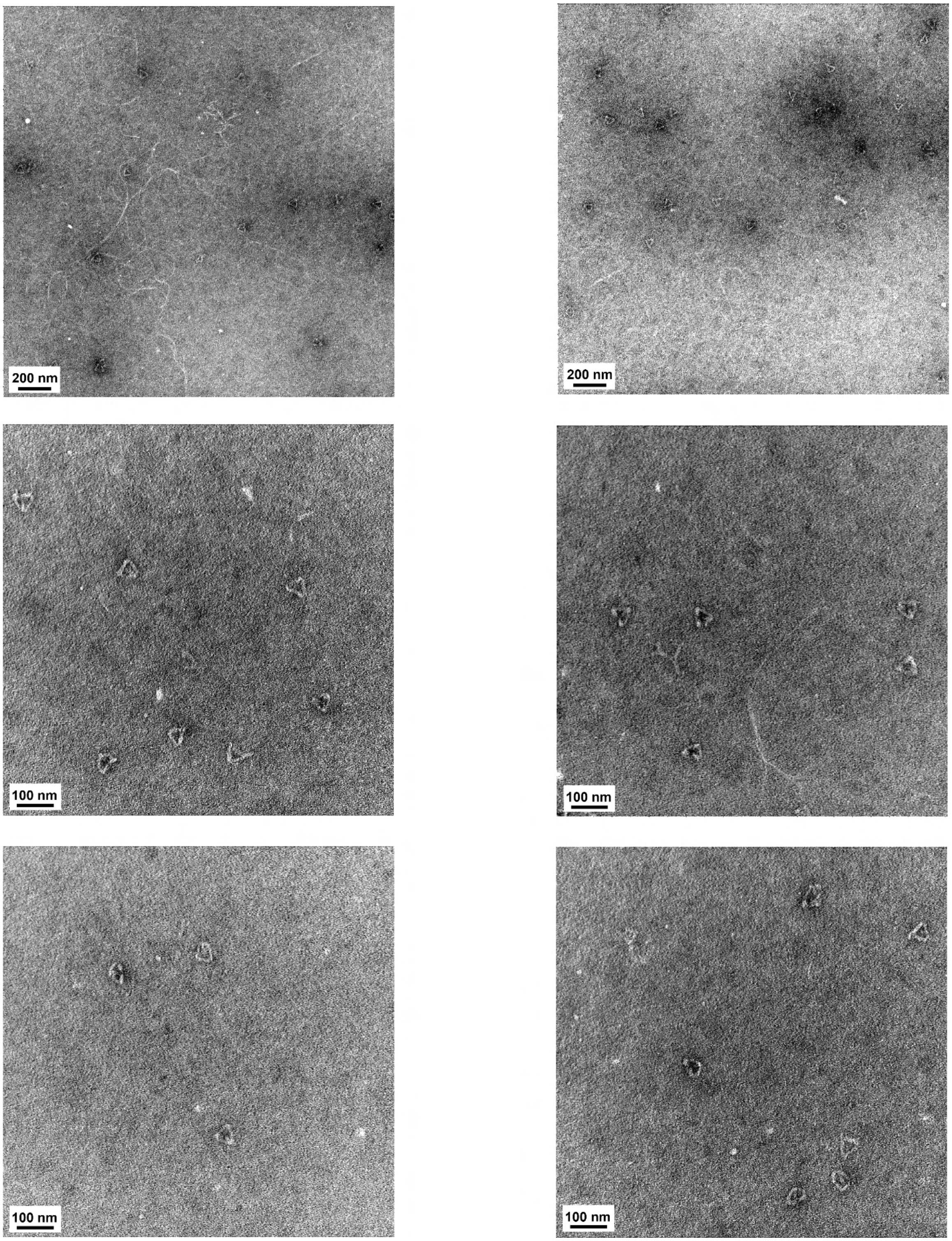


Figure 37. Images of the negatively stained Lower-monomer band from Figure 36C.

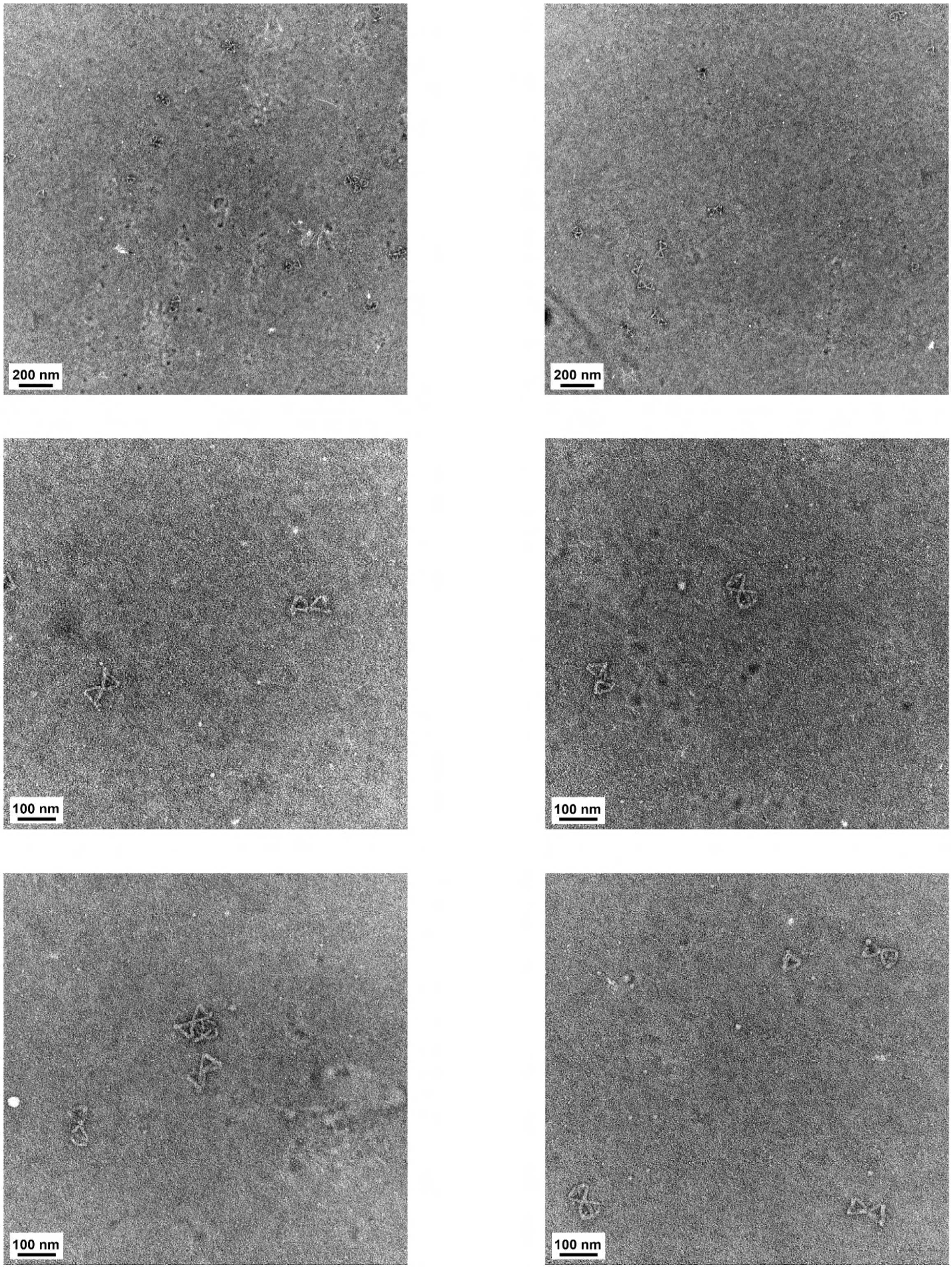


Figure 38. Images of the negatively stained Upper-Dimer band from Figure 36C.

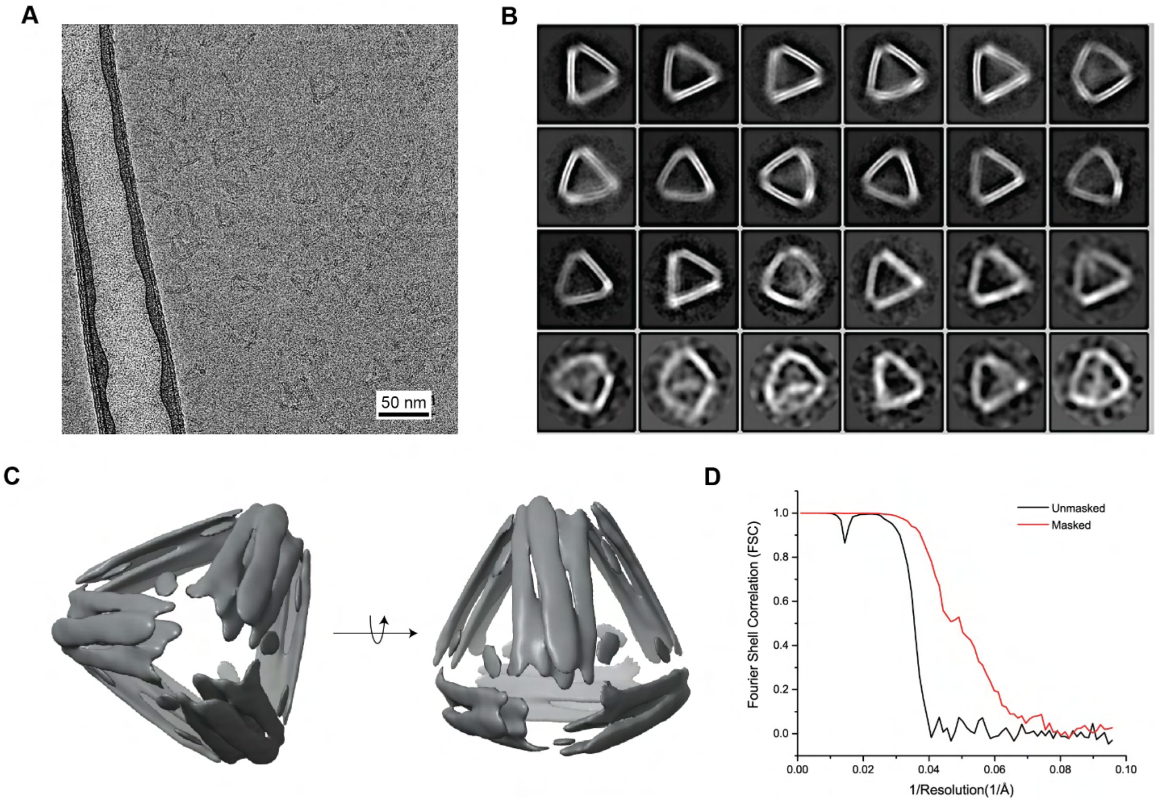


Figure 39. Cryo-EM analysis of the empty-cage tetrahedron. **A)** Electron micrograph of the cryogenically plunged tetrahedron. Black contrast represents the tetrahedral cages and white corresponds to the background. **B)** Representative 2D class averages. Box side lengths are 972 Å. **C)** Two different views of the cryo-EM density map. **D)** Fourier-shell correlation (FSC) plot of the 3D reconstruction.

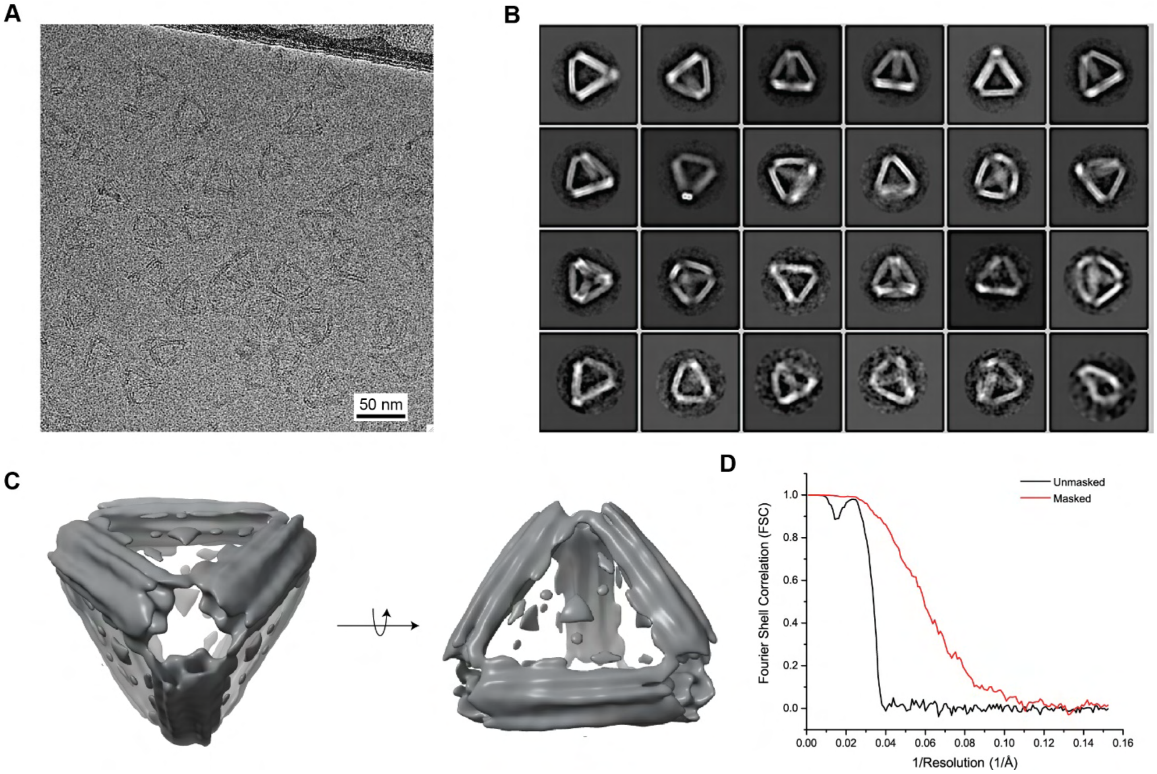
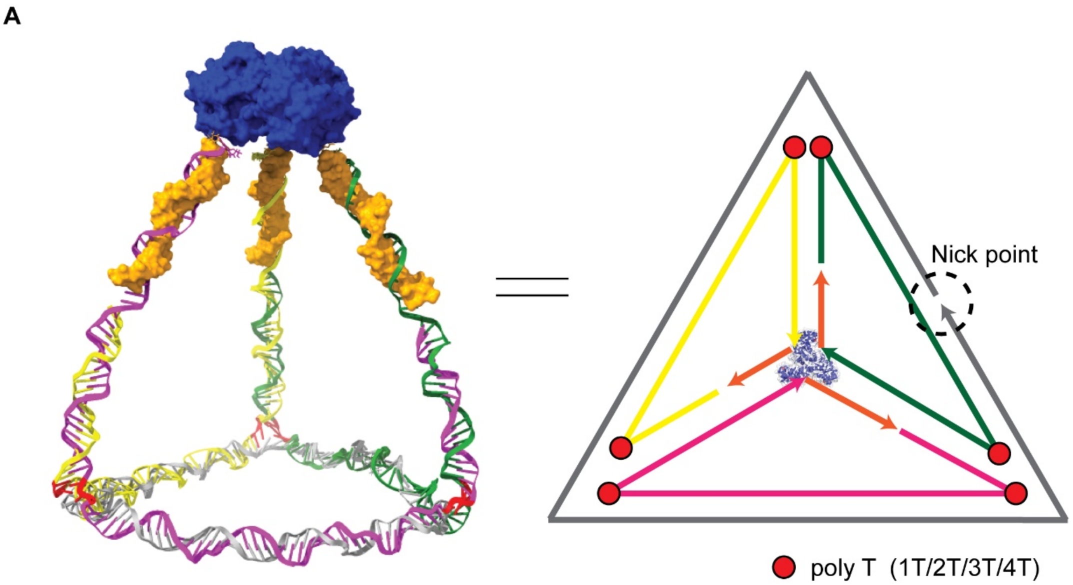


Figure 40. Cryo-EM analysis of the PDNA-bearing origami. **A)** Electron micrograph of the cryogenic PDNA bound tetrahedron. Black contrast represents the protein-bound tetrahedrons and white corresponds to the background. Scale bar indicates 50 nm. **B)** Representative 2D class averages. Box side lengths are 1,024 Å. **C)** Two different views of the cryo-EM density map. **D)** Fourier-shell correlation (FSC) plot of the 3D reconstruction.



B

Name	Sequence
Green	GAG CAG ACC TGA CGG AAC TCA AGG AGT GTG ATG GAG ATT TAT (1/2/3/4T)GA GAG AGA AGA TAG AGA GAT AAG AGA TAG ATA GAT AGA TAG ATT CCG TGT AGT GTT CAA CGC CT
Magenta	GAG CAG ACC TGA CGG AAC TCA AGG CGT TGA ACA CTA CAC GGA (1/2/3/4T)GA ATA GAT AGA TAG ATA GTA GAT AGA TGA TGA ATG AGA TGA GTC TCT CGT AGT TAA CAT CTA GC
Yellow	GAG CAG ACC TGA CGG AAC TCA GCT AGA TGT TAA CTA CGA GAG (1/2/3/4T)GA TGA GAG AGA AGA AG AGA GAT AAG TAG ATA GTA GAT AGT ATA TAA ATC TCC ATC ACA CTC CT
Ash	ATC TCT CTA TCT TCT CTC TC T TAC TAT CTA CTA TCT ACT TAT CTC TCT ATC TTC TCT CTC ATC TCT CAT CTC ATC TAT CAT CTA TCT ACT ATC TAT CTA TCT ATT CTT CTA TCT ATC TAT CTA TCT CTT

Figure 41. Line diagram and sequences of the 4 turn-DNA PDTet cage. **A)** Schematic illustration and line diagram of the 4 turn DNA tetrahedron. **B)** Sequences of the strands used, wherein the boxed region shows the complimentary area to the DNA conjugate and area marked in red shows the variations in the poly-thymidine resides used in the 1T, 2T, 3T and 4T variations.

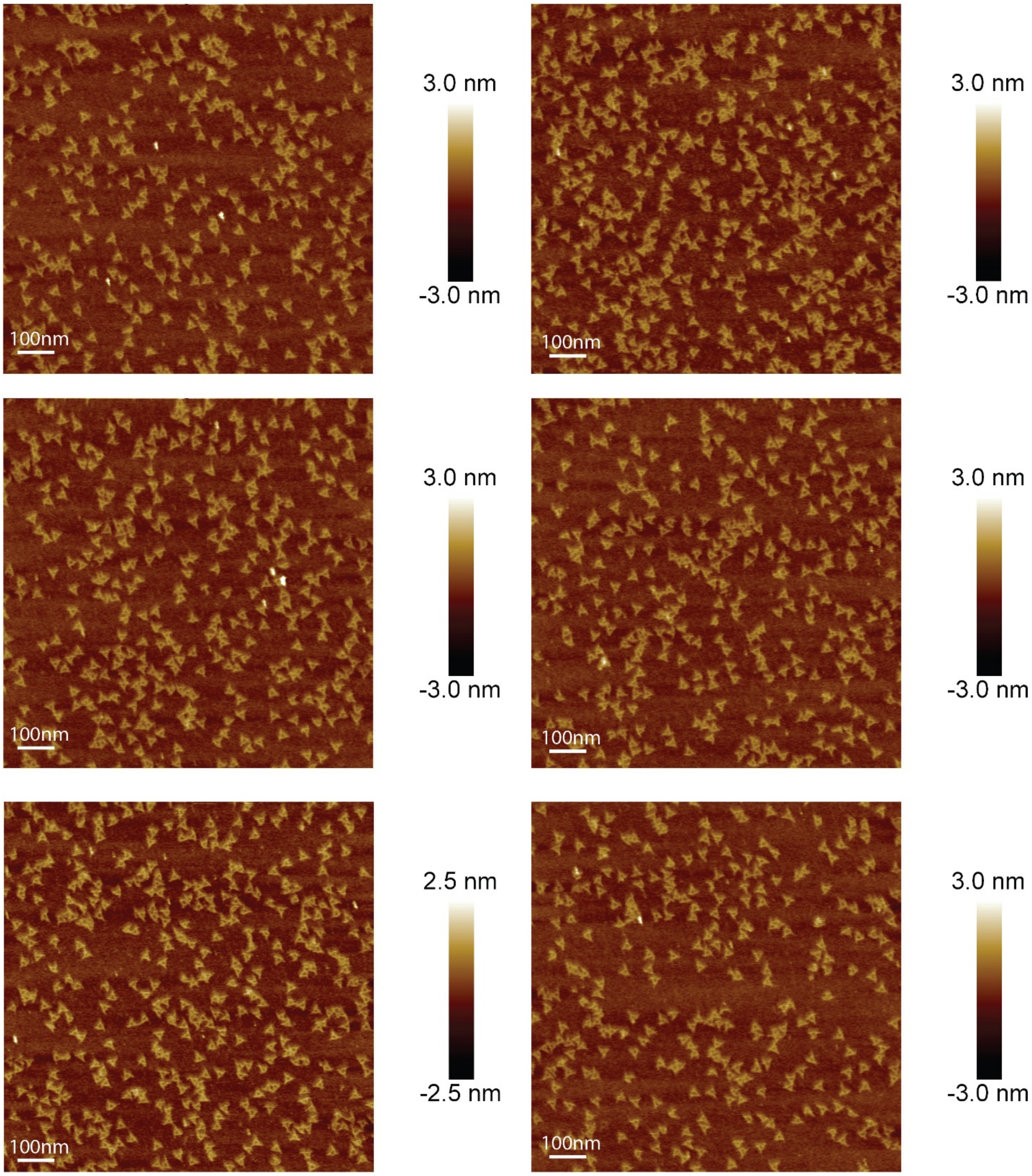


Figure 42. AFM images of the Open 1T triangular base structures.

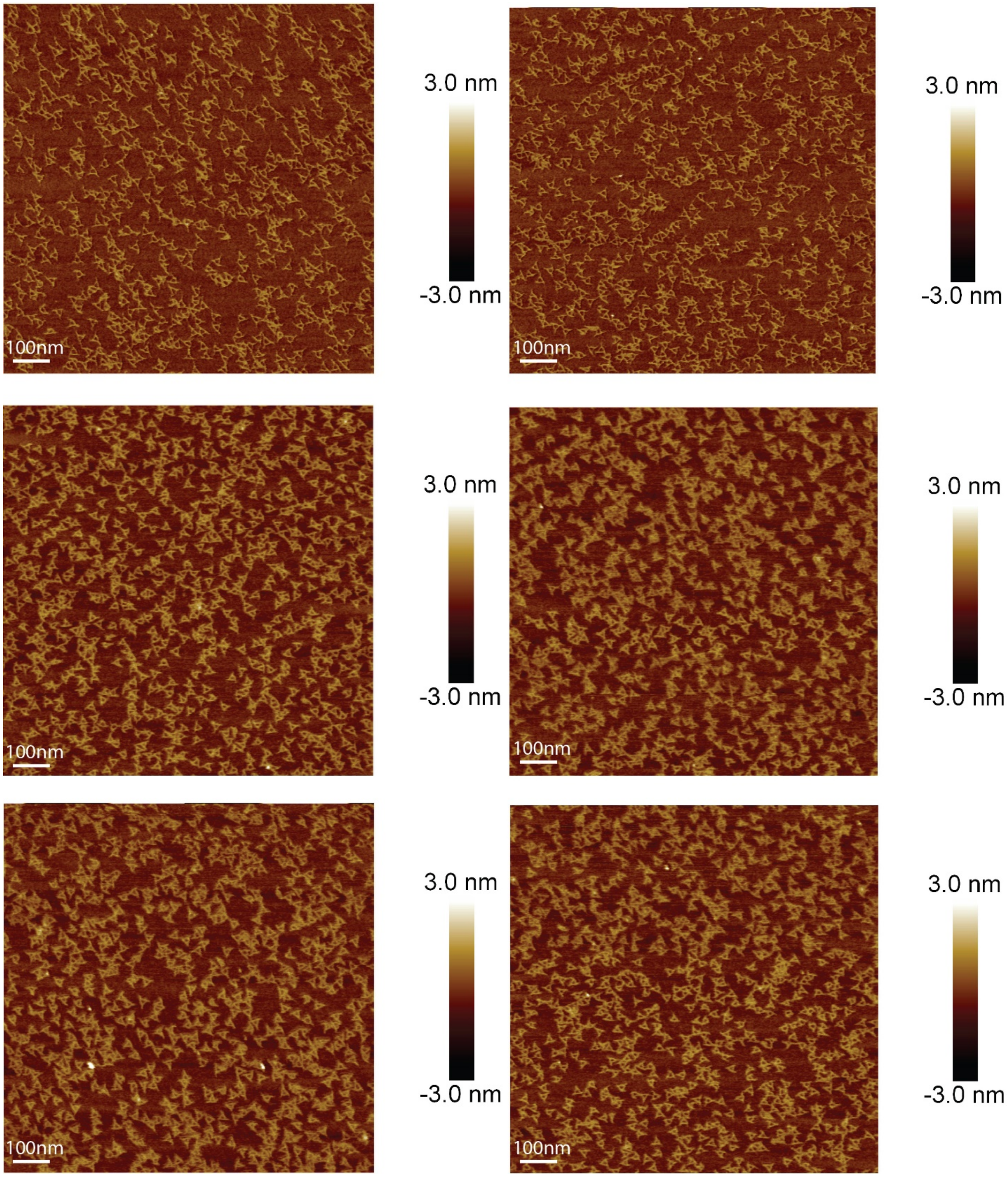


Figure 43. AFM images of the Open 2T triangular base structures.

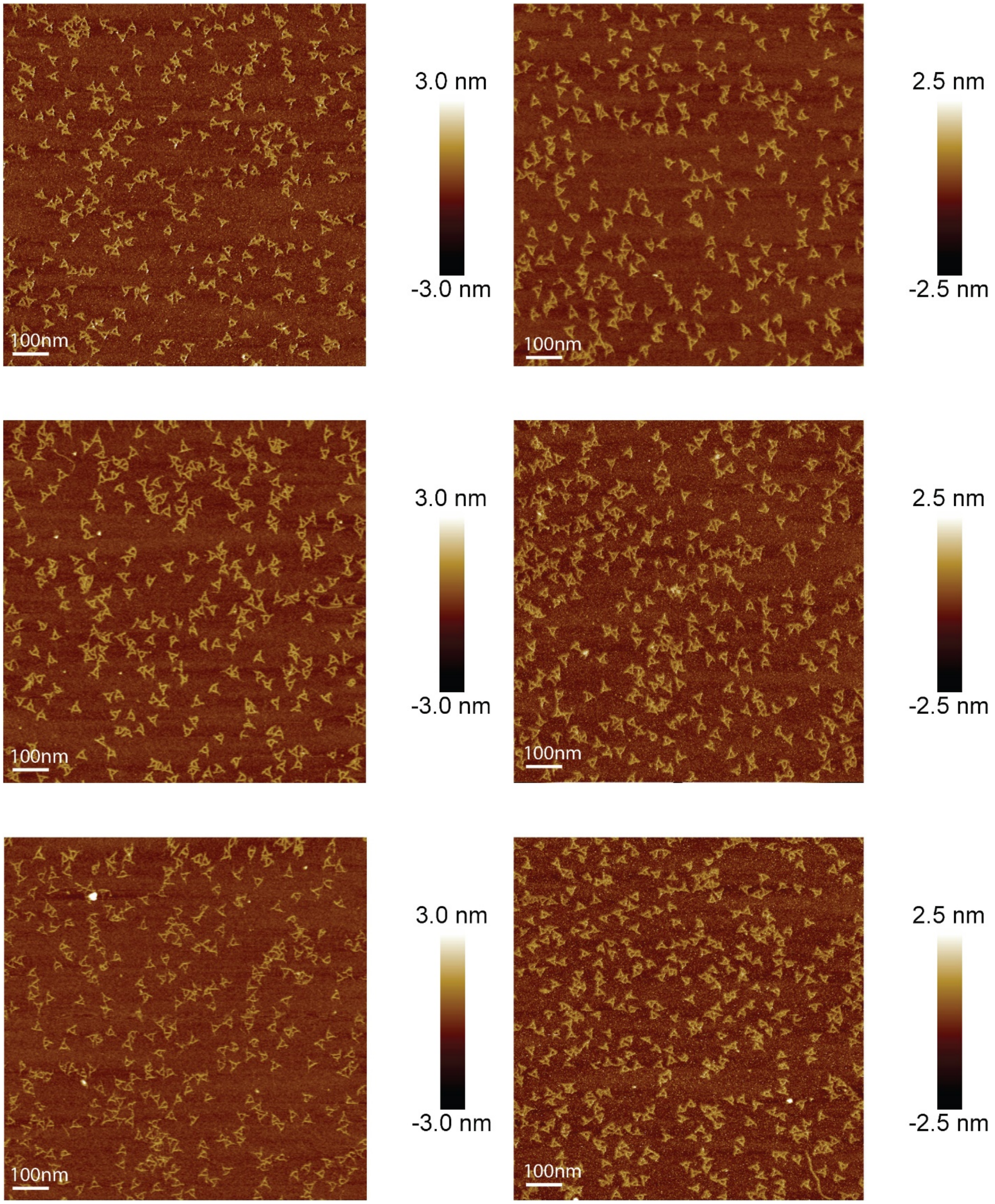


Figure 44. AFM images of the Open 3T triangular base structures.

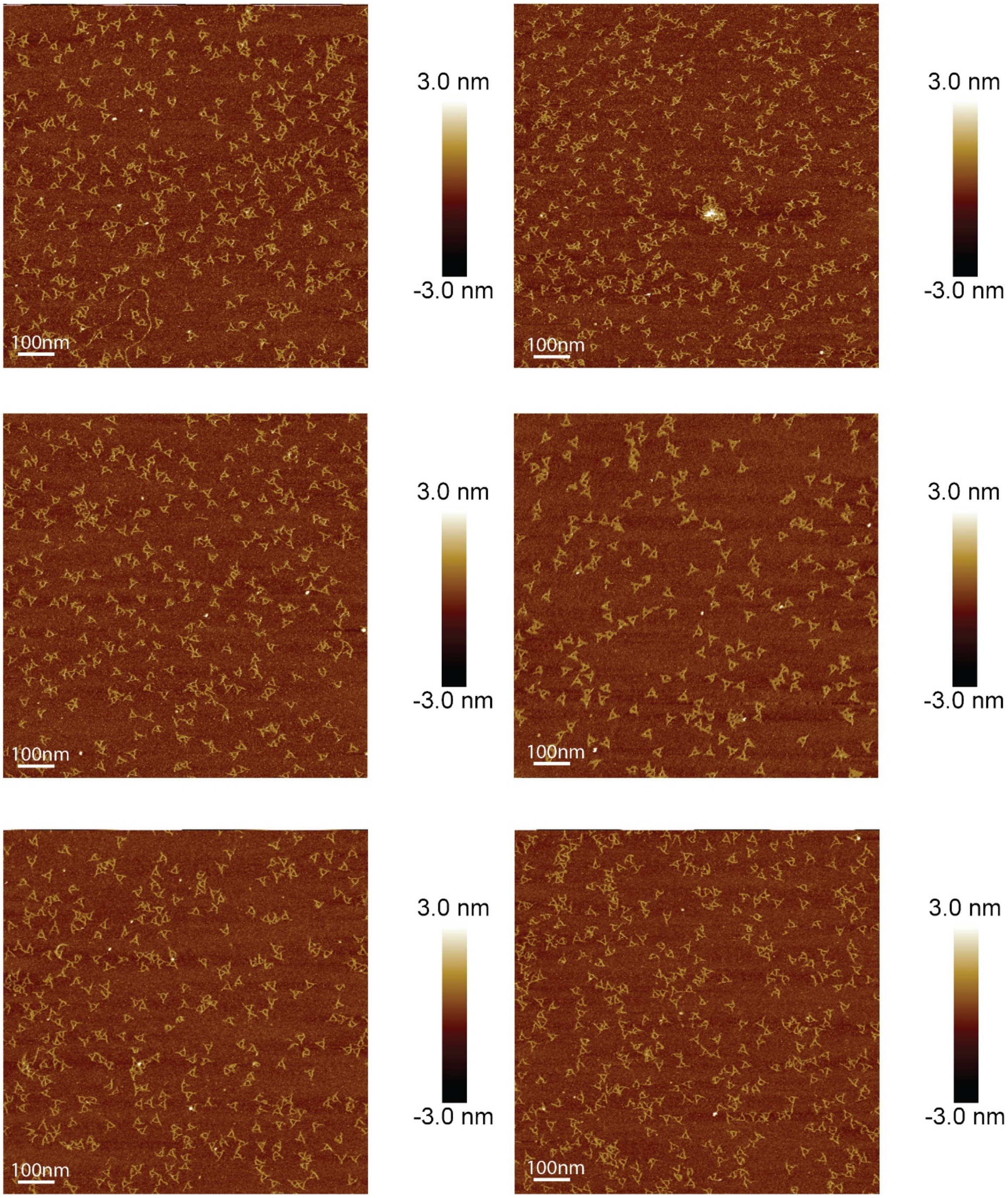


Figure 45. AFM images of the Open 4T triangular base structures.

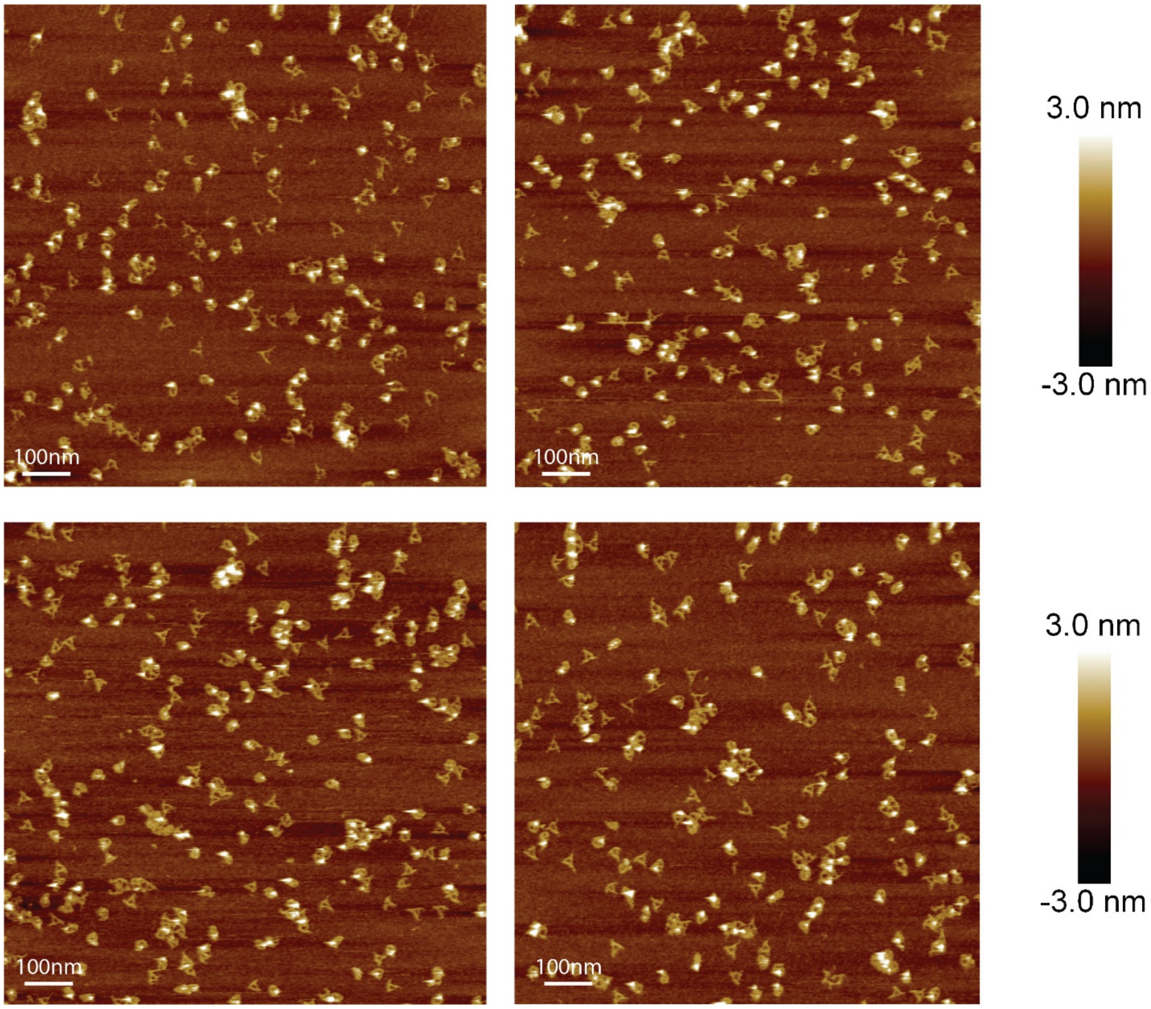


Figure 46. AFM images of the crude-Closed 1T tetrahedron.

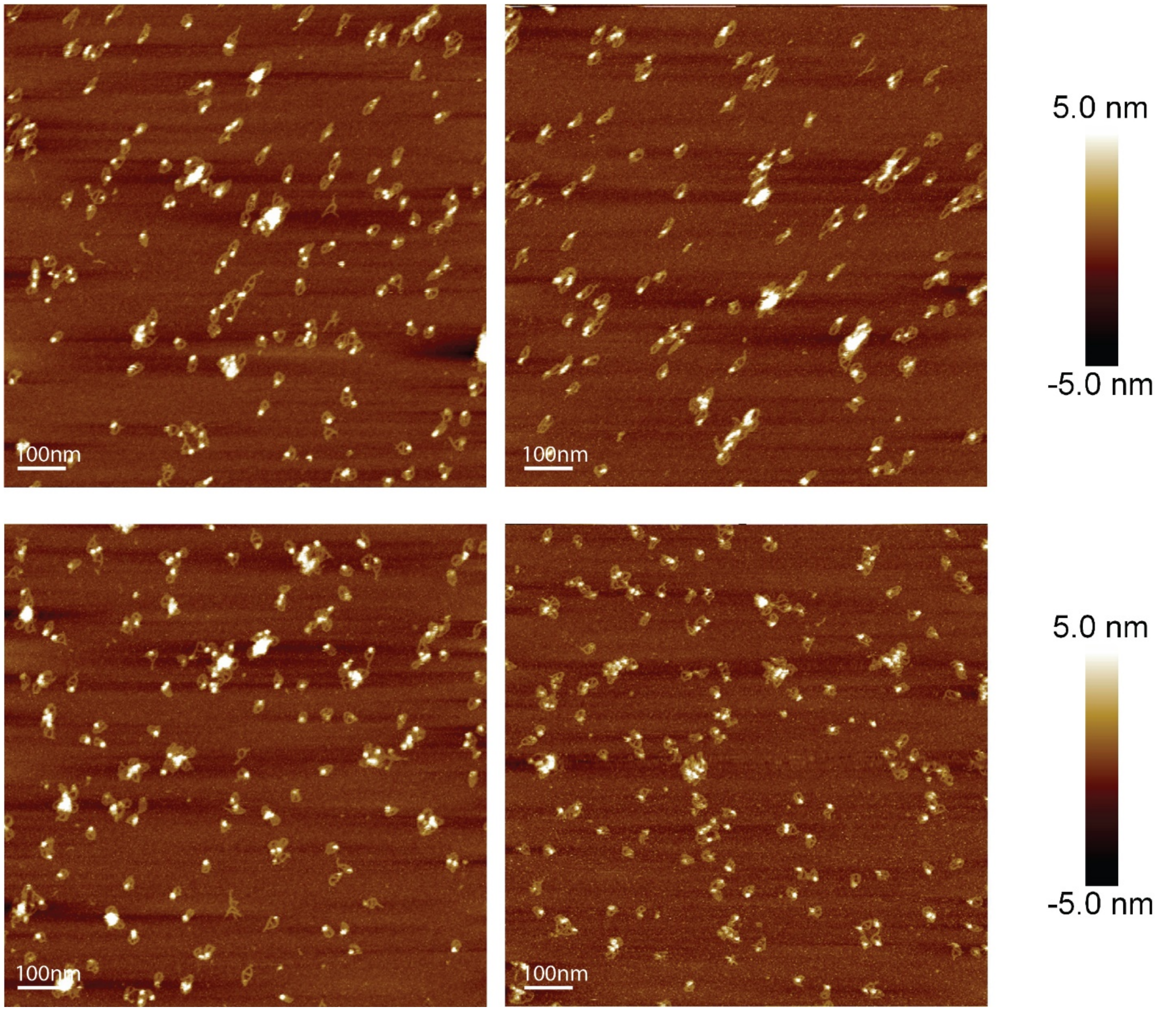


Figure 47. AFM images of the crude-Closed 2T tetrahedron.

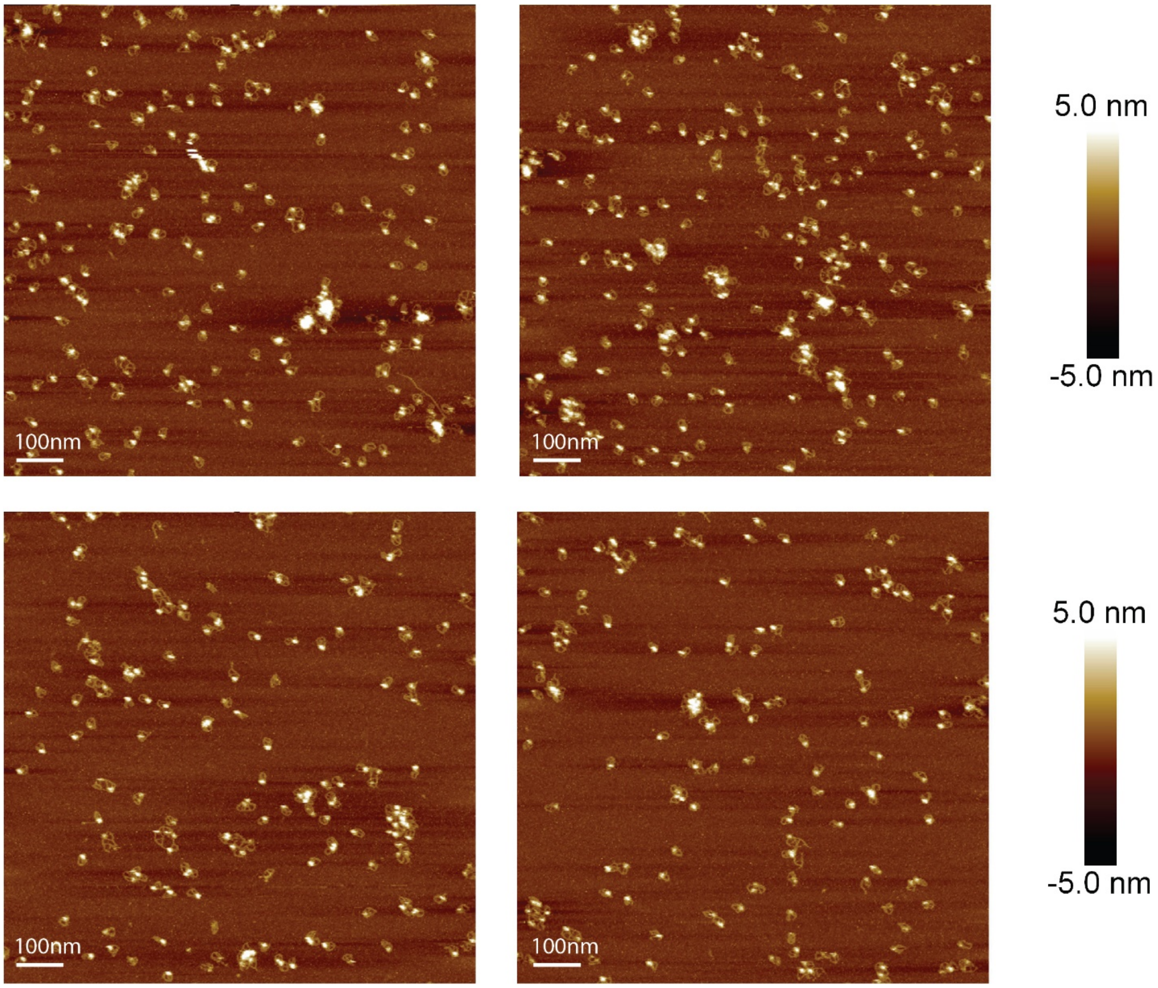


Figure 48. AFM images of the crude-Closed 3T tetrahedron.

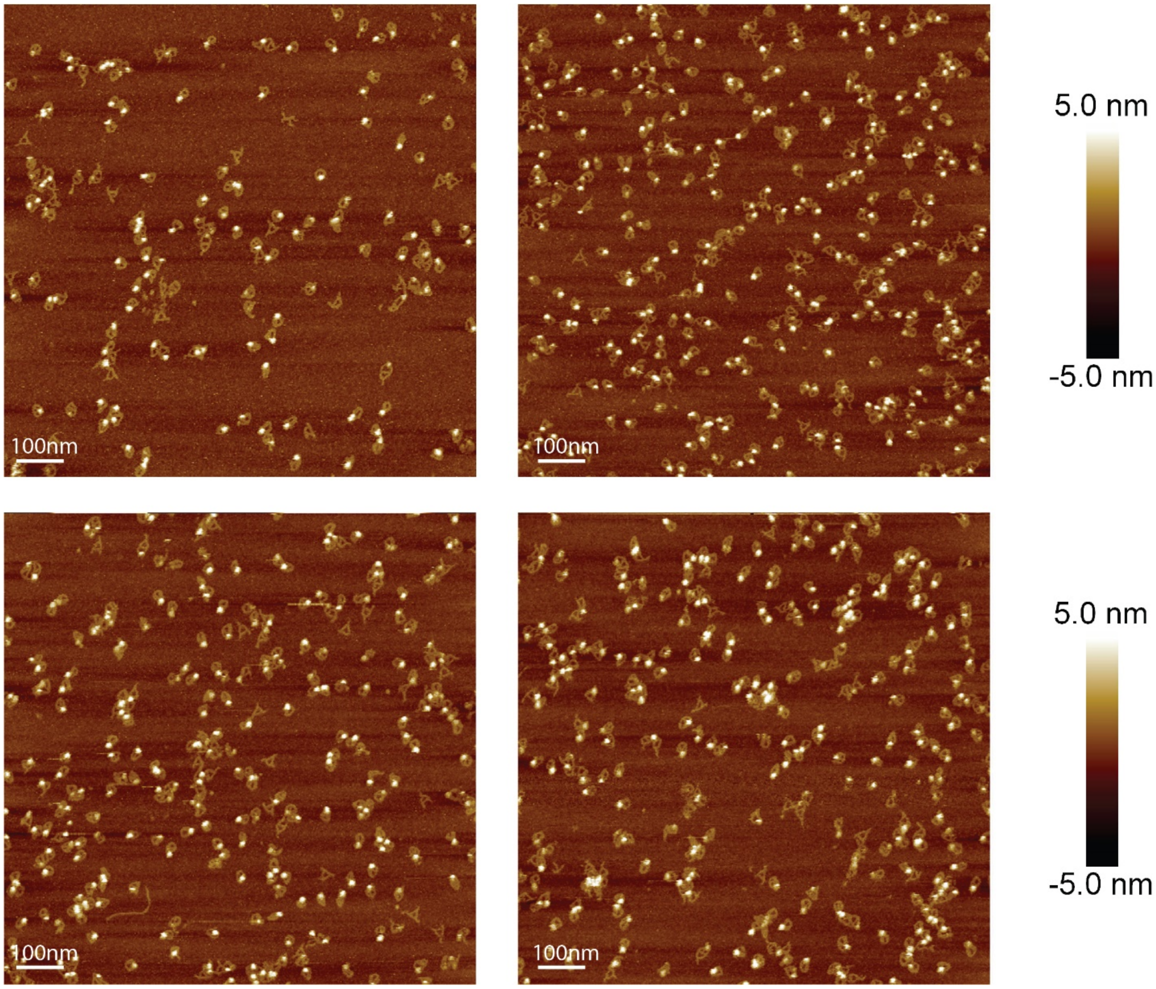
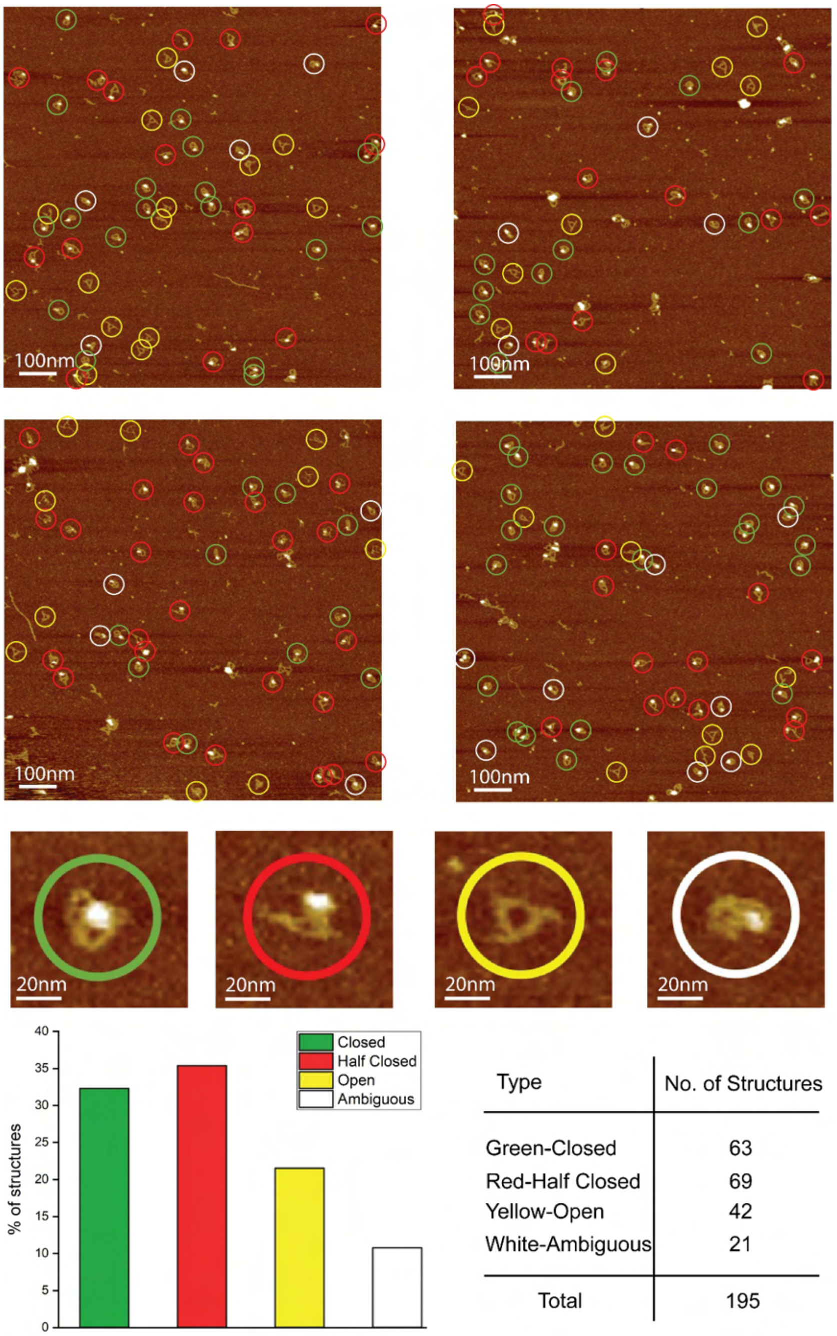
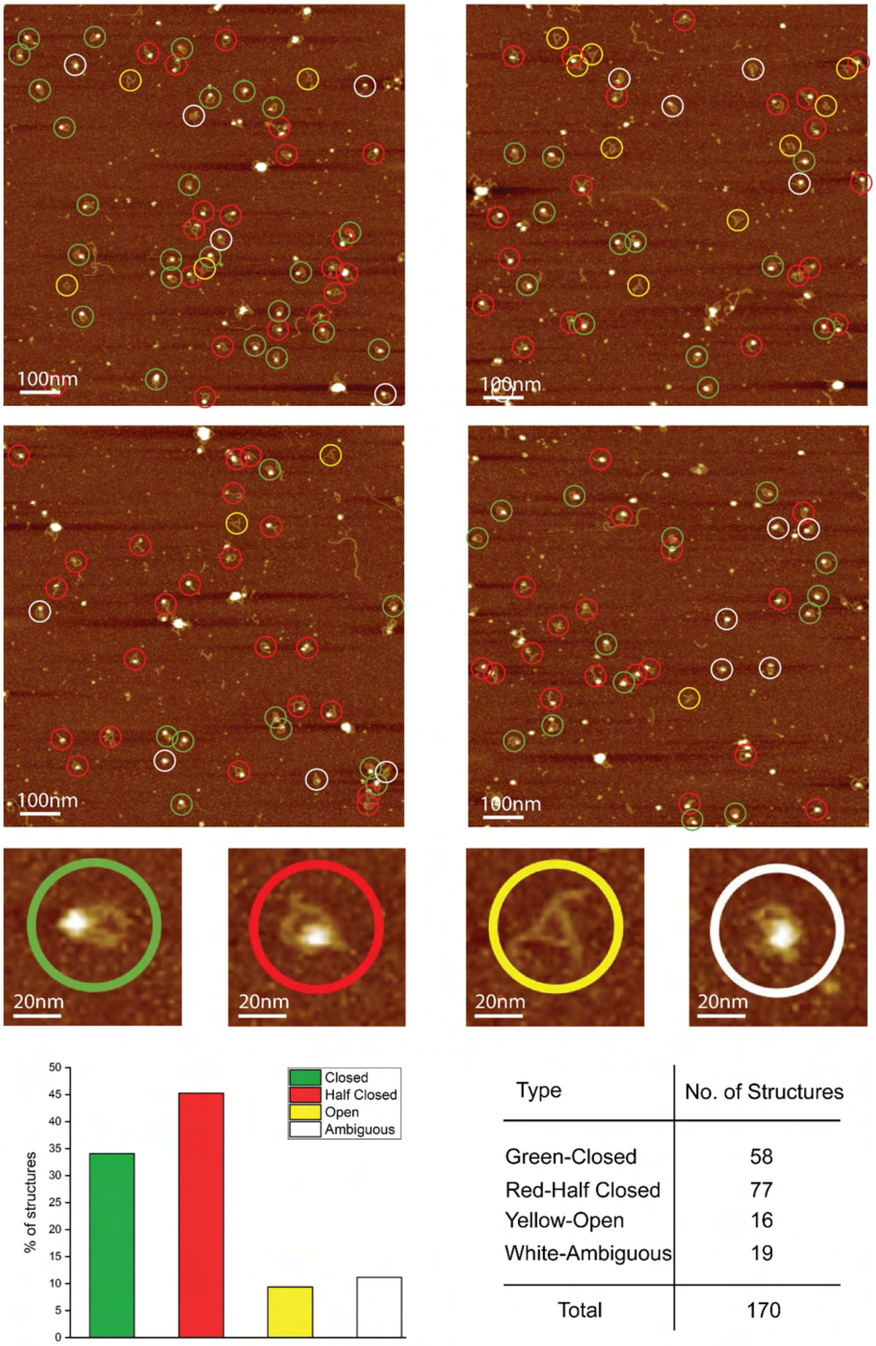


Figure 49. AFM images of the crude-Closed 4T tetrahedron.



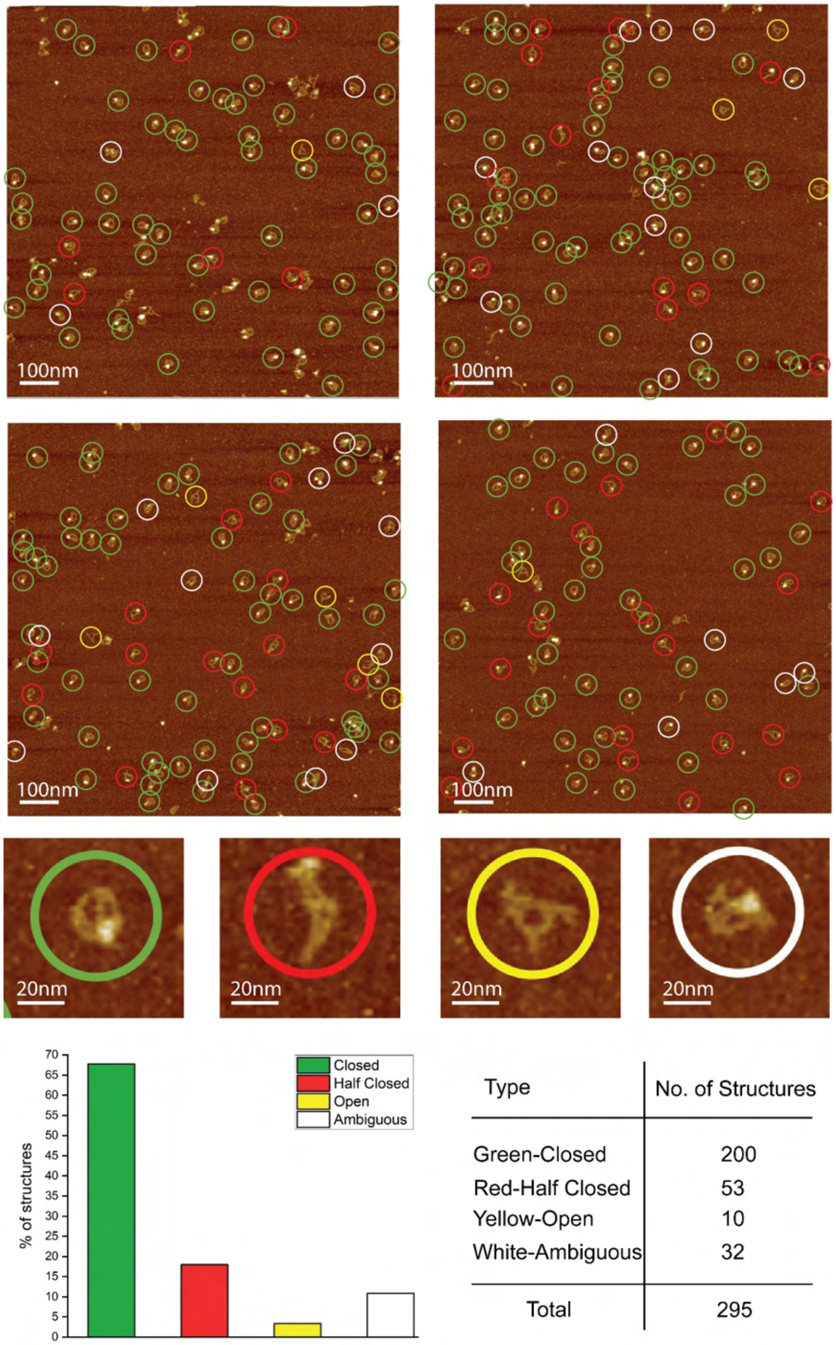
Type	No. of Structures
Green-Closed	63
Red-Half Closed	69
Yellow-Open	42
White-Ambiguous	21
Total	195

Figure 50. AFM images and yield for the PAGE-purified Closed 1T tetrahedron.



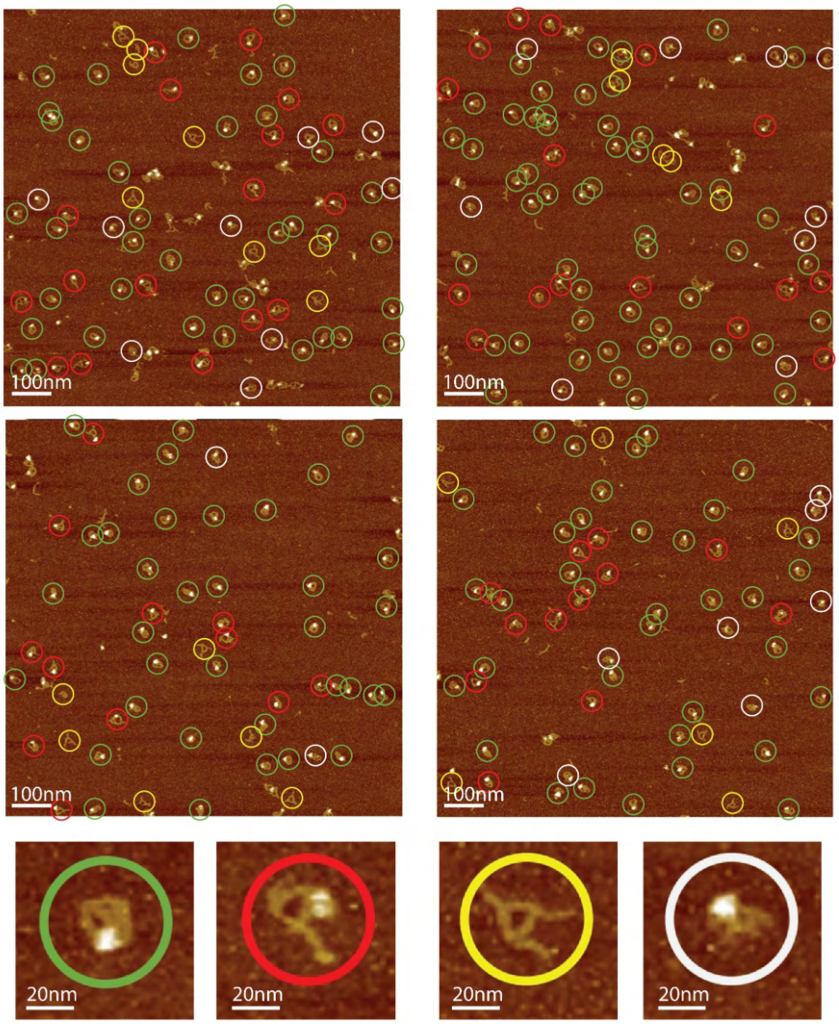
Type	No. of Structures
Green-Closed	58
Red-Half Closed	77
Yellow-Open	16
White-Ambiguous	19
Total	170

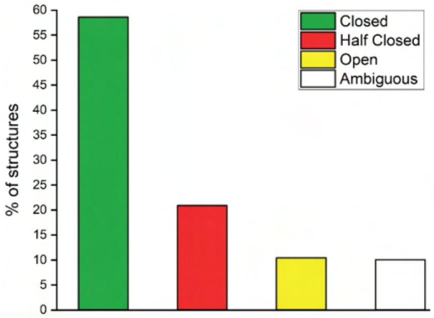
Figure 51. AFM images and yield for the PAGE-purified Closed 2T tetrahedron.



Type	No. of Structures
Green-Closed	200
Red-Half Closed	53
Yellow-Open	10
White-Ambiguous	32
Total	295

Figure 52. AFM images and yield for the PAGE-purified Closed 3T tetrahedron.





Type	No. of Structures
Green-Closed	146
Red-Half Closed	52
Yellow-Open	26
White-Ambiguous	25
Total	249

Figure 53. AFM images and yield for the PAGE-purified Closed 4T tetrahedron.

Face1-1 staple	AGGCGCGCCACCCTCAGGCGA
Face1-2 staple	GAGTTGATTCATCAGTTGAGATTTAACGCCATATCATAACCC
Face1-3 staple	TTCCAGAAAAGCCCCAAAACC
Face2-1 staple	TTCATCACAAAGTTACCAAAT
Face2-2 staple	TTTACGAGCCAGTAATAAGCAACAACGCCAACATG
Face2-3 staple	GAAAAGGTAACGAGTATAACAGTTG
Face3-1 staple	GAGACAGTGCGGAGTGTACTG
Face3-2 staple	ATTACGTGAGGATTTAGAAGTA
Face3-3 staple	TGAATTAACGTATCCAAATAAG
Face4-1 staple	TTTTAATGGAAACAAAGCATCACCTTGCTGGCAA
Face4-2 staple	TAATACCAAGCGCGAAACAAAACCGGAATCATAATTATTAA
Face4-3 staple	CAGCAGAACTGGCTCATTATACCT
Face1-1-handle	AGGCGCGCCACCCTCAGGCGA GAGCAGACCTGACGGAACTCA
Face1-2-handle	GAGTTGATTCATCAGTTGAGATTTAACGCCATATCATAACCC GAGCAGACCTGACGGAACTCA
Face1-3-handle	TTCCAGAAAAGCCCCAAAACC GAGCAGACCTGACGGAACTCA
Face2-1-handle	TTCATCACAAAGTTACCAAAT GAGCAGACCTGACGGAACTCA
Face2-2-handle	TTTACGAGCCAGTAATAAGCAACAACGCCAACATG GAGCAGACCTGACGGAACTCA
Face2-3-handle	GAAAAGGTAACGAGTATAACAGTTG GAGCAGACCTGACGGAACTCA
Face3-1-handle	GAGACAGTGCGGAGTGTACTG GAGCAGACCTGACGGAACTCA
Face3-2-handle	ATTACGTGAGGATTTAGAAGTA GAGCAGACCTGACGGAACTCA
Face3-3-handle	TGAATTAACGTATCCAAATAAG GAGCAGACCTGACGGAACTCA
Face4-1-handle	TTTTAATGGAAACAAAGCATCACCTTGCTGGCAA GAGCAGACCTGACGGAACTCA
Face4-2-handle	TAATACCAAGCGCGAAACAAAACCGGAATCATAATTATTAA GAGCAGACCTGACGGAACTCA
Face4-3-handle	CAGCAGAACTGGCTCATTATACCT GAGCAGACCTGACGGAACTCA

ACAGAGGCTTTGAGTAAACGGG
TGCCTTGAGTAAAGGATT
GCATAACCGAGGTGGCTCCAAAAGGAGCCAGC
TAAAACACGCGTTATACGACC
TATGCGATTTTACATTGCAACAGGAAAAACGCTCATATATATCCAG
AAAATCAGGTCTAGAGGGGGT TTTTTTTTTTT TTTTGTTAAAATT
CCATATTTAGAACGCGCAATT
TTAAAACGACGAAGTATAGCCCGTT
AGAATCAAATCTTTTCATAA
AGGCGAAGAAAAATCTACTACAGGTAGAAATAGAG
GTATATTCCTCACCCTCAGAACGTAATAGCGGGGTTTTTCC
GTAACGCCAGGCGGGCCGGATAGCAAGCCCACTCA
GCCTTCGGTCGCTGAGGCCCGGTTTATCAGC
GATTGCAGACTATTCAGAAAATCCCCCTCAAATGCGCTCCAATACTGCG
CTGACATTCTGGTCACACGACCAGTAATTTT
AAACACCGGATATTCATTAGAGTAACAAAGCG
CGACAAAAGGTAAAACCAAGATTACCGCGCCC
ATAAGAATAAACGTACAACGGAGATTTGTATCATCGACCGTTT
CTCAACCCTCAACCAGGCAGACTCCTCAA
AATAGAATTAATAACCCAGCGCCAAAAACGCAA
ACAAAGCT TTTTTTTTTTT CCGAACTGACAGACCAGGCGC
AAACCAATTTTAGTCTATATGTAAATGCTTA
CAAAATAAAAGAGACAAAAGGGCGATAATATCAAA
CGCAAGAC TTTTTTTTTTT AGACGCTGAGGTCTGAGAGAC
AATAGCACCGCTTCTGGAGCA
TTAGACTTAGTTACAAAATCGCGCATTGCTTCATTTGA
GATGTGAGAATAGAAAGAAAAAAGAATTTCTTAAA
AAAGAAGAGCGGGAACAGGAAGGGTCACGCTGCGCGTAAC
TCAACAGCCAACCTAAAACGAAAGGTACCTTTACTATACGTAATG
TCTAAAATATCTTTAGGAAGA
CGATTGGCCATTTTTTGTCAA
GCCGAATTGCGAATGAATCGGCCAACCAGGGTTGG
CTCGAATTGCAGCTTGCGTACGGAATTATCATCAATAAGTTT
AGCTAATGAGAATAAGAACAAGCAAGGCCTG
CGGAGCTAACTCACATTGAAGCATTCATGGTC
ATAAAAACCAAAATAGCGCAACACAAAGGAATTAC
AAAAATATTTTTTAAACAGGAAGATTGATCATATGTACCCCGGA
TAATTGTATAATTCTGCGGGC
AATGTTTATTTTGCCTTACCCTTCAAAAAGATTAAGA
GAACAAGAAAAATCTTTCCGAGAAAAAATCCAAT
CCAAATATTGACGGAAAAGCCACCTATTAGC
ATTACGTGACATCAAGAAAACATTTTCCTTCTGTAAAT
ATATATCAATAACTCATCGTAAAGTACCGACAATAAACAATCAACAATA

TGGCTATGAGAGCCAGCAAAC
CGATTAACGTCAGATGAAAAATAATTACAGAGAG
AGGGAGGCCTCAGAGCGTAATT
CCGGATGTGCTGCAAGGCCCAGCTGAGCCACCACC
TTACCTGAGCAAAAGATTAGAGCCGTCAATGCACT
AGAGAAGGCAAAAGAATGTTACTTTGTCGAAATCCGCG
CCTACCATATCAAAAAAT
TCGTTTACATAATCAGTGAGGCCACCTGAGAACTCAAAATTACCGC
TTATATATAAATAAGGCGCTA
CCGAGGTTTAAGAAGAGAGACTGAACACCC
TTTCCATTAAAACCCCGATTTAGATT
TTTTGAACCTCCCGACTAGTTGCTATTTTGCACCC
TCCATGCGGAACTTTTCACTGCATTTTGCGCTCAC
CAGTATTATATTAATTAAAAA
TTTGCAACAGCTGATTGTTTGATGGTGGTTCCGATTT
GGCGCAAAGGCCCGTGGGGTCTGGCCTTCCTGAAA
TGTCCATCACGCAAATTAACCACTAATGCAGATACATAGGAATACCACA
TACGAGCACCAGTATAAATCG
GCAGCACCGTAA TTTTTTTTTTT GCGCCGACA
GTTAAAGGCCGATTCCATAGATTTAGTTTGACTGG
TTTTCCAGAGGCATTTTGTAT
ATTTACCGTTCCAGTATGATTGTCACCGTAATAGGAACCCAT
TGCCTTCCACAGTGTGAAATTGTTATCTT
TTCAGTTGTAGCAATACTTCTCTACAGGGCTCGTC
CGTACCCTAAAGGTGCCGTAAAGCTACGTGAACCGTCT
GGAGAGAGGGCATGAAAGTATTAAAATGCCCTACA
ACCAGCAAATCGGAGGCGAGATGCCCGAA
TCGCAAATGGTCA TTTTTTTTTTT TTTGCGGATGGCCTCAACATGTT
ATACAATAGTGAGAATAACCTTGCCTTAGAA
AATAACATA TTTTTTTTTTT AGCTACAATTTTTTCCAGAGC
AGTCAGAGGCGCCACCCTAGA
CAGCAACCATTAAAGTTCGTCACCCCATTCGCCAT
ATTCCTTGCAGGGTTGATAATCACATTACTAATAG
GTTACGCATTAGACGGGAGCAGCCTACAGCCATA
TTTTGAGGGGACGGGCTGCGCAACT
GAGCGCCATTCAAAAAGGTGGCAACATCGTAGAAGAAGGAAA
TTTTGCAAACTCCAACCTTTTGATAAGAGGTCATTTT
GGTTTCATCGTAGGAATCTACCGCATCGG
TCAAAAAATCCTGCCCTTGTTTCCTCAACATACGA
GGTGTACCAACTTGTCAATCATAAGGGAATTTT
ATAGGCTGGCTGACCTTCATCAACCCAAAAGGCTTG
GGATTTGCTACGTTGGTACTCCAGCCAGCTT

AACAATCTATCGGCGTTGATTAGTAATGGAT
CATGCTGCCGTCGGTTTAGCATCAATATAATCCAGCGTCACCTGCCTA
CCTGGTTGAGGCAGGTCCAGAACCCCGC
TGGCTTTTGAGGCAATTTACCGCCTTTTCAGTCTT
GTCAGGACGTTGGGAAGTCTT
TTGCCATGTCATATAAGCTGTTAAATCAGCTCAAT
TAGGGGATCGTCACCCTAGTACGGTACATT
TTCGTATAACGTGCAGGAGGCCG TTTTTTTTTTT GGGAAACCT
CGCTATTACGGATTAAGTCGACTCTAGAGGAGCTC
TACCATACAAACAATTCTTGG
TAGTTGACCGTACGCCATC
TCCATGACTAAACCCAGCGATTATTCGAGCTTC
GGAAGCCCGAAAGACTTCAACCAGACCGGAA
TTTGCTCATTCAGACCATAAATCATT
ATCAGGGAATCCTTAATCAATCAATATCTTGAGGA
TTTGCTAGGGCGCTGGCAAGTGTAGCGTAAGAACCCTT
CTCGAAGGTAGCAAAATCACCAGTAGCACATCCGATTG
GAACTCTGAGTAGAAGAAGTG
AAGCCTGGGGTGGGTACCGAG
CGCATTTAGCCAGACCCGTCGGATTCTTATCAGGAAAC
GTAAAAATTGCGTAGATTTATCCCACAAAAATGAA
TTTGGTCATAGTAGCGCCAAGGCCGGAAACGAGAG
ACCATCGATAGACTTGATTAAAGGTGAATTATCTTTT
TTAATTACCAACAGTTGCGGT
TTATTTTCAGGTAGCCCTTAAACGCAAGTATGTTAG
ATATTTTCATTCATTAGATGTCTGGAAGTTTCCTT
CTGTAATATCCCATCCTCTGTTTACATGTTC
TTAATTGATTGCTCAGGTCAGGCAGACGTGAAAGAGGCCTGAT
TTGAACGGGTATTAAGTAA
AGACACATTACGCATAATAACGGAATAAGCTATCTGA
TAACAGTGTTGTTGAAGGAGTTGGGCGCGCG
CACAACGCTTTCCAGTCATTAAAGGGATTTTAGGCTAAACTTTCCTC
AAATTGAGCCGGAACGAGGCGATT
ACTTCTGAATAATGGAAGGTGGAAAGCCGCCGCCA
TCCGGGAAATGGGATAGGTCAAACAACTGACA
ATAACCTGTTTACTGAGAGAGTAACACTTTCATCAACATTAA
GAATACGTGGCACAGACAAGAACTGAACGAACC
TTTTTGGGGTCGAGGGAGCATACCGATAGCCC
ACATGGTTTGAAATACCGACCGTGTGAATAATTTC
AGAGGGAAATACCTACTTACATT
AAGGGGCCTAATGAGTGGGAG
TCGCAACAAACGGCGGATAGCATTGAGATCTACGAGCT

AATCAAGT TTTTTTTTTTT AATCGGCAAAATAGAACGTGGTTTT
TAAAACATCG TTTTTTTTTTT AAAAGGGACCTGAAAGCGTAATTT
ATCTTACTCAAGATTTGCGGGAAAC
CATCGGAACGAGGGTAGCGCTGTAGTTAGAGC
AATTTTTTCACGCCGATAGTT TTTTTTTTTTT TCAGTAGCGAC
TAATACATTATGGCCCACACA
TAAATTTACTGCTCCATACAC
ATAGAAGAGTGCGATAGCTTAGATTATT
ACTCCAACGTCAAAGGTAATTTTAAGCCGGAAAGGAGCGGGC
CGTTATGCGAAAAACCATCACCCA
TACCTTTTTAACCTCCGGCTGATGCACTTTTTCAA
ATCACTTGCAACGGAACATAA
TTCAGACGTTAGTATAAAGGAATTGCGAATAAT TTTTTTTTTTT ACCGTCACC
AGGTTTAGAAGATAAGTCTTTA
AATCCCAAAAGAACT TTTTTTTTTTT GTCCAGACGA
ACATACCATTACCATTAGGTT
GAACATATGGTTTACCACAAGAATTTAC
TATATTTTGACGCTCAATCGAGATGGTTTAATTTC
AGTGAACCGCGATTATCAGATGATTGATACACCGT
AATTAACGCTAACGAAACAATGAACGCGATAGAAGTTATCAAAATC
AAAGAAGGACTGGATAGCGATAGTAAGAGAG
TTTTAAAACAGGGAAGAGCCCAAGAAAATTTAAGTTTATTTTGTCA
GGACATTAAAGCCAGAAGTTAGAAGTTTGCCGTTTGCCTGAA
TTAAATATGCAACTAACAGCAGCGAA
AAGAGAATCGATGAAAGACAG
TTCT TTTTTTTTTTT GGCATGATTAAGACTCCTTCACG
CATATTCATCTTTGACCGACT
TCAACGACAGGTGCATCTGCCAGTTT
GCTTGACGGGGAAAAAGTTTG
GTCGTGCCAGCCAGTGAGACGG
ATGCGCTATTTTTAAGGGAAGAAAGCGCGAA
AAACAGACGTTGATATTCACAAACAAATATTA
GAAACAATGAAAGTAACAGTA
ATAAACAGTTGAGGCTGGGATAAGCTGCAGGTTGG
GAATAGGTGTATTTGGATTAT
AGGATGCGGTCCACTGGGGGTCAG
CGTCGCACACCGCCTGCAACAGTGCCACGCTTAAACAGA
AAAAATAAAGGCAGATAGCCGAGGCATTTTCGCACGTA
GACAATCATTGTGAATTACCA
CAAATATTTAACAATTTTGAA
TAGGGTGCTGGTTGAGAGAGTTCGTAAAAAGTGTA
GTTGGGAAGGGCGATCGGTGGTTTTCCCAAGCTTG

TTTTTCAGACGACGATAAAAC
ATTGGGCTTGTCTGAAATAAC
GAAAAACCGTTTTTATTTGGG
CTCAAATATCAAACCCGGGAA
CGTTGT TTTTTTTTTTT GTACCGTAACACTGAGTTTTTGTCGTCTTTC
GGCAGATAATGCGCCCTTGCTCGGTACGCCAGAATCCGAGTA
CTAATTTGCCAGTTACATATACATAGC
ATCAATTCTCAACAGTTTCTT
TTCACCACACCCGCATGGTTGCTTTGACGAGCA TTTTTTTTTTT CGCT
TTTTAAATATTTAAATTGAATCGTATCATTGCGCT
TTGCTTTCGATATACTCATAGCGCCTGTTGCCGGA
TCCTTGAAGGTTTTATTCTAAAACGGATTCGCCTGAGAGGCG
ATGAAAAATCTAGTACATAAATCAATATATGTGAGTATGCTTATCCGCTT
ATCTTCTAAATTCTTATGTAG
GGTGAGGAAAGGAATGGTCAGGACAACTCGT
AACCAGGAGATCGCGTAGATGGGCGCATTTTCTGTAACGATCCGCCCAC
AACTTTAAGAACCAGAACGAGTAGACATTATGTTA
CCTTTTACATCGGGAGAGCGTCTATCCTGA
ATATTCATTGAACGAGAATGTGAATATCAACGTA
ATGACAACTTGATATTGAAAATCTC
TTTTTCAAAATCACCGCTCAGAGCCGCCACCAGAATT
ATTATTCTGAAATTGATATGCCAGTGCCAGTCACGA
AATAGCAAGCAAATCAGATAGGCGTTTTAGC
CCACTACGAAGGCATAGGGCTTAATTGAGAAGCCAACGC
TTTAAAGAACGCTTATCATTCCAA
GGACAAAGCGAGTACAAACTACAATTAGCGTATGG
GATAAGTCCT TTTTTTTTTTT CAATCAATATAATAAGAGCAATT
GTTAGAAGCGTACTCGCGCTTTCACCAGCCAAC
CCCTGACAACAGTTATAGTCATTTTGCA
AACAACTAATAGAAGATGATGAAACAAAAGCCTTAAACATTCATTTCAA
CAAGAACAACAATGAATCGTAACCTATCGGCCTCA
TAAAGAATATAATAACGGCTAAAGCGAAATATCGAGATGAAC
ATAGCTCACCGCCGGTTTTTCAAAGAAACCACCACCAGTTTGCCCGAGA
GAATTTCACCAATTTAGCGTCAGACTGCCC
CGGTTAAACGTTAATA TTTTTTTTTTT AATAGTAA
AGTAACATTATCATTTCTATTAACCCTTATAAA
TTTCGGAACCT TTTTTTTTTTT CCACCACCAGAGCGCAGTCTCTGA
CCTACCGGAAGCCACCCTCAGAGCGACA
CCCTTGCCCCAGCAGGCGAAGAATAGGAACAAGAG
GAATCGTCATAA TTTTTTTTTTT ATGTGAGCGTCTGGAGCAAAC
TAATTTAGGATGAGGAAGTT
GCATTCACCACCGAACCAGTTATTCAGCCATTTGG

Figure 53. Staple and Handle Sequences

APPENDIX B

SUPPLEMENTARY INFO FOR CHAPTER 4

B.1 Experiments

B.1.1 Gel Electrophoresis

The formation of the Thrombin/DNA complex is sensitive to temperature as well as the presence of K^+. Therefore, each 5% native gel incorporated K^+ into its matrix and was run at 15C for 90 minutes at 200V. Running buffer was 10mM K^+ 7mM Mg^{2+} 1x TAE at 8 pH. Each gel was stained with SYBR gold prior to imaging at 300nm. If using a fluorophore labeled strand, the gel was was not stained prior to imaging.

B.1.2 DNA Sample Preparation

Each RBM-generated 20nt sequence was supplemented with two complementary 18nt regions to form a stem loop structure (56nt total). The RBM-generated stem loops were designed and their secondary structure predicted (see Figure 54) using NUPACK's webserver [264]. NUPACK results showed no other complex formation except for the desired stem loop. The sequences were ordered, HPLC purified from IDT and re-suspended in 10mM K^+ 7mM Mg^{2+} 1x TAE. The stem loops were annealed for 12hrs to ensure proper secondary structure formation, and their concentrations standardized to 500nM by measure of the 260nm absorbance using a Nanodrop Spectrometer. All DCA-generated sequences were originally designed to form the nanotile from the SELEX experiment which generated our dataset [210]. Using each loop individually resulted in a 15nt stem loop with non-pairing regions. These sequences were ordered in a plate from IDT with their standard desalting. Each DCA-generated sequence was purified by using a 5% or 6% denaturing gel (depending on the sequence size) in 1x TBE buffer, cutting the resulting band and precipitating the DNA out with ethanol. The stem loops were annealed for 12hrs to ensure proper secondary structure formation, and their concentrations standardized to 500nM by measure of the 260nm absorbance using a Nanodrop Spectrometer. All sequences used throughout the main text are shown in Table 23 except for any 5' 6FAM modifications which are marked in any figures in which they are used.

B.1.3 Control Sequence Verification

To confirm the binding band and establish the interaction between the stem loop sequences and thrombin, the control strands ThA and ThD were exposed to varying concentrations of thrombin shown in Figure 55b,c. For both ThA and ThD, the almost complete uptake of the stem-loop from the starting position (Figure 55) to the stem-loop / complex band at a ratio of 1:1.08 indicates the stem loop / complex band interaction is made up of

Sample Name	Full Reported Sequence
r1	CTCGAGAGTTGCAGAAGTAGTGATGATGTGTGGTAGGCACTTCTGCAACTCTCGAG
r2	CTCGAGAGTTGCAGAAGTAGTGTAGGTGTGGATGATGCACTTCTGCAACTCTCGAG
r3	CTCGAGAGTTGCAGAAGTTAGGTTTTGGGTAGCGTGGTACTTCTGCAACTCTCGAG
r4	CTCGAGAGTTGCAGAAGTAGGGATGATGTGTGGCAGGAACTTCTGCAACTCTCGAG
r5	CTCGAGAGTTGCAGAAGTCTAGGACGGGTAGGGCGGTGACTTCTGCAACTCTCGAG
r6	CTCGAGAGTTGCAGAAGTAGGGATGTGTGTGGTAGGCTACTTCTGCAACTCTCGAG
r7	CTCGAGAGTTGCAGAAGTAGGGATGCTGCGTGGTAGGCACTTCTGCAACTCTCGAG
r8	CTCGAGAGTTGCAGAAGTGAGGGTTGGTGTGGTTGGCAACTTCTGCAACTCTCGAG
r9	CTCGAGAGTTGCAGAAGTAGGGTTGGTGTGTGGTTGGCACTTCTGCAACTCTCGAG
r10	CTCGAGAGTTGCAGAAGTATGGTTGGTTTATGGTTGGCACTTCTGCAACTCTCGAG
r11	CTCGAGAGTTGCAGAAGTGAAGGGTGGTCAGGGTGGGAACTTCTGCAACTCTCGAG
r12	CTCGAGAGTTGCAGAAGTGGAGGGTGGGTCGGGTGGGAACTTCTGCAACTCTCGAG
r13	CTCGAGAGTTGCAGAAGTGGGGTTGGTACAGGGTTGGCACTTCTGCAACTCTCGAG
r14	CTCGAGAGTTGCAGAAGTAGATGGGCAGGTTGGTGCGGACTTCTGCAACTCTCGAG
r15	CTCGAGAGTTGCAGAAGTAGATGGGTGGGTAGGGTGGGACTTCTGCAACTCTCGAG
r16	CTCGAGAGTTGCAGAAGTATAGGGTGGGTGGGTGGGTAACTTCTGCAACTCTCGAG
r17	CTCGAGAGTTGCAGAAGTTGGTGGTTGGGTTGGGTTGGACTTCTGCAACTCTCGAG
r18	CTCGAGAGTTGCAGAAGTTGGGATGGGATTGGTAGGCGACTTCTGCAACTCTCGAG
r19	CTCGAGAGTTGCAGAAGTAGGGTTGGTTATGTGGTTGGACTTCTGCAACTCTCGAG
r20	CTCGAGAGTTGCAGAAGTATTGGTTGGGTAGGGTGGTTACTTCTGCAACTCTCGAG
r21	CTCGAGAGTTGCAGAAGTAAACGGTTGGTGAGGTTGGTACTTCTGCAACTCTCGAG
r22	CTCGAGAGTTGCAGAAGTCGGGGTGGTGTGGGTGGGAGACTTCTGCAACTCTCGAG
r23	CTCGAGAGTTGCAGAAGTTATTGGTTGGATAGGTTGGTACTTCTGCAACTCTCGAG
r24	CTCGAGAGTTGCAGAAGTAGGGTTGGGTGGTTGGATGAACTTCTGCAACTCTCGAG
r25	CTCGAGAGTTGCAGAAGTCGGGTTGGGGGGTTGGATTCACTTCTGCAACTCTCGAG
r26	CTCGAGAGTTGCAGAAGTCGGTTGGGGGGGTTGGATACACTTCTGCAACTCTCGAG
r27	CTCGAGAGTTGCAGAAGTTGTGGGTTGGTGAGGTAGGTACTTCTGCAACTCTCGAG
ThA	CTCGAGAGTTGCAGAAGTAGGGATGATGTGTGGTAGGCACTTCTGCAACTCTCGAG
ThD	CTCGAGAGTTGCAGAAGTGTAGGATGGGTAGGGTGGTCACTTCTGCAACTCTCGAG
p1	CTCGAGAGTTGCAGAAGTAGGGATGATGTGTGGTTGGCACTTCTGCAACTCTCGAG
p2	CTCGAGAGTTGCAGAAGTAGGGATGGTGTGTGGTAGGCACTTCTGCAACTCTCGAG
p3	CTCGAGAGTTGCAGAAGTAGGGTTGATGTGTGGTAGGCACTTCTGCAACTCTCGAG
p4	CTCGAGAGTTGCAGAAGTAGGGATGGTGTGTGGTTGGCACTTCTGCAACTCTCGAG
p5	CTCGAGAGTTGCAGAAGTAGGGTTGATGTGTGGTTGGCACTTCTGCAACTCTCGAG
p6	CTCGAGAGTTGCAGAAGTAGGGTTGGTGTGTGGTAGGCACTTCTGCAACTCTCGAG
d1	TCAGGCTCTCGAGAGTTGCAGAAGTAGGGTAGGTGTGGGGTATGCACTTCTGCCTGCATCGAGACA
d2	TCAGGCTCTCGAGAGTTGCAGAAGTAGGGTAGATGTGTAGGATGCACTTCTGCCTGCATCGAGACA
d3	TCAGGCTCTCGAGAGTTGCAGAAGTAGGGATGATGGTTGGTAGGCACTTCTGCCTGCATCGAGACA
d4	TCAGGCTCTCGAGAGTTGCAGAAGTAGGGATGATGTGGATTAGGCACTTCTGCCTGCATCGAGACA
d5	TCAGGCTCTCGAGAGTTGCAGAAGTAGGGTGGGAGCGGGGGACGCACTTCTGCCTGCATCGAGACA
d6	TCAGGCTCTCGAGAGTTGCAGAAGTCGGGTAGGTGTGGATTATGCACTTCTGCCTGCATCGAGACA
d7	TCAGGCTCTCGAGAGTTGCAGAAGTGTAGGACGGGTAGGGCGGTCACTTCTGCCTGCATCGAGACA
d8	TCAGGCTCTCGAGAGTTGCAGAAGTGGGGGTTGGGCGGGATGGGCACTTCTGCCTGCATCGAGACA
d9	TCAGGCTCTCGAGAGTTGCAGAAGTGCGGGTTGGGCAGGATCAGCACTTCTGCCTGCATCGAGACA
d10	TCAGGCTCTCGAGAGTTGCAG AAGTAGGGATGATGTGTGGTAGGCACTTCTGCCTGCATCGAGACA
d11	/5PHOS/CCAGTTTTTCTGGTGAGCTAGTGCAGACATGATCGTAGGATGGGTGGGGTGGGAGATCATGTAACTCCTAGCTGCCTGA
d12	/5PHOS/CCAGTTTTTCTGGTGAGCTAGTGCAGACATGATCGTAGGATGGGTAGGGTGGTAGATCATGTAACTCCTAGCTGCCTGA
d13	/5PHOS/CCAGTTTTTCTGGTGAGCTAGTGCAGACATGATCCTAGGTTGGGTAGGGTGGTGGATCATGTAACTCCTAGCTGCCTGA
d14	/5PHOS/CCAGTTTTTCTGGTGAGCTAGTGCAGACATGATCCTAGCATGGGTAGGGTGGTGGATCATGTAACTCCTAGCTGCCTGA
d15	/5PHOS/CCAGTTTTTCTGGTGAGCTAGTGCAGACATGATCGTAGCATGGGTAGGGTGGTCGATCATGTAACTCCTAGCTGCCTGA
d16	/5PHOS/CCAGTTTTTCTGGTGAGCTAGTGCAGACATGATCTTGGGTGGTGTAGGTTGGCGGATCATGTAACTCCTAGCTGCCTGA
d17	/5PHOS/CCAGTTTTTCTGGTGAGCTAGTGCAGACATGATCTTGGGTGGTGCAGGTTCGCGGATCATGTAACTCCTAGCTGCCTGA
d18	/5PHOS/CCAGTTTTTCTGGTGAGCTAGTGCAGACATGATCCTAGGATGGGTAGGGTGGTGGATCATGTAACTCCTAGCTGCCTGA

Table 23. The full sequences from all experiments carried out in this work, with their loop region underlined for easy identification. r1-27 correspond to sequences generated from sampling our RBM. All sequences with p labels (p1-p6) are along the mutation pathway from sequence ThA to r9. Sequences d1-d9 and d11-d17 are were generated from sampling from the DCA parameters. Sequences d10, d18, ThA, and ThD were used as controls throughout.

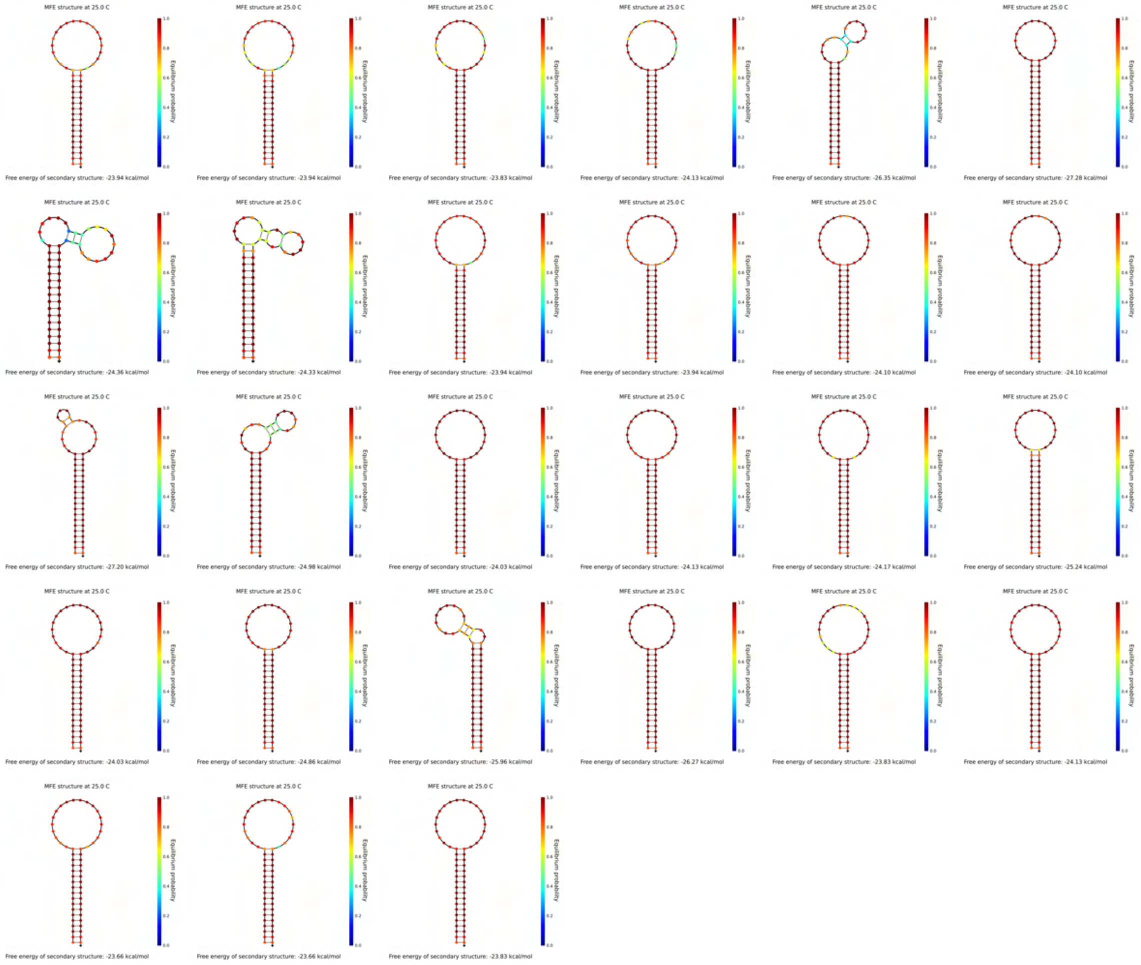

MFE structure at 25.0 C
1.0
0.8
0.6
0.4
0.2
0.0
Equilibrium probability
Free energy of secondary structure: -23.94 kcal/mol
Free energy of secondary structure: -23.94 kcal/mol
Free energy of secondary structure: -23.83 kcal/mol
Free energy of secondary structure: -24.13 kcal/mol
Free energy of secondary structure: -26.35 kcal/mol
Free energy of secondary structure: -27.28 kcal/mol
Free energy of secondary structure: -24.36 kcal/mol
Free energy of secondary structure: -24.33 kcal/mol
Free energy of secondary structure: -23.94 kcal/mol
Free energy of secondary structure: -23.94 kcal/mol
Free energy of secondary structure: -24.10 kcal/mol
Free energy of secondary structure: -24.10 kcal/mol
Free energy of secondary structure: -27.20 kcal/mol
Free energy of secondary structure: -24.98 kcal/mol
Free energy of secondary structure: -24.03 kcal/mol
Free energy of secondary structure: -24.13 kcal/mol
Free energy of secondary structure: -24.17 kcal/mol
Free energy of secondary structure: -25.24 kcal/mol
Free energy of secondary structure: -24.03 kcal/mol
Free energy of secondary structure: -24.86 kcal/mol
Free energy of secondary structure: -25.96 kcal/mol
Free energy of secondary structure: -26.27 kcal/mol
Free energy of secondary structure: -23.83 kcal/mol
Free energy of secondary structure: -24.13 kcal/mol
Free energy of secondary structure: -23.66 kcal/mol
Free energy of secondary structure: -23.66 kcal/mol
Free energy of secondary structure: -23.83 kcal/mol

Figure 54. NUPACK Predictions of the minimum free energy structure (MFE) of each DNA stem-loop at 25 ° C. Figures start from r1 in the top left corner to r27 in the bottom right corner.

a single stem loop binding to a single molecule of thrombin. Further, the combination of the two stem loops binding to thrombin at the same concentrations (Fig 55d) confirmed the cooperative binding seen in previous experiments as well as indicated a downshift of the stem-loop / protein band upon 2 stem-loops binding to thrombin.

B.1.4 Competition Assays

Competition assays were performed by mixing equimolar amounts (2.5um) of a fluorophore labeled DNA strand and non-labeled DNA strand that bind to the same Thrombin exosite. The reverse is simultaneously tested, where the fluorophore labeled version of the non-labeled strand is substituted with a fluorophore labeled version and the previously non-

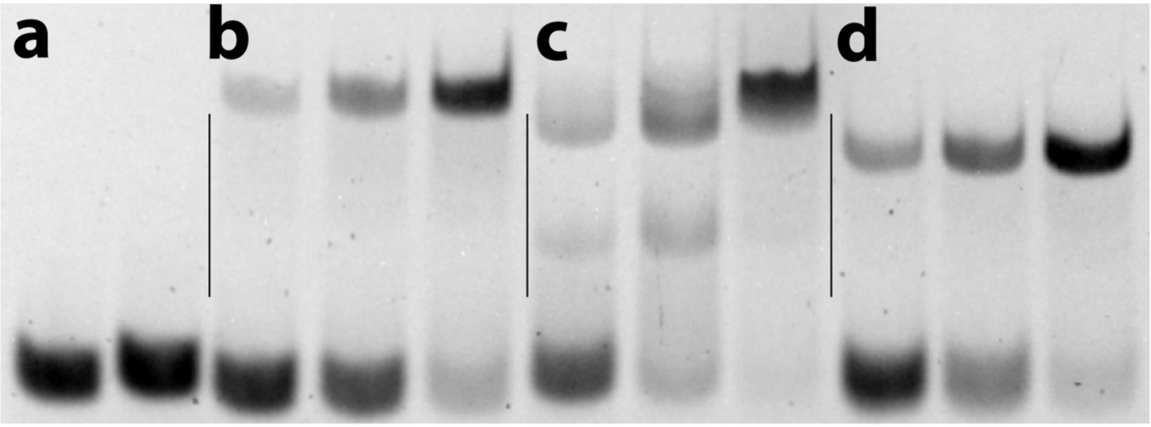


Figure 55. Panel a: lane 1 contains 5' 6FAM labeled control sequence ThA, lane 2 contains 5' 6FAM labeled control sequence ThD. Panel b: ThA mixed in varying ratios with Thrombin (1:0.32, 1:0.64, 1:1.08). Panel c: ThD mixed in varying ratios with Thrombin (1:0.32, 1:0.64, 1:1.08). Panel d: ThA + ThD mixed in varying ratios with Thrombin (1:1:0.32, 1:1:0.64, 1:1:1.08).

labeled strand is substituted for a fluorophore labeled version. In both, Thrombin is added in a 1:2 ratio (2.5um) and allowed to mix at 25°C for 30 min. Comparing the results of the assays yields a conclusive ranking of the relative binding affinity of the two sequences. Competition assays using 5' 6FAM modified sequences are depicted in Fig. 56.

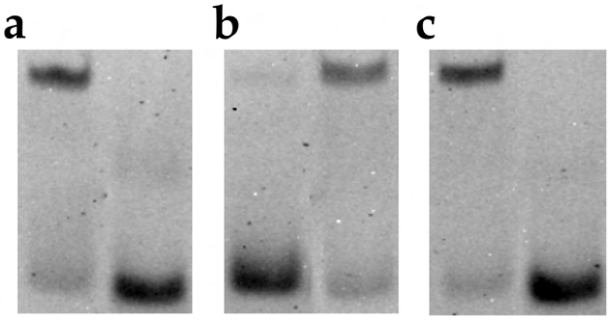


Figure 56. Competition assay of r8F vs r14 and r14F vs r8 (panel a). r8F vs r19 and r19F vs r8 (panel b), and r19F vs r14 and r14F vs r19 (panel c). The F suffix indicates the strand is fluorohore labeled with a 5' 6FAM modification.

Additionally, one-sided competition assays for all exosite-I binders and all sequences between r9 and ThA were tested against fluorophore-labeled versions of the best binding aptamers from the previous study (ThA and ThD) to assess whether any novel binder performed better. From Fig. 57b,c we see that no exosite-II binding aptamer was found which bound better than ThA and no exosite-I binding aptamer was found which bound better than ThD. Additionally, the tested exosite-I binders Fig. 57(a) are worse binders than r8 and r19 but better binders than r14.

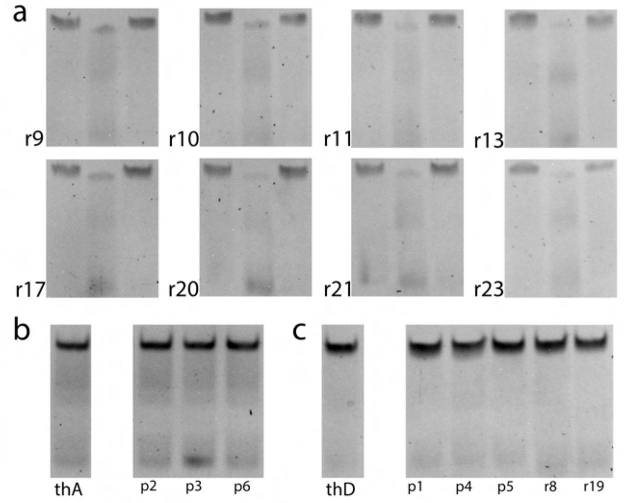


Figure 57. One sided competition assays of all exosite-I binders vs. a different fluorophore labeled strand in each well, r8, r14, and r19 respectively (panel a). Numbers to the left of each trial indicate the identity of the non-labeled strand. Additionally exosite-II binders were tested against fluorophore labeled ThA with negative control labeled ThA (panel b) and select exosite-I binders were tested against ThD with negative control labeled ThD (panel c).

B.1.5 Thrombin Sample Preparation

We used 1 mg Human α-thrombin manufactured by Haematologic Technologies Inc. and purchased from Fisher Scientific Co. Concentrations were assessed by 280nm absorbance using a Nanodrop Spectrometer. The stock was stored at -20C. Sample concentrations were made at 500nm and 250nm in 1x PBS 10mM K$^+$ 7mM Mg^{2+}. Each sample was made fresh prior to being used in an assay.

B.2 Inference of sequencing error probability

We describe here our method for inferring the single-site sequencing error probability. The analysis here discussed is based on sequences from the left loop, collected at the last selection round (round 8). Repeating the analysis on the right loop provides analogous results.

Given a sequence σ with high copy number $n_\sigma \gg 1$, the method uses as signal the number μ_σ of sequences that are at Hamming distance 1 from σ and are never observed in the dataset. This number depends on the error rate, since a higher error rate is expected to cause more of these sequences to be detected. Since we consider only the left loop, sequences have a length $L = 20$ nt.

In Fig. 58A we provide a representation of the sequence space around σ = GGGTGATGTGTGGTAGGC , which is the sequences with highest copy number $n_\sigma = 8034$ in our dataset. The dots in a circle around σ represent the $3 \times L = 60$ sequences that belong to the neighborhood of the main sequence $\mathcal{N}(\sigma)$, with color encoding their copy number. Some of these sequences are present > 100 times, and are unlikely to be an artifact of sequencing error. Other are present 1-2 times and can potentially be generated by sequencing errors. Finally, a number $\mu_\sigma = 12$ of sequences are absent in the sample (red crosses). These are mostly related to mutations removing one Guanine from the sequence, which might be related to a loss of fitness. While it is not possible to know with certainty whether one of the present neighbouring sequences with low copy-number was originated by sequencing error, the fact that some of these sequences are absent implies that σ was never mis-read into these sequences. This information will be used in our inference. We start by selecting a number of sequences with high copy number. In fig. 58B we plot the number of sequences that have copy-number higher than a given threshold, as a function of the threshold. For our analysis we select as "peaks" all sequences with $n_\sigma > 1000$ (21 such sequences in the dataset). In Fig. 58C we report the Hamming distance matrix for the selected sequences. As can be expected peaks tend to cluster together, with most of the peaks having at least one other peak in their neighbourhood. This can potentially increase the bias in our upper bound for the sequencing error probability. We will later introduce a correction to reduce this bias.

As a next step we define a probability for μ_σ as a function of the reading error probability. We call ϵ the probability of mis-reading a single nucleotide in the sequence. We consider this probability to be uniform along the sequence and on the real/read nucleotides, so that the probability of obtaining as outcome of sequencing σ', one of the single-site mutations $\mathcal{N}(\sigma)$ of σ, when in reality reading σ is:

$$P(\sigma'|\sigma) = p(\epsilon) = \frac{\epsilon}{3}(1-\epsilon)^{L-1}. \tag{B.1}$$

The real copy-number $\widetilde{n}_\sigma$ of σ in the sample might be slightly different from the observed copy number n_σ, due to sequencing error. If we call $P(\sigma|\sigma) = (1-\epsilon)^L$ the probability of correctly reading σ, then for a small enough error, we can approximate

$$n_\sigma \simeq \widetilde{n}_\sigma P(\sigma|\sigma) + p(\epsilon) \sum_{\sigma' \in \mathcal{N}(\sigma)} \widetilde{n}_{\sigma'} \simeq \widetilde{n}_\sigma P(\sigma|\sigma). \tag{B.2}$$

For any given sequence $\sigma' \in \mathcal{N}(\sigma)$, the probability of never mis-reading σ' when in reality sequencing σ is given by:

$$P(n_{\sigma'} = 0) = (1 - p(\epsilon))^{\widetilde{n}_\sigma} = q(\epsilon, n_\sigma). \tag{B.3}$$

Finally, the probability that in the neighbourhood of σ a number μ_σ of sequences are never observed, provided that in reality they were never present, is:

$$P(\mu_\sigma|n_\sigma,\epsilon) = \mathrm{Binom}\big[|\mathcal{N}(\sigma)|, q(\epsilon,n_\sigma)\big](\mu_\sigma) = \binom{|\mathcal{N}(\sigma)|}{\mu_\sigma}\big(q(\epsilon,n_\sigma)\big)^{\mu_\sigma}\big(1-q(\epsilon,n_\sigma)\big)^{|\mathcal{N}(\sigma)|-\mu_\sigma}, \tag{B.4}$$

where $|\mathcal{N}(\sigma)| = 60$ is the size of the neighbourhood of σ. When writing this equation we are making a number of simplifications. On one hand we are considering that all sequences in $\mathcal{N}(\sigma)$ were originally absent in the sample. Moreover we are neglecting the probability that reads of these sequences might be generated from the sequencing of other sequences different from σ (e.g. other peaks). All of these effects will bias our estimate, but the bias is always in the same direction, leading us to overestimate ϵ. For this reason the result of the inference represents a reliable upper bound.

To reduce the bias we can remove from the total number of trials in the binomial the number of sequences that we are confident to be really present in the original sample. As a simple correction, we substitute the term $|\mathcal{N}(\sigma)| = 60$ in eq. (B.4) with $|\{\sigma' \in \mathcal{N}(\sigma) \text{ s.t. } n(\sigma') \leq 10\}|$, i.e. the number of sequences in the neighbourhood with no more than 10 counts. That is to say we consider all sequences with more than 10 counts to be really present in the original sample. We perform the inference both with and without this correction (cf. fig. 58D). At this point we can write the total log-likelihood of our data as a function of the error probability ϵ as:

$$\log\mathcal{L}(\mathrm{data}|\epsilon) = \sum_{\sigma\in\mathrm{peaks}} \log P(\mu_\sigma|n_\sigma,\epsilon) \;\propto\; \log\mathcal{L}(\epsilon|\mathrm{data}), \tag{B.5}$$

where the inversion was operated using Bayes theorem with an uniform prior for ϵ. In fig. 58D we display the behavior of the log-likelihood as a function of ϵ for the two cases, with and without correction. Numerical maximization of these functions yields values of $\epsilon^* \sim 10^{-3}$ as an upper bound for the error probability. To obtain a confidence interval on these bounds one can perform a Gaussian fit on the likelihood (i.e. a quadratic fit of the log-likelihood) around its maximum, and use the variance of the inferred Gaussian to obtain a confidence interval for the inferred value. In this case we obtain a standard deviation of the order of 4×10^{-5}. In conclusion, we are confident that the single-site sequencing error probability in our dataset is smaller than 10^{-3}.

B.2.1 Estimation of number of sequencing error artifacts in the dataset

We can make use of the previously derived upper bound for ϵ to provide an upper bound for the number of unique sequences in our dataset that could be generated by sequencing error.

Since we expect double errors to be sufficiently rare in our dataset (for $\epsilon^* \sim 10^{-3}$ the probability of having more than 1 error is $\sim 2 \times 10^{-4}$), we can consider that in order to be an error, all the reads of a sequence σ in our dataset must be generated by sequences in its neighbourhood $\mathcal{N}(\sigma)$, with the probability of mis-reading being equal to $p(\epsilon)$ (see Eq. (B.1)).

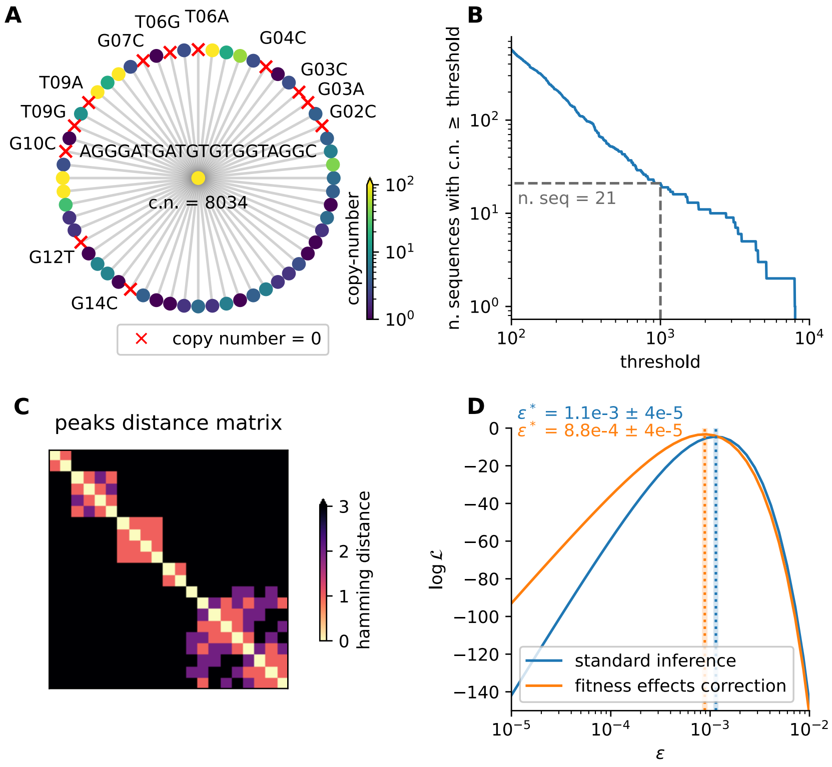


Figure 58. Inference of an upper bound for the sequencing error probability in our sample, using sequences from the left loop in the 8th round. Panel A: example of sequence space around the most abundant sequence in the dataset. The main sequence is represented as a dot in the center, and the full DNA sequence and copy number (c.n.) are reported. Dots around it represent sequences at hamming distance 1, with color encoding their copy number. Sequences that were never detected in the sample are indicated with red crosses. For these sequences we report the difference from the main sequence as a triplet (original nucleotide, position, substituted nucleotide). Notice how some of the neighboring sequences have high copy number, indicating probable fitness effects. Most of the non-detected sequences are associated with removal of a guanine, which might decrease binding affinity. Panel B: number of sequences with copy-number greater than a given threshold. For our analysis we select only sequences with c.n. ≥ 1000 (21 such sequences in the dataset). These sequences are referred to as “peaks” in the analysis. Panel C: relative Hamming distance between peak sequences. High-copy-number sequences tend to cluster together. This can cause a less precise estimation of the inferred sequencing error upper bound, since the neighbourhood of a peak can be populated by other high-fitness sequences. To correct for this we introduce a correction that removes sequences with c.n. > 10 from the expression of the likelihood. Panel D: log-likelihood of the single-site sequencing error probability ϵ. The inference was performed in two ways: either using the standard approach (blue) or introducing the correction for fitness effects (orange). In each case we mark the inferred value ϵ^* with vertical dotted lines. The thin shaded area represent the confidence interval, that was derived through a Gaussian fit of the log-likelihood in proximity of its maximum.

For each sequence we define:

$$N_\sigma = \sum_{\sigma' \in \mathcal{N}(\sigma)} n_{\sigma'}. \tag{B.6}$$

This is the total number of sequences in the neighborhood of σ. Because of sequencing error the real number might be slightly higher, and as done for n_σ one can introduce the correction $\widetilde{N}_\sigma = N_\sigma/(1-\epsilon)^L$. We can take as an upper bound for the probability of σ to be an artifact of sequencing error, the probability that by reading $\widetilde{N}_\sigma$ sequences in the neighbourhood of σ, we read σ a number of time equal or greater than the observed copy-number n_σ:

$$P(n_{\text{err}} \geq n_\sigma) = \pi(\sigma, \epsilon) = \sum_{k=n_\sigma}^{\infty} \text{Binom}[N_\sigma, p(\epsilon)](k) \tag{B.7}$$

We numerically evaluate this probability for every sequence σ. The value of N_σ is efficiently computed by generating all possible single mutations $\sigma' \in \mathcal{N}(\sigma)$, and quickly recovering their copy-number using a hash table.

In Fig. 59 we report the distribution of $\pi(\sigma, \epsilon^* = 10^3)$ for all of the sequences in our dataset. For the great majority of the sequences this probability is very low. From the procedure we employ it follows that sequences with the highest probability of being errors are ones that have very low n_σ and with a highly populated neighbourhood (high N_σ). By treating the reality of each unique sequence as a Bernoulli random variable, the mean and variance for the total number of unique sequences that we expect to be an artifact of sequencing error can be expressed as:

$$E[N_{err}] = \sum_\sigma \pi(\sigma, \epsilon^*) \quad Var[N_{err}] = \sum_\sigma \pi(\sigma, \epsilon^*)(1 - \pi(\sigma, \epsilon^*)) \tag{B.8}$$

This gives an estimate $N_{err} \sim 941 \pm 28$. Since our dataset is composed of roughly 2×10^5 unique sequences this upper bound represents only 0.5% of the total dataset, and it is not expect to meaningfully impact the training of our models.

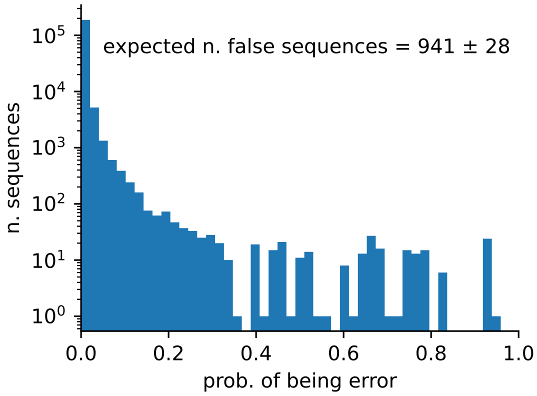


Figure 59. Distribution of inferred single-sequence error probabilities. For each sequence in the considered dataset (round 8, left loop) we infer the probability of being an artifact of sequencing error, using the approach described in Methods B.2. In the inference the single-site error probability was set equal to the upper bound $\epsilon^* = 10^{-3}$. The vast majority of sequence have a zero or low probability of being sequencing error artifacts. From this distribution one can evaluate the mean and standard deviation of the total number of artifacts. This gives an upper bound of $N_{err} = 941 \pm 28$, which corresponds to a 0.5% of the total number of unique sequences in the dataset considered.

B.3 Details of RBMs' training

We trained several RBMs which have been used for the analysis presented in this manuscript. For the training of each RBM, we used 90% of the dataset as training set and 10% of the dataset as validation set to check that no overfitting is observed. For RBMs trained using information on counts, the training dataset is obtained by sampling from the dataset of unique sequences many sequences (below the exact number for each RBM is given), with a probability proportional to each sequence's count (this gives the same results as long as the size of the re-sampled dataset is large enough, and allows to avoid having too large dataset which considerably slow down the RBM training). The training set was divided in mini-batches and for each epoch each mini-batch was used to perform an update of the parameters, using the persistent contrastive divergence algorithm with few (below the exact numbers are given) number of Monte-Carlo steps for each update of the paramters. In all cases the training stopped after 20000 updates of the RBM parameters. Finally, we used a L_1^2 regularization of the form given in Eq. (4.6) to increase sparsity in the weights, which in turn improves the interpretability of the contribution of each hidden unit to a sequence's log-likelihood. Below we give the regularization parameter λ used for each RBM trained (see Eq. (4.6)).

For the full range of explored hyperparameters (size of mini-batches, number of Monte-Carlo steps, regularization strength), we never saw any sign of relevant overfitting, and we motivated this with the very large datasets that are available for training the models.

The code used to train the RBMs can be obtained from https://github.com/jertubiana/PGM. We now give more details about the training of each RBM model used in this manuscript. To distinguish RBMs trained with sequences observed in different rounds, we will append the round number to the model name. In particular, we used the following RBMs in this manuscript:

- RBM-DC6 (Fig. 20, Suppl. Figs. 72, 80), trained on the double aptamers (40 nucleotides) obtained from the SELEX 6th round. The training set is built by re-sampling 736436 sequences from the dataset of unique double-loop sequences observed in round 6, using their number of counts as weight for the sampling. The parameters are: 40 visible units, 90 hidden units, $\lambda = 0.01$, 10 Monte-Carlo steps for each update of the parameters, mini-batches of size 500.
- RBM-DC8 (Figs. 21, 22, Suppl. Figs. 71, 73), trained on the double aptamers (40 nucleotides) obtained from the SELEX 8th round. The training set is built by re-sampling 719413 sequences from the dataset of unique double-loop sequences observed in round 8, using their number of counts as weight for the sampling. The parameters are: 40 visible units, 90 hidden units, $\lambda = 0.01$, 10 Monte-Carlo steps for each update of the parameters, mini-batches of size 500.
- RBM-SC8 (Figs. 21, 22, 26, Table 4, Suppl. Figs. 69, 70, 76, 78, 79), trained on the single aptamers (20 nucleotides) obtained from the SELEX 8th round. The training set is built by re-sampling 725431 sequences from the dataset of unique single-loop left or right sequences observed in round 8, using their number of counts as weight

for the sampling. The parameters are: 20 visible units, 80 hidden units, $\lambda = 0.01$, 2 monte-carlo steps for each update of the parameters, mini-batches of size 1000.

- RBM-SU8 (Figs. 23, 26, Table 4, Suppl. Figs. 67, 73, 74, 76, 78), trained on the single aptamers (20 nucleotides) obtained from the SELEX 8-th round, merging sequences from the left and right loops (unique single-loop sequences: 382094; with counts: 1450862). Multiple copies of the same aptamer are neglected. The parameters are: 20 visible units, 70 hidden units, $\lambda = 0.01$, 4 Monte-Carlo steps for each update of the parameters, mini-batches of size 500.
- RBM-SC5 (Suppl. Figs. 77), trained on the single aptamers (20 nucleotides) obtained from the SELEX 5th round. The training set is built by re-sampling 1375403 sequences from the dataset of unique single-loop left or right sequences observed in round 5, using their number of counts as weight for the sampling. The parameters are: 20 visible units, 70 hidden units, $\lambda = 0.01$, 8 monte-carlo steps for each update of the parameters, mini-batches of size 1500.
- RBM-SC6 (Suppl. Figs. 77), trained on the single aptamers (20 nucleotides) obtained from the SELEX 6th round. The training set is built by re-sampling 598696 sequences from the dataset of unique single-loop left or right sequences observed in round 6, using their number of counts as weight for the sampling. The parameters are: 20 visible units, 80 hidden units, $\lambda = 0.01$, 8 monte-carlo steps for each update of the parameters, mini-batches of size 600.
- RBM-SC7 (Suppl. Figs. 77), trained on the single aptamers (20 nucleotides) obtained from the SELEX 7th round. The training set is built by re-sampling 419934 sequences from the dataset of unique single-loop left or right sequences observed in round 7, using their number of counts as weight for the sampling. The parameters are: 20 visible units, 70 hidden units, $\lambda = 0.01$, 8 monte-carlo steps for each update of the parameters, mini-batches of size 500.
- RBM-SU6 (Suppl. Fig. 80), trained on the single aptamers (20 nucleotides) obtained from the SELEX 6-th round, merging sequences from the left and right loops (unique single-loop sequences: 598696; with counts: 1472872). Multiple copies of the same aptamer are neglected. The parameters are: 20 visible units, 70 hidden units, $\lambda = 0.01$, 4 Monte-Carlo steps for each update of the parameters, mini-batches of size 600.
- RBM-LC8, RBM-RC8 (Suppl. Fig. 69), trained on the single aptamers (20 nucleotides) obtained from the SELEX 8-th round. The training sets of RBM-LC8 (RBM-RC8) is built by re-sampling 177014 (227789) sequences from the dataset of unique left-loop (right-loop) sequences observed in round 8, using their number of counts as weight for the sampling. The parameters of both models are: 20 visible units, 70 hidden units, $\lambda = 0.01$, 4 Monte-Carlo steps for each update of the parameters, mini-batches of size 500.
- RBM-NPU8 (Suppl. Fig. 70), trained on the single aptamers (20 nucleotides) obtained from the SELEX 8-th round, merging sequences from the left and right loops, after excluding parasite sequences. Parasite sequences are obtained here as single-loop sequences with log-likelihood computed by RBM-SU8 lower than -24.8, with the partner loop having log-likelihood computed by RBM-SU8 larger than -24.8 (procedure resulting in 276682 unique single-loop non-parasite sequences). Multiple copies of the same

aptamer are neglected. The parameters are: 20 visible units, 70 hidden units, $\lambda = 0.01$, 4 Monte-Carlo steps for each update of the parameters, mini-batches of size 500.

- RBM-NPC8 (Suppl. Fig. 70), trained on the single aptamers (20 nucleotides) obtained from the SELEX 8-th round, merging sequences from the left and right loops, after excluding parasite sequences. Parasite sequences are obtained here as single-loop sequences with log-likelihood computed by RBM-SC8 lower than -26.6, with the partner loop having log-likelihood computed by RBM-SC8 larger than -26.6 (procedure resulting in 274250 unique single-loop non-parasite sequences). The training dataset is built by sampling 246825 non-parasite sequences, using their number of counts as weight for the sampling. The parameters are: 20 visible units, 70 hidden units, $\lambda = 0.01$, 4 Monte-Carlo steps for each update of the parameters, mini-batches of size 500.

All the parameters for the training which are not given here are the default parameters as defined in the code. The trained RBMs are provided in the Github repository (https://github.com/adigioacchino/RBMsForAptamers), together with a jupyter notebook that can be used to re-train them.

As a final remark, we checked that the results obtained here depend very little on the precise values of the hyperparameters used here (see Suppl. Fig. 75). The only notable exception being the usage of counts to weight multiple occurrences of the same aptamer in the dataset. We decided to exclude multiple occurrences from the training to regularize the RBM, as discussed in in details in Suppl. Sec. 79.

B.4 DNN and Traditional Machine Learning

B.4.1 Dataset preparation

Starting with the raw SELEX data from the 8th round of selection of our previous study, we have both 20nt aptamer sequences in each arm of the DNA scaffold and a copy number, representing the number of times that sequence was observed during sequencing. Any sequence not matching the 40nt length was assumed to have a reading error and excluded from the dataset. Independent counts for each arm of the sequence were generated by counting their occurrence throughout the dataset. Using their individual counts, each 20nt sequence was categorized as either as a "good" (copy number > 10) or "bad" (copy number < 10) binder. Note that this approach is distinct from our training of the RBMs, where we considered all sequences in the training sample and used counts for weighting the sequences.

Our analysis of the dataset found a subset of bad binder sequences far in sequence space from any other observed sequence in the dataset that were paired with good binder sequences. We concluded that these sequences were most likely carried through the selection process by their good binder and subsequently excluded these sequences from our training set.

Three datasets were generated from the remaining sequences: sequences from the left loop (L), sequences from the right loop (R), and sequences from both loops (B). Each dataset consists of the entire set of good binders from the appropriate loop and 5 randomly sampled bad binders per good binder. Training sets (80% of good binders, $\sim 35k$ in total sequences for L and R, $\sim 70k$ for B) and validation sets (20% of good binders, $\sim 12k$ in total sequences for L and R, $\sim 25k$ for B) were split from our dataset. As further verification of our DNN and traditional ML models we used the experimental results from both the RBM-generated sequences as well as the DCA-model generated sequences to assess our models accuracy. All sequences were one-hot encoded prior to training, validation, or prediction.

As using only sequences from the final round of the SELEX procedure introduces a general bias of all sequences interacting with thrombin, three more datasets were created (GL, GR, and GB) with good binders selected as previously done but bad binders were randomly sampled from a set of random sequences outside the SELEX dataset's sequence space. These datasets had the same amount of sequences as those mentioned previously (L, R, B). We assume that if there is no bias in the initial random library, most of the possible aptamer sequences of length 20 were initially present, and hence a randomly generated sequence which is not encountered in the SELEX dataset is most likely not going to be able to bind to thrombin.

B.4.2 Model Selection

For the classification task we used 5 different deep learning models: 2 versions of a Variational Auto Encoder [265], 2 versions of a Resnet [266] and a Siamese Network Model [267] outlined in Suppl. Table 24. A schematic description of the DNN model specifications used in this work is provided in Table 24. Additionally we used 3 classic Machine Learning methods: a decision tree, a random forest and a gradient boosted tree classifier to also classify the sequences as binders or non binders.

B.4.3 DNN Training Specifics

All 5 models were written as pytorch lightning modules and hyperparameter optimization was done using the raytune library. Integration of each pytorch module with raytune enabled simultaneous distributed hyperparameter optimization. All models were trained for either 30 or 50 epochs. No significant performance increase or decrease was observed between models trained for 30 vs 50 epochs.

Hyperparameter optimization was performed using the raytune library. For resnet, we optimized the batch size, learning rate (lr) and dropout (dr) prior to the dense layer and softmax. For variational Auto-Encoders, we optimized the batch size, learning rate, dropout and z_dim (embedding dimension). For the siamese network, we optimized the learning rate, batch size, and distance cutoff (Euclidean distance cutoff, being less means a match while being greater indicates a nonmatch). As a grid search, the AsyncHyperBandScheduler (AHSA) was given 10 trials with the goal to find the model with best accuracy on the validation set. Bayesian Optimization was performed on the same hyperparameters as the ASHA, save the integer valued batch size. Bayesian optimization was given a different directive, to minimize the mean loss (training+validation). Population-based training was only performed on the siamese network with the goal of maximizing the accuracy on the validation set.

B.4.4 DNN Results

To compare performance of our DNN models, we assessed the accuracy of each model to predict a binder/nonbinder label for each experimentally validated dataset: the RBM generated dataset and the DCA generated dataset. We also calculated the F1 score metric by comparison of each model's prediction with the ground truth. The F1 score is the harmonic mean of precision, the number of true positives divided by the sum of true positives and false positives, and recall, the number of true positives divided by the sum of true positives and false negatives, in a binary classification task. A F1 score was calculated for each dataset and a mean F1 score was determined by weighting each individual F1 score by the number of total sequences in the dataset. Scores for the DNN models are provided in Table 25.

DNN (L, R, B) models (i.e. models trained on L, R or B dataset) failed to generalize to our experimental datasets. In every case, prediction of binding ability on the RBM and DCA datasets results in a significant number of false positives and false negatives. Bayes hyperparameter optimized models were directed to either minimize the loss on the validation dataset or maximize the accuracy on the validation dataset whereas AsyncHyperBandScheduler (AHSA) hyperparmater optimized models were directed to only maximize the accuracy

Model	**Description**
Long Resnet	A 152 layer Resnet followed by a dropout layer, a 512 input to 2 output linear layer followed by a softmax layer. Residual networks guarantee performance of subsequent layers in the network by mapping to a residual function F(x) = H(x)-x. This network architecture has been shown to avoid vanishing gradients and accuracy degradation present in traditional network architecture learning [266]. During training, this model used label smoothed [268] (smoothing=0.01) cross entropy as its loss function.
Short Resnet	An 18 layer Resnet followed by a dropout layer, a 512 input to 512 output linear layer, a DReLU activation function, a 512 input to 2 output linear layer, and finally a softmax layer. During training, this model used label smoothed (smoothing=0.01) cross entropy as its loss function.
Long Variational AutoEncoder	A 2d convolution with ReLU activation function followed by three encoder blocks encoded the embedding. Encoder blocks consisted of a spectral normalized 2d convolution layer [269], followed by 2d batch normalization and a leaky ReLU activation function. The decoder consisted of 4 decoder blocks made up of a transposed 2d convolution followed by 2d batch normalization and a leaky ReLU activation function. Self attention layers were added in between both encoder and decoder blocks [270]. Binary classification of binder vs. nonbinder was performed on each embedding by two fully connected layers (sizes 128 and 64, consisting of: a dropout layer, linear layer, 1d batch norm, and a leaky ReLU) followed by a dropout layer, linear layer (size 2), and a final softmax layer. Similarly mu and logvar were generated by two fully connected layers and a final layer (sizes 128, 112, 100). Variational AutoEncoders are generative models designed to sample across a continuous latent space[265]. During training this model used label smoothed (smoothing=0.01) cross entropy on the predictions and symmetric MSE loss on the decoder's reconstruction. The loss functions were mixed for the total training loss.
Short Variational AutoEncoder	A 152 layer Resnet encoder and 2 decoder blocks (separated by an attention layer). A 512 input to 2 output linear layer was trained on each embedding with a log softmax layer on the end to predict a binding vs. nonbinding result. During training this model used label smoothed (smoothing=0.01) cross entropy on the predictions and symmetric MSE loss on the decoder's reconstruction. The loss functions were mixed for the total training loss.
Siamese	A Siamese network trained on pairs of sequences to discriminate between binder-binder pairs and nonbinder-binder pairs. The Siamese network used here consisted of a single resnet made up of 4 layers to a 512 input to 256 output linear layer, a sigmoid activation function, and a 256 input to 2 output linear layer following. Each iteration was run individually on pairs of sequences [267]. The Euclidean distance between the resulting embeddings is used to assign our binary classification value. During training this model used contrastive loss as its loss function.

Table 24. Descriptions of all DNN models used in this work.

on the validation dataset. The most accurate (L, R, B) models on the RBM generated dataset (Bayes Resnet L and Bayes Resnet L and R) were directed to minimize the loss for hyperparameter optimization and achieved 74.1% (20/27) accuracy on the RBM experimental dataset with poor performance on the DCA generated dataset at 31.3% (5/16) accuracy. A distinct correlation between optimization directive and performance metrics was observed. Models that were optimized to minimize the loss of the validation dataset performed worse in validation set accuracy, better in RBM generated dataset binder prediction, and worse in DCA generated dataset binder prediction to a significant degree than those optimized to maximize the validation set accuracy. From the confusion matrices of loss minimized models on the RBM generated dataset (Fig. 61), we see these models are completely unable to distinguish between nonbinders and binders in both the RBM generated and DCA generated datasets.

DNN (L, R, B) models trained to maximize the accuracy on the validation set performed poorly overall. The best performing of them (ASHA VAE short R) managed the highest mean F1 score, excellent accuracy on the DCA generated dataset at 87.5% (14/16) accuracy but poor performance on the RBM generated dataset with 48.1% (13/27) accuracy. The poor performance of all DNN (L, R, B) models indicates the sequencing info of the last round of selection is not sufficient for DNN models to classify sequences on their ability to bind a target.

DNN (GL, GR, GB) models were trained as a more naive classifier using good binders and randomly generated bad binders for both training and validation. As the random bad binders were guaranteed be to outside the sequence space of the entire 8th round of selection, we would expect these models to over-predict binders in our datasets which contain binders and nonbinders separated by small distances in sequence space. Indeed, all (GL, GR, GB) models have higher accuracy values than their (L, R, B) counterparts, but consistently have little to none false negatives and a large number of false positives on the sequences generated using RBM (Fig. 62). Additionally the higher accuracy scores on the RBM generated dataset and lower accuracy scores on the DCA generated dataset is due to the difference in population group membership (binder vs. nonbinder) of the two datasets. Their ability to predict thrombin binding ability from sequences close in sequence space is subpar due to their overfitting to the aptamer sequence space.

The performance of all DNN models on predicting thrombin binding ability from sequence alone was poor. DNN (L, R, B) models tend to generate a notable amount of false positives and false negatives, while (GL, GR, GB) models generate false positives almost exclusively on the RBM generated dataset. Overall, using the last round of selection for our dataset exclusively (L, R, B) or for just the good binders (GL, GR, GB) did not allow accurate prediction of thrombin binding ability from any of the DNN models.

B.4.5 Traditional ML Training Specifics

Three traditional models: a single tree, a random forest, and a gradient-boosted forest were used to classify the experimental dataset as binders or nonbinders. The training and validation datasets used were the same as those used for the deep learning models. The scikit-learn python library implementations of each of the three models were used in this work.

B.4.6 Traditional ML Results

Our traditional ML techniques' performance was measured by the same metrics as for our DNN models, namely the accuracy on the RBM generated sequences, the accuracy on the DCA generated sequences, and the F1 mean of both datasets shown in Table 26.

Traditional (L, R, B) models very rarely predicted a nonbinder correctly in our RBM generated dataset, instead predicting almost every sequence to be a binder. Their validation set accuracy never crossed 30%. Similar to our DNN models, the accuracy on the validation sets of the (GL, GR, GB) models was significantly better than (L, R, B) models due to the difference in sequence space of the bad binders. Traditional (GL, GR, GB) models suffered from the same issue of an overabundance of false positives including the single tree models which had the best performance of any machine learning model besides the RBM. The GR single tree achieved an accuracy of 85.2% (23/27) on the RBM generated dataset and an accuracy of 81.3% on the DCA generated dataset. Despite the high accuracy, these models suffer from the same over-fitting that the DNN (GL, GR, GB) models where binders are over-predicted significantly. The small difference in single tree models GR and GB illustrate how decreasing the amount of false positives by one in the RBM generated set has the effect of predicting almost 20% less binders in the DCA generated dataset. This ability to overestimate binders is especially apparent in the confusion matrices of the random forest (GL, GR, GB) models Fig. 63. The random forest on average performed worse than the single tree, performing as well as most DNN models. This is in stark contrast to our gradient boosted classification tree which performed poorly on every dataset no matter the hyperparameters tried.

B.4.7 Additional ML Results

The main results for the DNN models and traditional ML models referenced in the main text are shown in Table 25 and Table 26 respectively. Fig. 60 shows the AUC, several binary performance metrics, and the performance diagram for the VAE Long ASHA model in (a-c) respectively, for the six training data sets. Additional ML results in the form of confusion matrices of each model's performance on the RBM-generated sequence dataset are included in Figs. 62, 61 and 63.

Model	Validation Acc.	RBM Acc.	DCA Acc.	F1 mean
AHSA Resnet Long				
L	0.792	0.333	0.750	0.428
R	0.711	0.296	0.562	0.469
B	0.751	0.444	0.625	0.578
GL	0.999	0.778	0.375	0.718
GR	0.999	0.889	0.562	0.790
GB	0.998	0.778	0.438	0.729
Bayes Resnet Long				
L*	0.304	0.741	0.312	0.697
R*	0.281	0.704	0.312	0.683
B*	0.280	0.741	0.312	0.697
AHSA Resnet Short				
L	0.758	0.333	0.688	0.463
R	0.789	0.407	0.750	0.586
B	0.767	0.407	0.875	0.617
GL	0.998	0.778	0.438	0.729
GR	0.999	0.778	0.438	0.729
GB	0.999	0.852	0.562	0.777
Bayes Resnet Short				
L*	0.384	0.741	0.312	0.697
R	0.796	0.444	0.688	0.528
B	0.758	0.333	0.625	0.470
AHSA VAE Long				
L	0.828	0.333	0.812	0.416
R	0.837	0.333	0.625	0.492
B	0.819	0.333	0.875	0.510
GL	1.000	0.815	0.562	0.766
GR	1.000	0.778	0.500	0.741
GB	1.000	0.778	0.562	0.754
Bayes VAE Long				
L	0.804	0.307	0.688	0.401
R	0.838	0.407	0.688	0.505
B	0.829	0.296	0.750	0.450
AHSA VAE Short				
L	0.757	0.296	0.688	0.365
R	0.789	0.481	0.875	0.623
B	0.776	0.407	0.812	0.574
GL	1.000	0.741	0.562	0.739
GR	1.000	0.889	0.688	0.822
GB	1.000	0.889	0.562	0.790
Bayes VAE Short				
L	0.804	0.333	0.562	0.360
R	0.803	0.370	0.812	0.542
B	0.801	0.407	0.875	0.581
PBT Siamese				
L	0.687	0.458	0.662	0.300
R	0.598	0.491	0.600	0.391
B	0.643	0.467	0.508	0.314

Table 25. Accuracy Scores for all models trained on the Left Arm (L), Right Arm (R) Both Arms (B), Generated Left Arm (GL), Generated Right Arm (GR) or Generated Both Arms (GB) datasets. Models with a star(*) were optimized to minimize the validation set loss. Validation sets were taken as 10% of the training data, while the experimental datasets consisted of the 27 RBM generated sequences and the 16 DCA generated sequences.

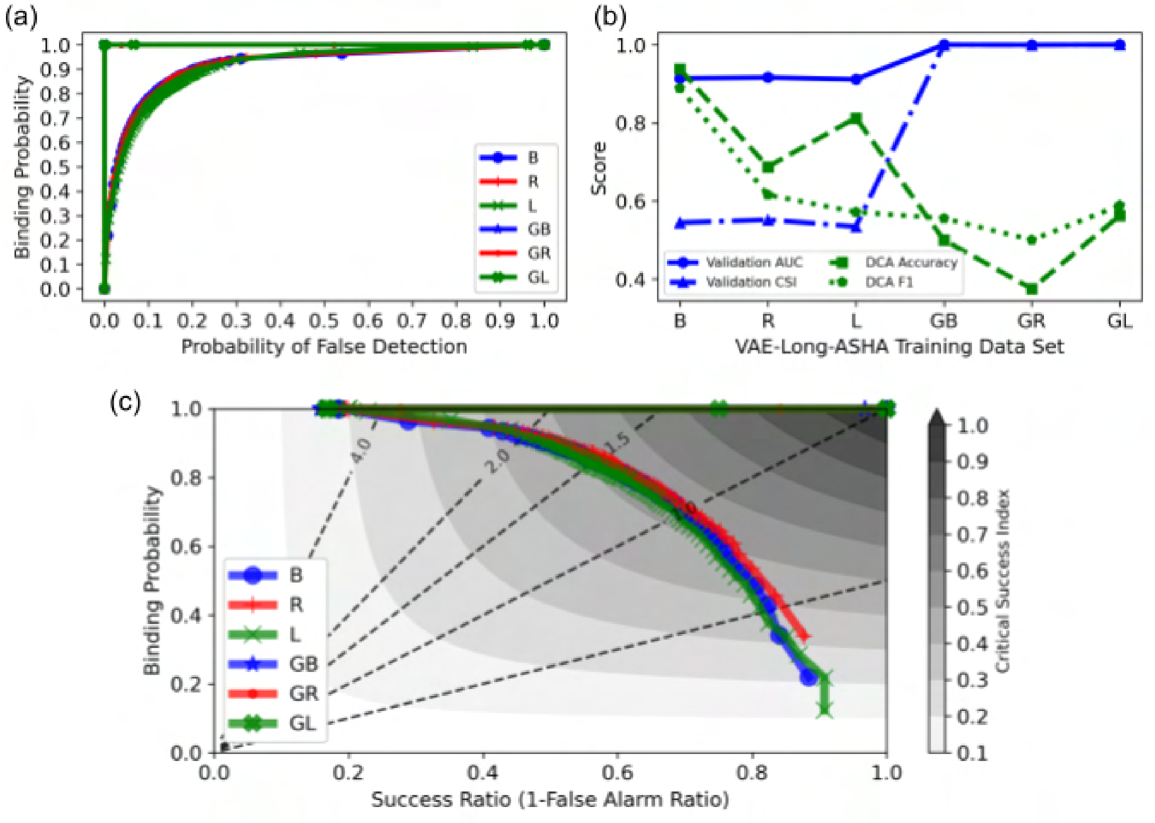


Figure 60. AUC (panel (a)), performance metrics (panel (b)), performance diagram (panel (c)) showing CSI for the VAE Long ASHA model.

Model	Validation Acc.	RBM Acc.	DCA Acc.	F1 Mean
Single Tree				
L	0.116	0.778	0.313	0.708
R	0.122	0.697	0.313	0.741
B	0.114	0.778	0.375	0.718
GL	0.999	0.704	0.813	0.764
GR	0.999	0.852	0.813	0.832
GB	0.885	0.889	0.625	0.781
Random Forest				
L	0.242	0.630	0.438	0.665
R	0.270	0.630	0.313	0.651
B	0.294	0.630	0.313	0.651
GL	0.950	0.741	0.375	0.684
GR	0.942	0.778	0.375	0.695
GB	0.939	0.778	0.438	0.706
Gradient Boosted Forest				
L	0.098	0.741	0.313	0.697
R	0.097	0.741	0.313	0.697
B	0.099	0.741	0.313	0.697
GL	0.091	0.741	0.313	0.697
GR	0.091	0.741	0.313	0.697
GB	0.167	0.741	0.313	0.697

Table 26. Accuracy Scores for single tree, random forest and gradient boosted forest trained on the Left (L), Right (R), Both (B), Generated Left Arm (GL), Generated Right Arm (GR) or Generated Both Arms (GB) datasets. Validation sets were taken as 20% of the training data, while the experimental dataset consisted of the 27 RBM generated sequences and the 16 DCA generated sequences.

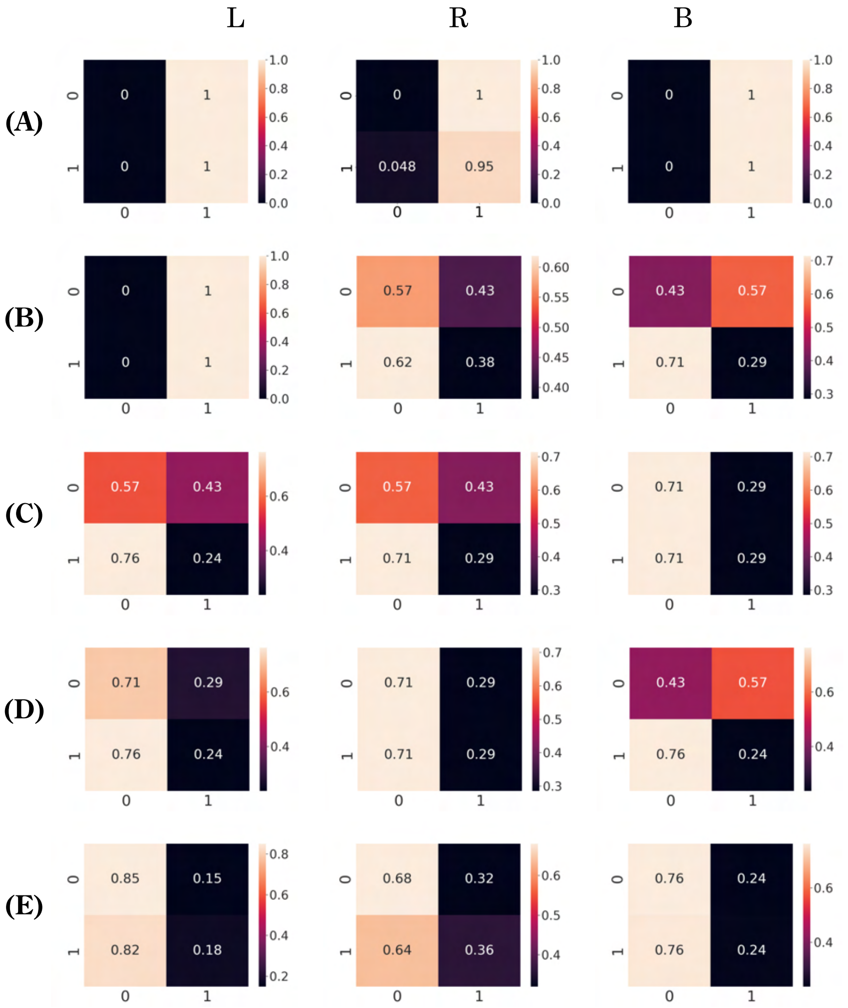


Figure 61. Confusion Matrices of trained loss-minimized or accuracy maximized bayesian optimized hyper-parameters on RBM generated dataset, (A) Long Resnet, (B) Short Resnet, (C) Short VAE, (D) Long VAE, and Population based training of Siamese Network (E). Predicted Label (0) nonbinder or (1) binder is shown on the x-axis with the true label being the y-axis.

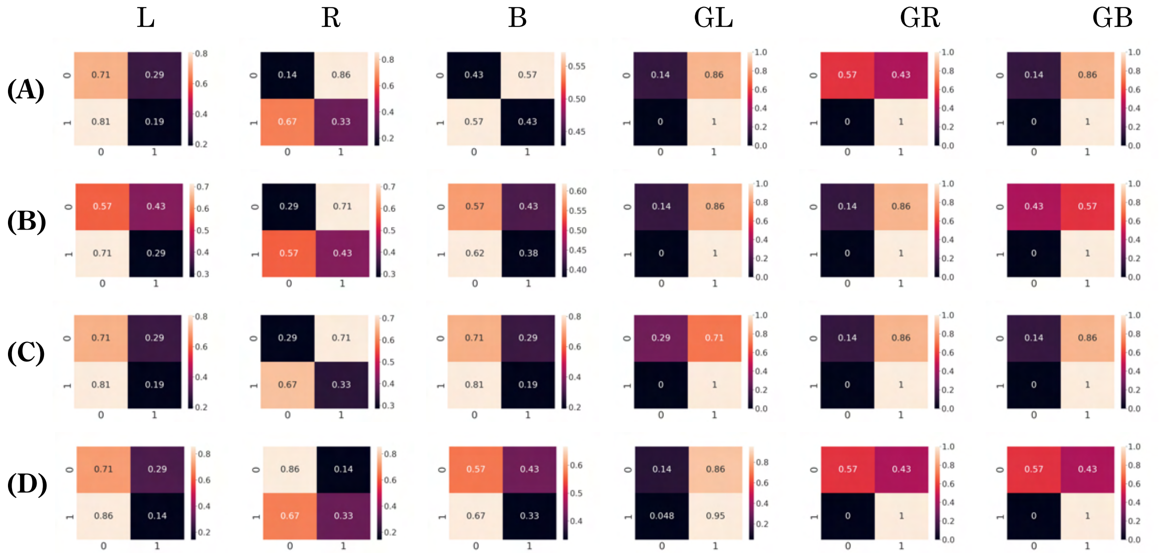


Figure 62. Confusion Matrices of accuracy maximized ASHA scheduler for hyper-parameter optimization using deep learning models: Long Resnet (A), Short Resnet(B), Long VAE (C) and Short VAE (D) on the RBM generated dataset. Predicted Label (0) nonbinder or (1) binder is shown on the x-axis with the true label being the y-axis.

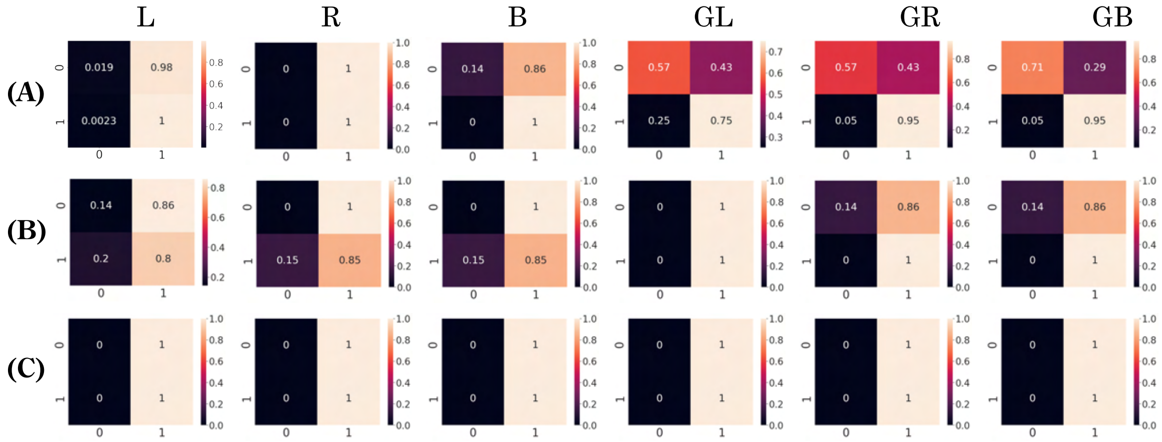


Figure 63. Confusion Matrices of traditional machine learning models: Single Tree, Random Forest and Gradient Boosted Forest on the RBM generated dataset. Predicted Label (0) nonbinder or (1) binder is shown on the x-axis with the true label being the y-axis.

B.5 Direct Coupling Analysis

Direct Coupling Analysis (DCA) is a method of analysis originally used for contact prediction in proteins from sequence alignments of homologues. The basis of this method is that the homologue alignments have the same general native state to carry out their function. Despite their differences in sequence, all homologues will have similar inter-domain contacts. To maintain these contacts, detrimental single site mutations must be offset by compensatory mutations in other parts of the sequence. DCA is a maximum-entropy method, where the model parameters are fixed so that the one- and two-point correlations along the sequences are fixed to those observed in the training homologue-sequence aligment. The sequence probability is given in Eq. (B.9) and is dependent on the learned single position parameters (h) and pairwise interactions (J_{ij}) of the multiple sequence alignment.

$$P(\sigma) = \frac{1}{Z} \exp\left(\sum_{i=1}^{L} h_i(\sigma_i) + \sum_{1 \leq i < j \leq L} J_{ij}(\sigma_i, \sigma_j) \right). \tag{B.9}$$

Similar to the protein case, we applied DCA on our aligned DNA aptamer dataset to approximate the aptamer sequence space with the learned single site and pairwise correlations. Sequences unobserved in the original dataset were generated from the learned parameters and tested experimentally.

B.5.1 DCA Training

The training set used for DCA analysis was a subset (90%) of sequences with copy number >1 from the 8th round of selection. Rather than separate the arms of each nanotile, the DCA model was trained with on 40 nt long sequences containing both arms. The normalization constant Z is difficult to calculate, so we use psuedolikelihood maximzation DCA (plmDCA) [271] to obtain local fields (h_i) and pairwise coupling (J_ij) for the model, given the aligned aptamer dataset. Monte Carlo sampling was applied across a range of temperatures and mutation steps to sample from the learned parameters. In total $2{*}10^9$ sequences were sampled, and from those 16 sequences shown in Table 27 were selected for experimental validation of the model.

B.5.2 DCA Sequence Selection

From the generated sequences, we wanted to find not only novel binders but also verify the learned model parameters. Sequences are scored according to the sum of their single position and pairwise parameters. A sequence's higher score indicates it is more likely to bind while a lower score indicates it is less to bind. Predicted binders (sequences d1, d2, d3, d4, d5, d11, d12, d13) were selected from the MC-generated sequences by having the highest score while being at least 3 mutations away from anything observed in the entirety

Label	Sequence	Score	Binder Prediction	Experimental result
d1	AGGGTAGGTGTGGGGTATGC	86.92	B	NB
d2	AGGGTAGATGTGTAGGATGC	87.86	B	NB
d3	AGGGATGATGGTTGGTAGGC	84.76	B	NB
d4	AGGGATGATGTGGATTAGGC	86.03	B	NB
d5	AGGGTGGGAGCGGGGGACGC	75.01	B	NB
d6	CGGGTAGGTGTGGATTATGC	77.59	B	NB
d7	GTAGGACGGGTAGGGCGGTC	67.57	NB	NB
d8	GGGGGTTGGGCGGGATGGGC	72.15	B	NB
d9	GCGGGTTGGGCAGGATCAGC	44.58	NB	NB
d10	AGGGATGATGTGTGGTAGGC	N/A	Cntrl	Cntrl
d11	GTAGGATGGGTGGGGTGGGA	86.46	B	B
d12	GTAGGATGGGTAGGGTGGTA	84.76	B	B
d13	CTAGGTTGGGTAGGGTGGTG	75.01	B	B
d14	CTAGCATGGGTAGGGTGGTG	77.59	B	B
d15	GTAGCATGGGTAGGGTGGTC	65.57	NB	NB
d16	TTGGGTGGTGTAGGTTGGCG	72.15	B	B
d17	TTGGGTGGTGCAGGTTCGCG	44.58	NB	NB
d18	CTAGGATGGGTAGGGTGGTG	N/A	Cntrl	Cntrl

Table 27. Result of thrombin binding assays with all DCA-generated sequences and sequences of exosite I control d18 and exosite II control d10. B indicates a binder while NB indicates a nonbinder.

of the 8th round of sequencing data. Two predicted nonbinders (d6, d14) were selected for having the lowest score within 2 mutations of the dataset. Rationally designed binders (d7, d8, d9, d15, d16, d17) were generated by randomly selecting a good and bad binder from the original dataset and altering them to either have the highest or lowest score possible by exhaustively calculating the entire sequence space within 3 mutations and finding the variant with the highest or lowest score. Model parameters used to generate all sequences are shown in Fig. 64.

B.5.3 DCA Gel Shift Assay

Sequences generated using the plmDCA model were tested experimentally for their ability to bind Thrombin. Binding sequences formed a clear protein / stem-loop band. Sequences were tested the same way as done for the RBM-generated sequences in the main text. Fig. 65 shows the experimental results of a gel shift assay for the plmDCA generated sequences.

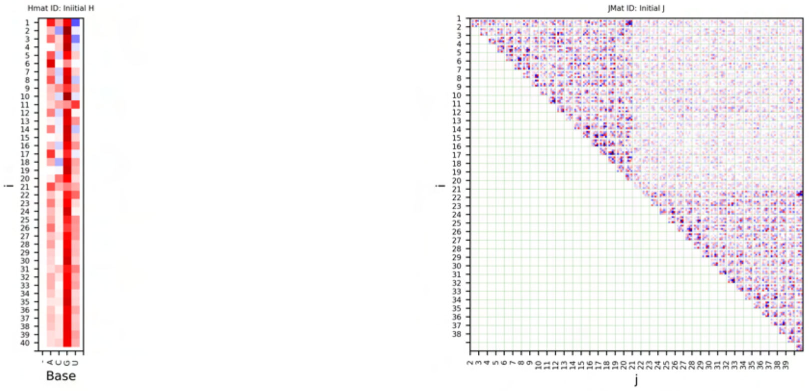


Figure 64. Single position (H) and pairwise correlations (J_{ij}) learned by the plmDCA model and used in both sampling and sequence selection.

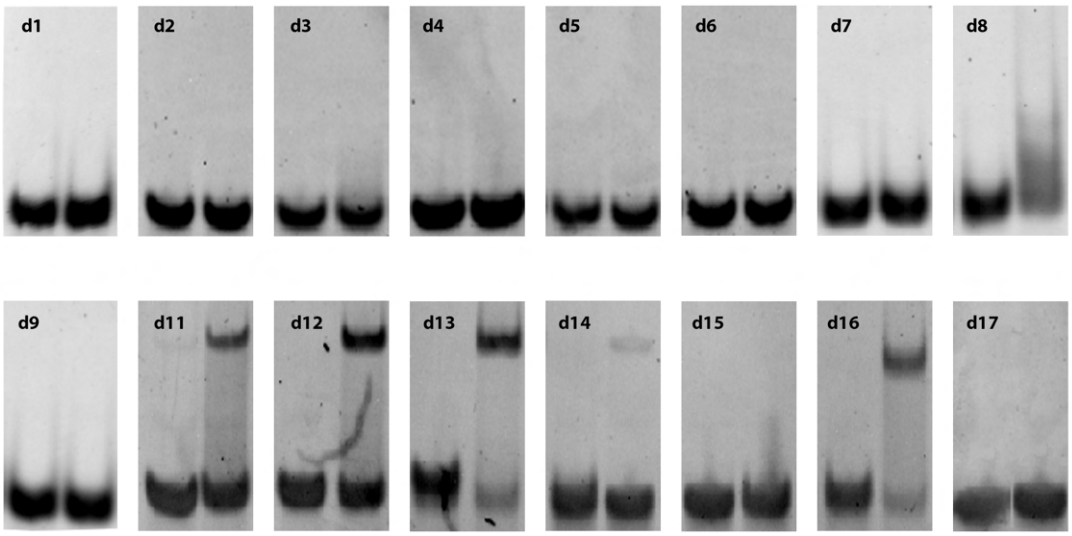


Figure 65. Thrombin binding assay of DCA generated sequences. Lane 1 has the stem loop alone, whereas lane 2 has the same stem loop exposed to thrombin. Binding sequences are indicated by a high visible band in lane 2.

B.5.4 DCA Binding Site Assay

Thrombin binding sequences generated via plmDCA (d11, d12, d13, d14, d16) were tested against known binders 5' 6FAM labeled ThA and ThD to determine their binding site as described in the main text. Table 28 contains the results and Fig. 65 shows the gel results.

Label	Sequence	Binding Site
d11	GTAGGATGGGTGGGGTGGGA	exosite I
d12	GTAGGATGGGTAGGGTGGTA	exosite I
d13	CTAGGTTGGGTAGGGTGGTG	exosite I
d14	CTAGCATGGGTAGGGTGGTG	exosite I
d16	TTGGGTGGTGTAGGTTGGCG	exosite I

Table 28. Exosite prediction of DCA sequences that bound thrombin from our gel shift assays.

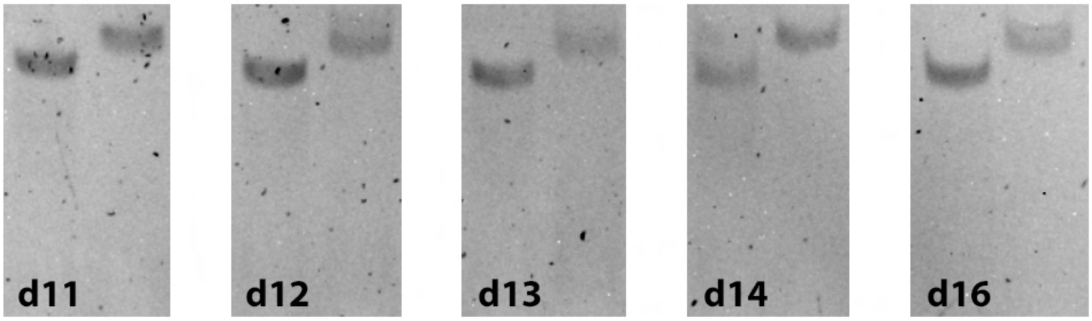


Figure 66. Binding site assay using the same method discussed in the main text. Lane 1 is the result of the preincubated strand exposed to exosite-II binder ThA and lane 2 is the preincubated strand exposed to exosite-I binder ThD.

B.5.5 DCA Results

The weak pairwise correlations seen in the top right corner of the pairwise correlation matrix (J_{ij}) confirm the lack of correlation between the two arms of each nanotile. The plmDCA method did see limited success in generating novel binders (d11, d12, d13, d16) from the right loop sequences but no success in generating binding left loop sequences (d1, d2, d3, d4, d5) (Fig. 66).

We tried also to train a DCA using the same algorithm we used for the RBM models to obtain the model parameters, i.e. the persistent contrastive divergence algorithm. Moreover, building on the results obtained with our RBM models, we decided to use all the available sequences to train the BM model, neglecting the counts. Then we compared, for the obtained DCA model trained with single-loop sequences at round 8, the log-likelihood assigned by the DCA with the one assigned by an RBM trained on the same data. The resulting plot is given in Suppl. Fig. 67, and this test gave a very good linear correlation between the log-likleihoods of the two models (slope of the linear fit: 1.09; R^2 score: 0.97), suggesting that the DCA model trained with persistent contrastive divergence has superior generalization capabilities with respect to plmDCA models. This result is compatible with what observed in [272].

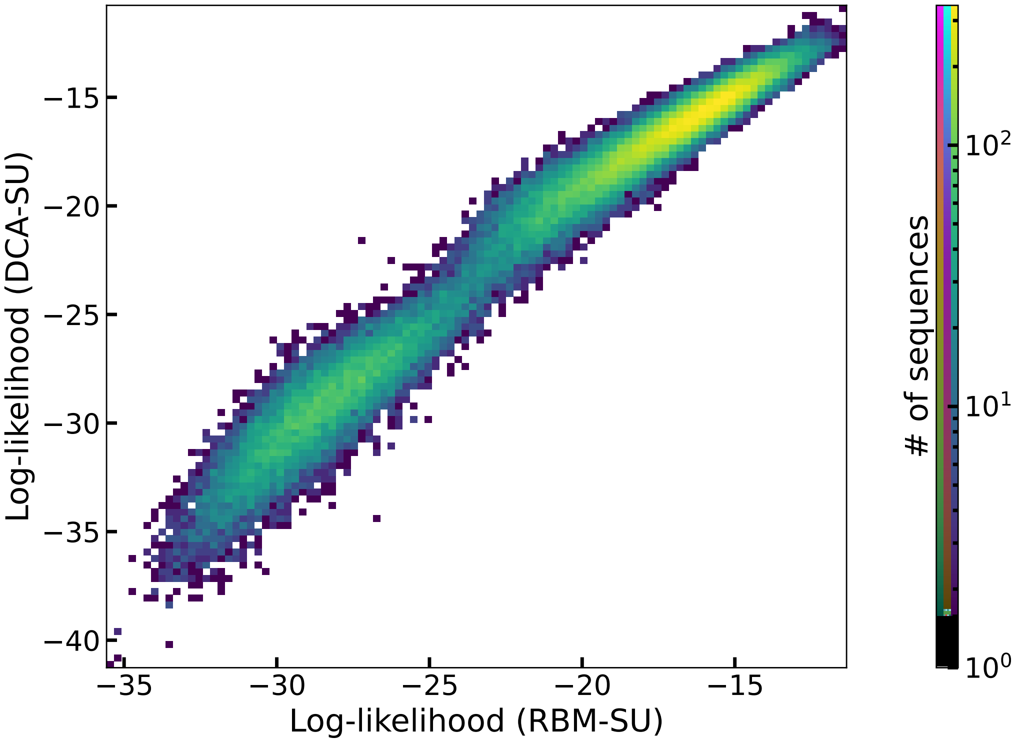


Figure 67. Log-likelihood of all unique single-loop aptamers observed at round 6, as computed by a DCA and an RBM model trained through persistent contrastive divergence. The corresponding linear fit resulted in a slope of 1.09 and an R^2 of 0.97.

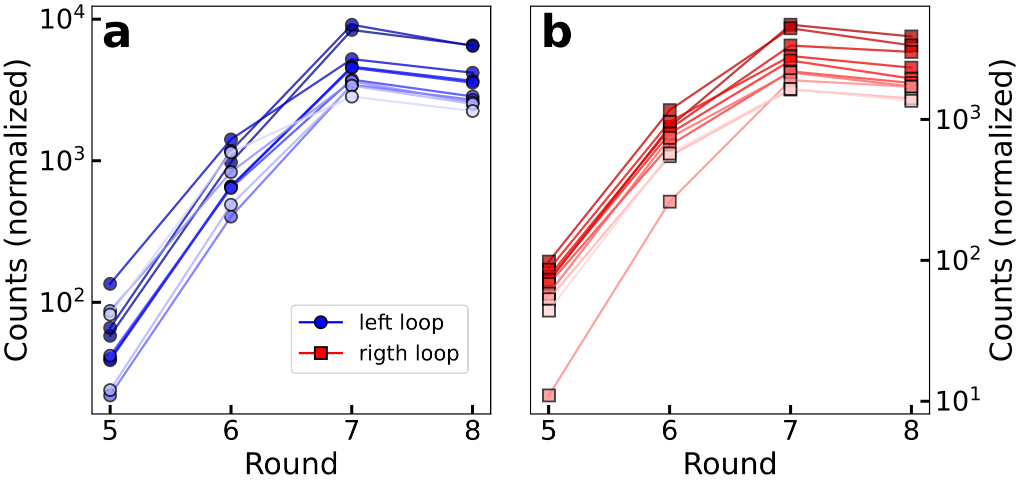


Figure 68. Evolution of counts of the 10 left (panel a) and right (panel b) aptamers with largest number of counts at round 8. Counts have been re-scaled by a factor so that the total number of counts in each round is constant.

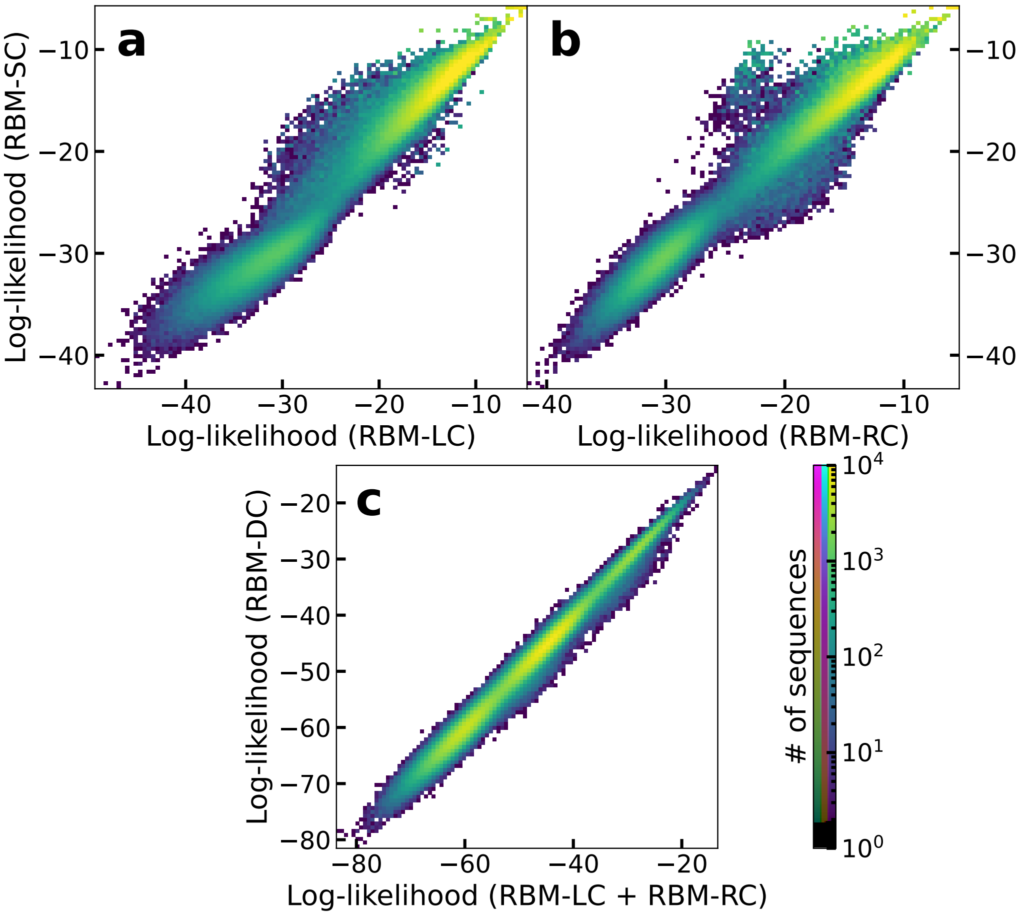


Figure 69. Log-likelihood computed with the RBM-SC model and with the RBM-LC model (trained on left single-loop sequences at round 8, see B.3) in panel a or RBM-RC model (trained on right single-loop sequences at round 8, see B.3) in panel b for the single-loop sequences observed at round 8. The slope and the R^2 values of the linear fit are respectively 0.96 and 0.98 for panel a, and 1.05 and 0.97 for panel b. Panel c: log-likelihood computed with the RBM-DC model for the double-loop sequences observed at round 5, compared with the sum of the log-likelihood obtained by using RBM-LC to score the left loop and RBM-RC to score the right loop. The slope and the R^2 value of the linear fit are, respectively, 0.99 and 0.99.

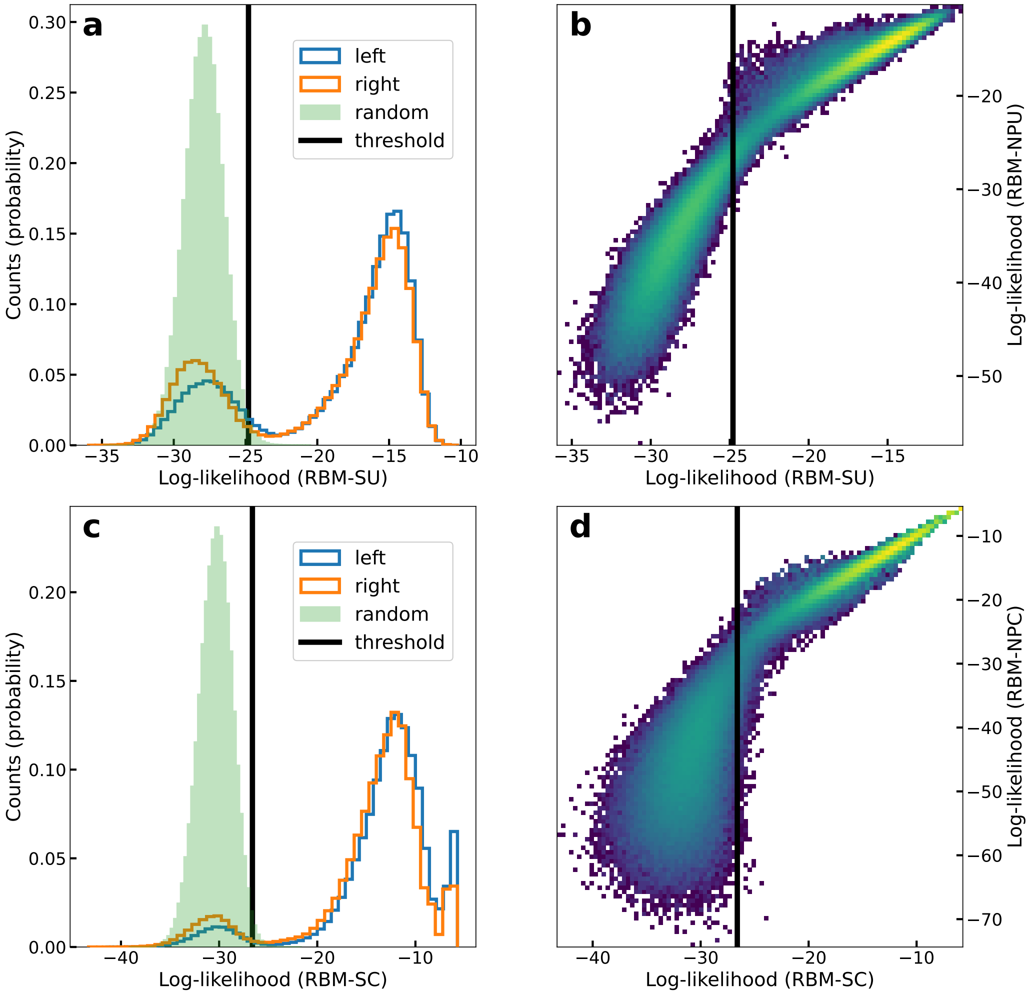


Figure 70. Left side: histograms of log-likelihoods of left (blue) and right (orange) loops computed with RBM-SU (panel a) or RBM-SC (panel c) for sequences observed in round 8 (unique in panel a, with their counts in panel b), together with that of $5 \cdot 10^5$ random uniform sequences (light green); the black line is the 99-quantile of the light green histogram, and parasite sequences are defined as those which have lower log-likelihood than the black line, while at the same time the other loop of the 40-nt aptamer has log-likelihood larger than the threshold. Right side: log-likelihood of the RBM trained after excluding parasite sequences at round 8 (RBM-NPU for panel b, RBM-NPC for panel d) versus that of the RBM-SU (panel b) or RBM-SC (panel d) model. A linear fit for the points at the right-hand side of the black line (which is the same of panels a for panel b, and of panel c for panel d) gives a slope of 1.0 and a R^2 of 0.92 for panel b, and a slope of 1.0 and a R^2 of 0.96 for panel d. For points at the left-hand side of the black line the slope is 2.6 with an R^2 of 0.79 for panel b, and the slope is 2.0 with an R^2 of 0.33 for panel d.

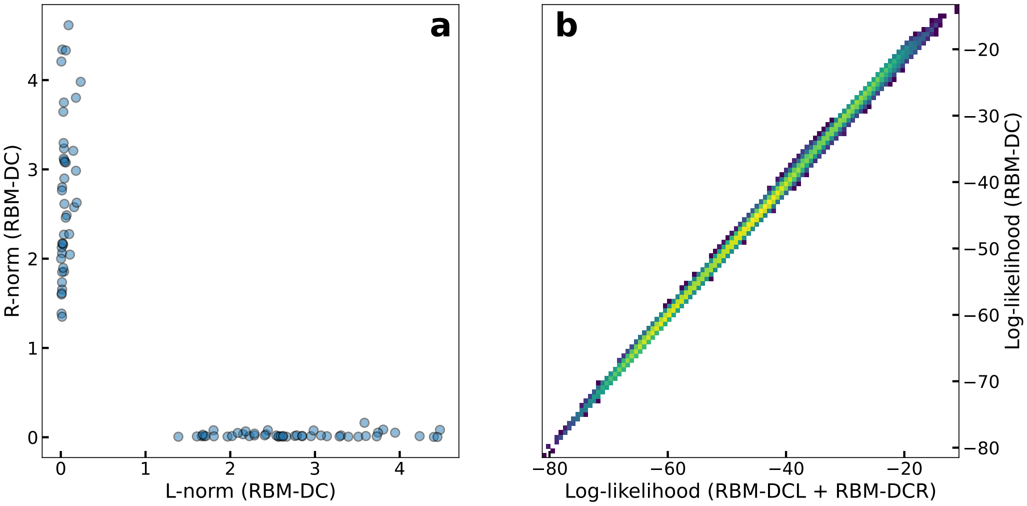


Figure 71. Panel a: Frobenius norms obtained for each weight of RBM-DC computed using only the first 20 visible units (L-norm in the x axis) or the last 20 visible units (R-norm in the y axis).

Panel b: RBM-DCL and RBM-DCR are two RBMs with 20 visible units used to score left and right loops. RBM-DCL (RBM-DCR) is obtained from RBM-DC by using only its first (last) 20 visible units and their fields, and the hidden units with L-norm > R-norm (R-norm > L-norm) with their potentials, ignoring their interactions with the last (first) 20 visible units. In this panel, we compare, for each unique double-loop sequence observed at round 5, the log-likelihood of the RBM-DC model with the sum of the log-likelihoods obtained by using RBM-DCL to score the left loop and RBM-DCR to score the right loop. The slope of the linear fit is 0.99 and the R^2 score is > 0.99.

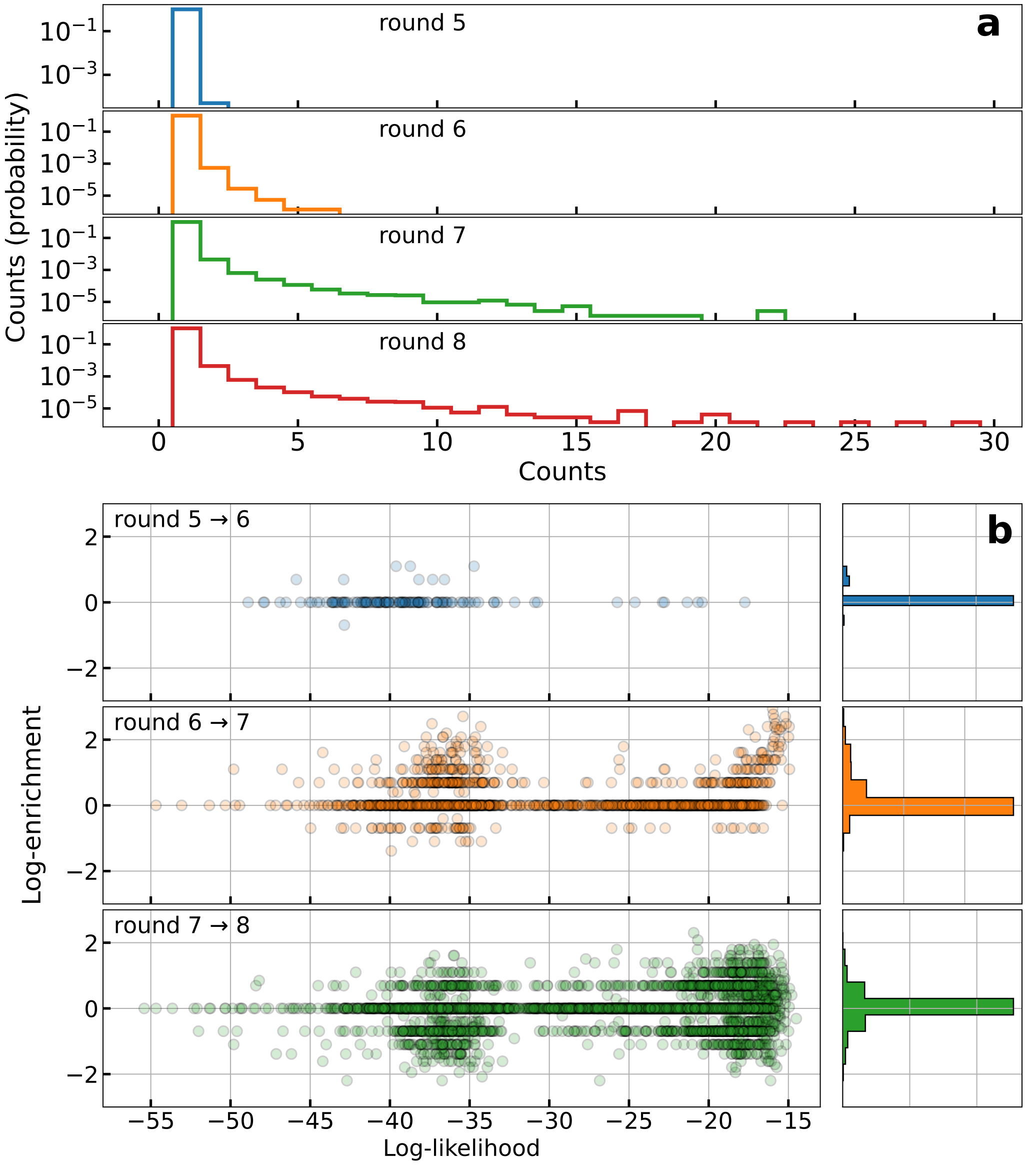


Figure 72. Panel a: probability density function of the counts observed for the double aptamers in each round. Notice the log scale on the y axis. Panel b: for each pair of consecutive rounds, we plot here the logarithm of the ratio of counts of the sequences present in both rounds (left) and the corresponding histogram (right), against the log-likelihood of the sequence computed with the RBM-DC model.

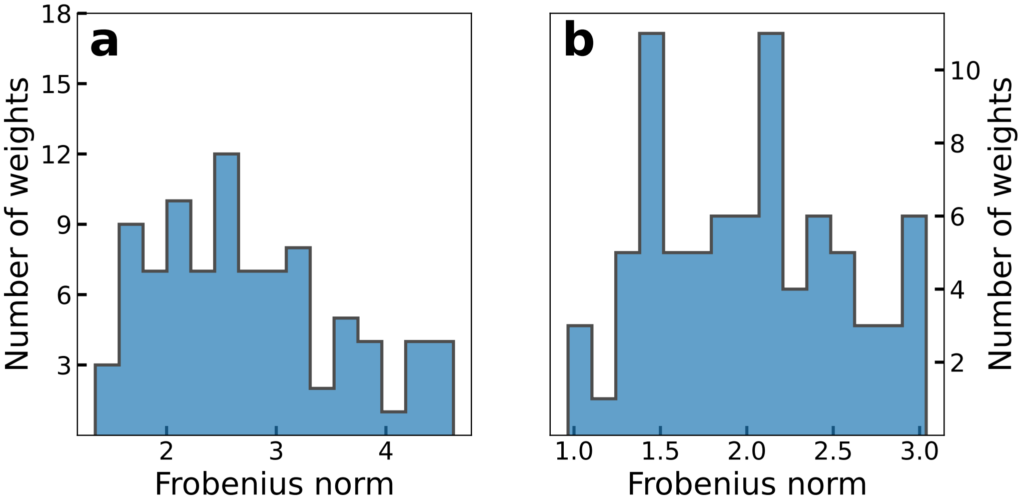


Figure 73. Panel a: Frobenius norms of the weights for RBM-DC. The logos corresponding to the 3 weights with largest Frobenius norm are given in Fig. 22a-c. Panel b: Frobenius norms of the weights for RBM-SC. The logos corresponding to the weight with the 2nd largest Frobenius norm and the one with the 7th largest Frobenius norm are given in Fig. 22e-f.

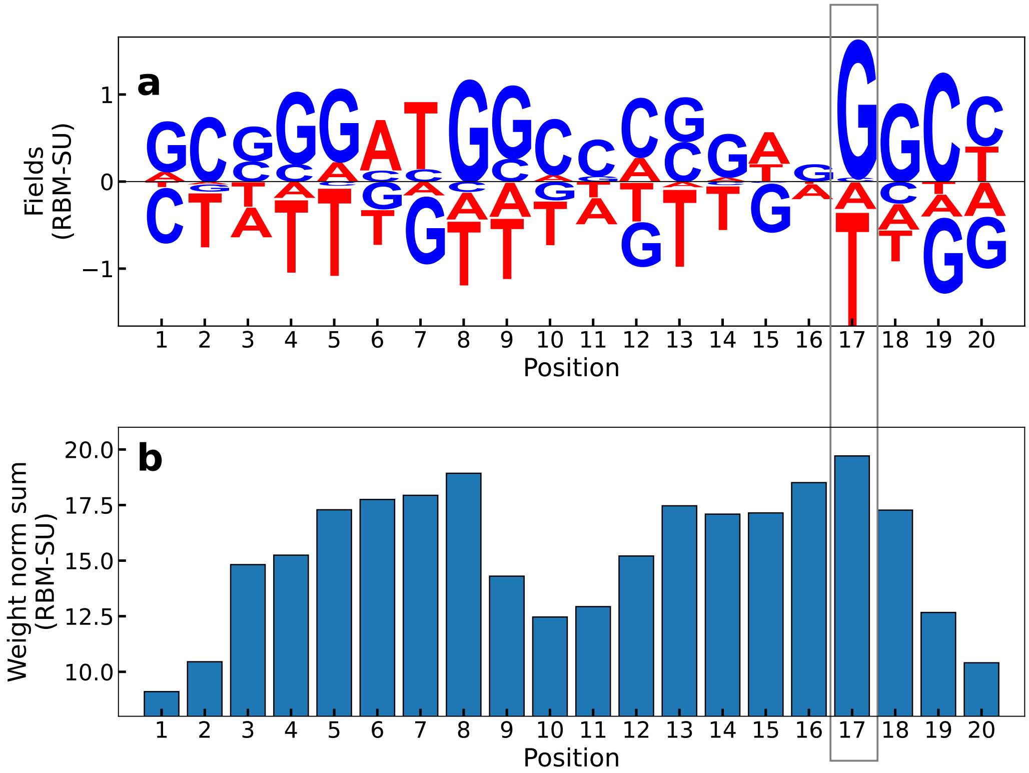


Figure 74. Panel a: fields of RBM-SU. The largest field (in norm) corresponds to position 17 (gray box), which is the one that in Fig. 78 determines the binding exosite. Panel b: sum of the norms of each weight of RBM-SU, at fixed sequence position. The largest sum corresponds again to position 17 (gray box).

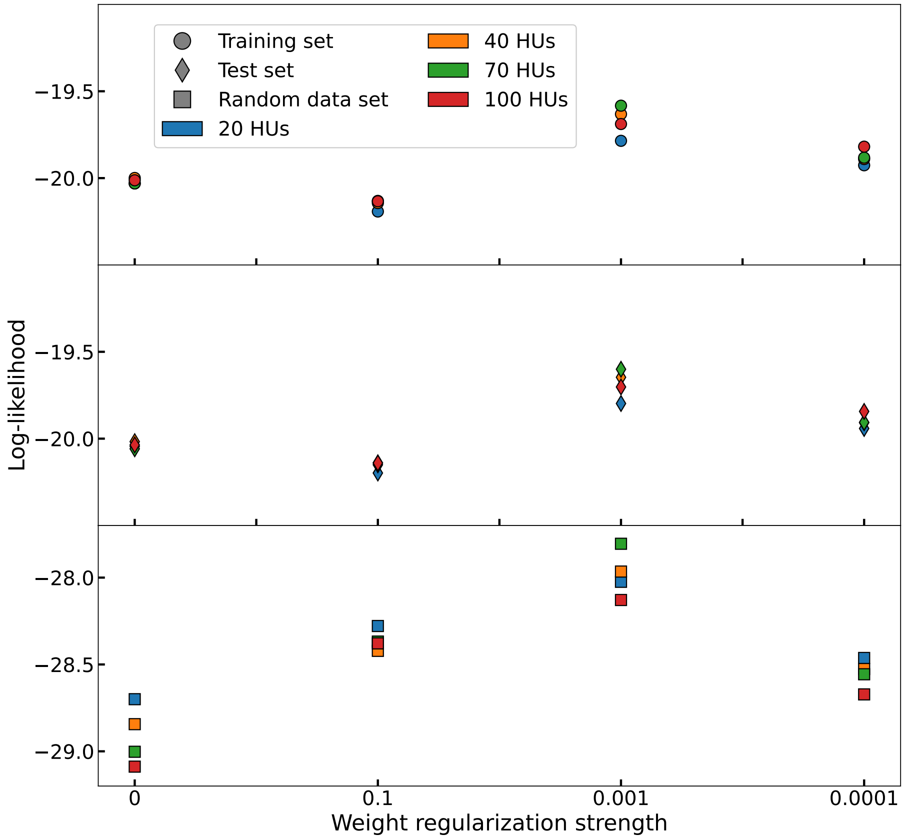


Figure 75. Average log-likelihoods computed with 16 RBMs trained with different choices of hidden unit numbers and weight regularization on the single aptamers obtained from the 8-th round. The scale on the y-axis is kept constant across the different sub-plots to highlight how the difference in average log-likelihoods are much smaller than the difference between the log-likelihood of training (and test) data and that of random sequences. The green circle at 0.001 regularization strength correspond to the RBM used in the paper (RBM-SU).

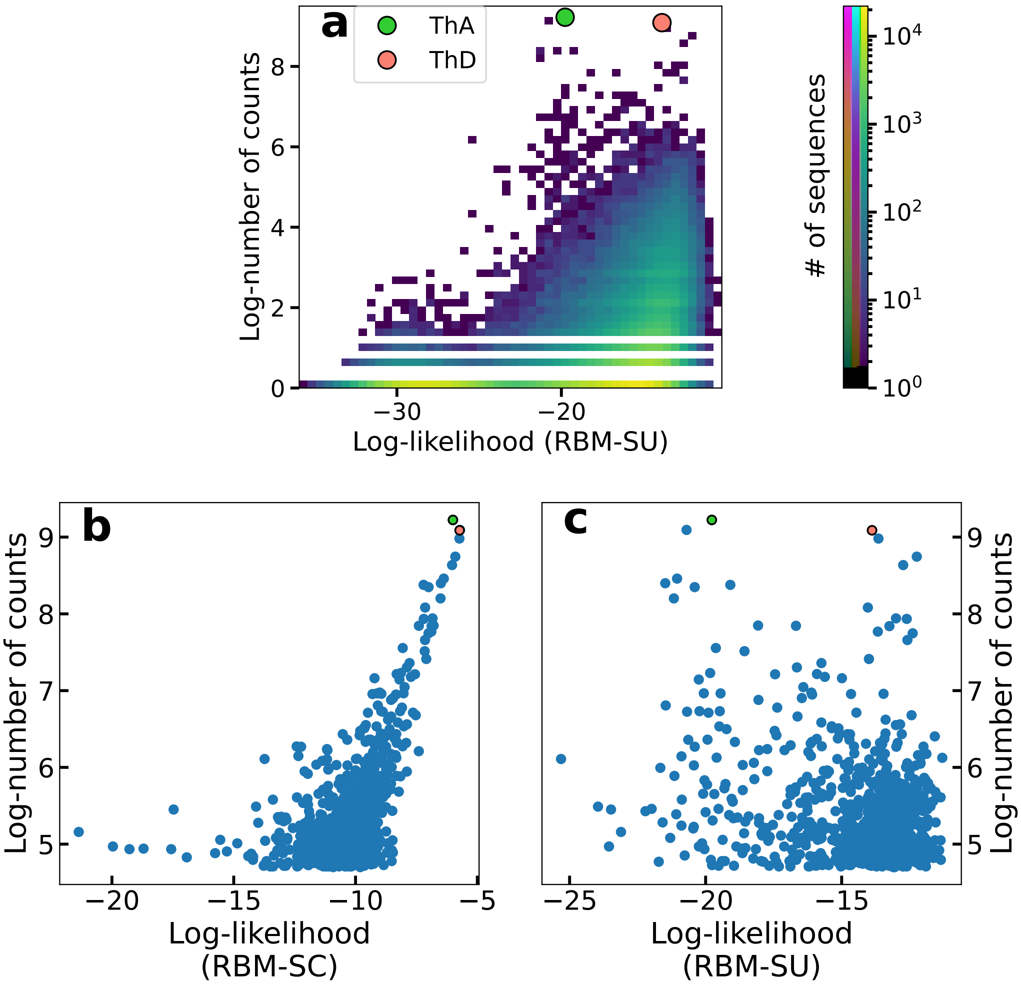


Figure 76. Panel a: Log-likelihoods (computed with RBM-SU) versus log number of counts for the unique single-loop sequences observed at round 8. ThA (counts: 10132, log-likelihood: -19.8) and ThD (counts: 8853, log-likelihood: -13.9) are highlighted with circles. Panels b, c: Log-likelihoods computed with RBM-SC (for panel b) or RBM-SU (for panel c) versus log number of counts for the 1000 unique single-loop sequences observed at round 8 with highest number of counts.

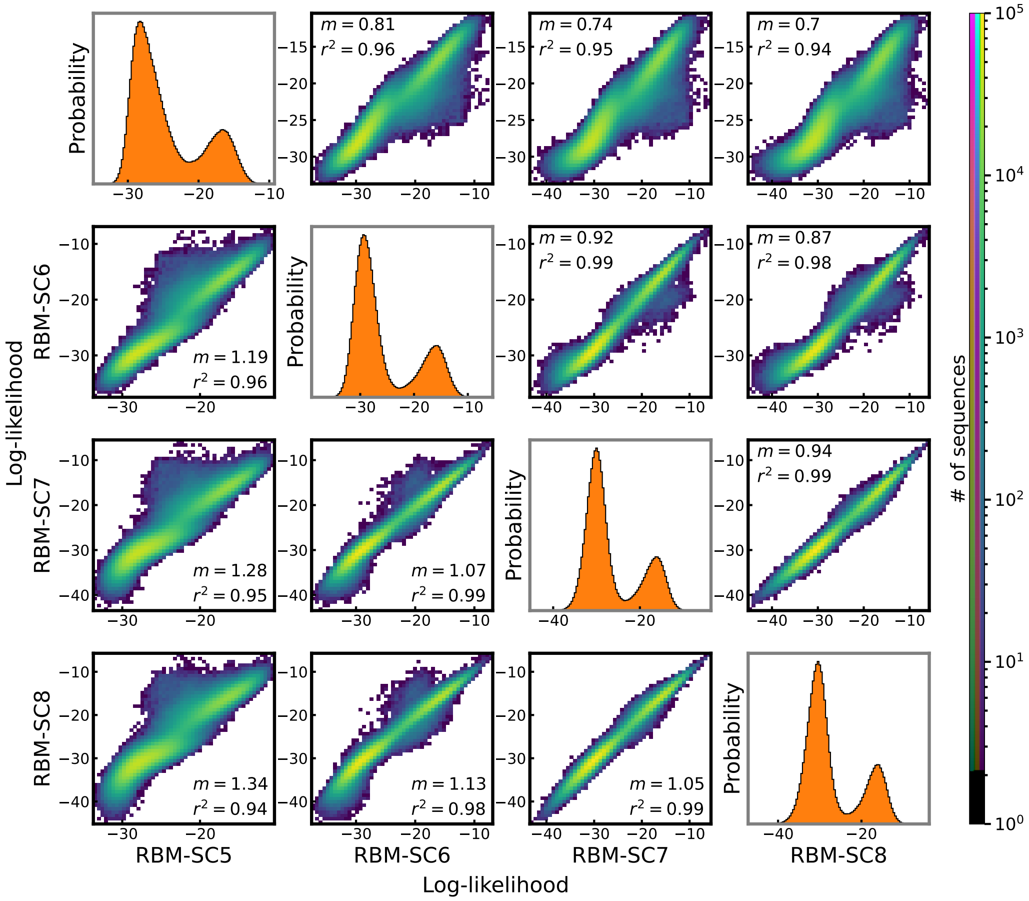


Figure 77. Comparison of the log-likelihoods computed with RBM-SC trained at different rounds (named RBM-SC5, RBM-SC6, RBM-SC7 and RBM-SC8 if trained respectively on sequences observed in round 5, 6, 7, 8). Plots on the diagonal are the distribution of the log-likelihoods of each RBM. The sequences used to prepare each histogram are the full set of sequences observed in round 5, 6, 7, or 8 (discarding counts). In each-non diagonal plot, the slope m and the coefficient of determination r^2 for the linear fit are given.

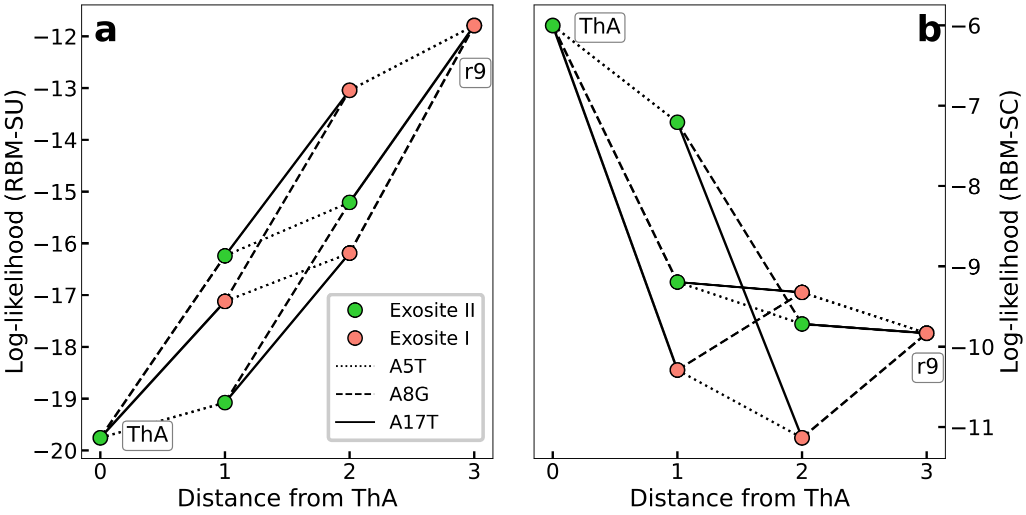


Figure 78. RBM-SU (panel a) or RBM-SC (panel b) log-likelihood versus distance from ThA for sequences p1 to p6 in Table 4. Different mutations are represented with different line styles: dotted lines for mutations involving position 5 (mutating A into T when going from ThA to r9), dashed lines for mutations involving position 8 (mutating A into G when going from ThA to r9), and solid lines for mutations involving position 17 (mutating A into T when going from ThA to r9).

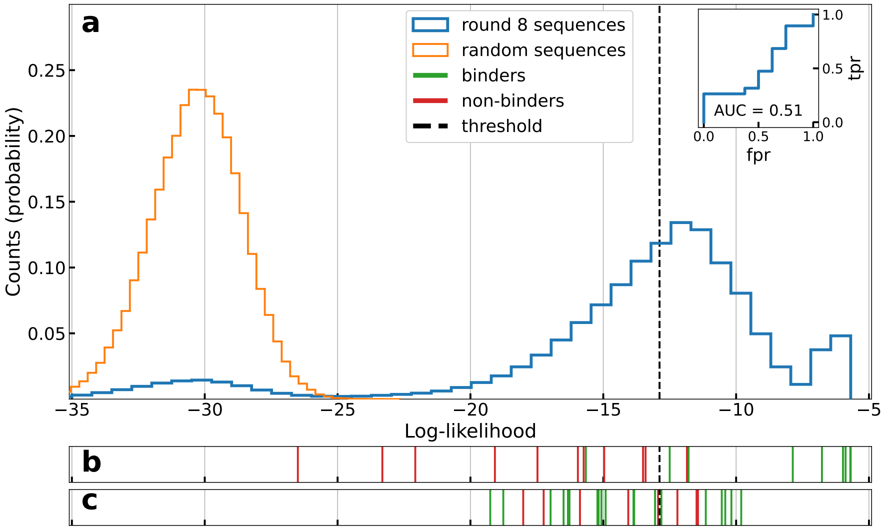


Figure 79. Panel a: Histogram of the log-likelihoods of all unique aptamers observed in the last round (blue line) and of uniformly random sequences (orange line), computed with RBM-SC trained on single-loop sequences from round 8, keeping information about the counts. Inset: AUC computed on the sequences generated by the RBM-SU model (panel c). Panel b: Vertical lines locate the log-likelihoods of sequences experimentally validated to be binders (green) or non binders (red). Sequences taken from a preliminary set described in Suppl. Table 27. Results allows us to determine the binding/non binding threshold, shown with the black dashed line. Panel c: same as panel b for sequences designed with the RBM-SU model, as described in Sec. 4.3.6 (see Table 4).

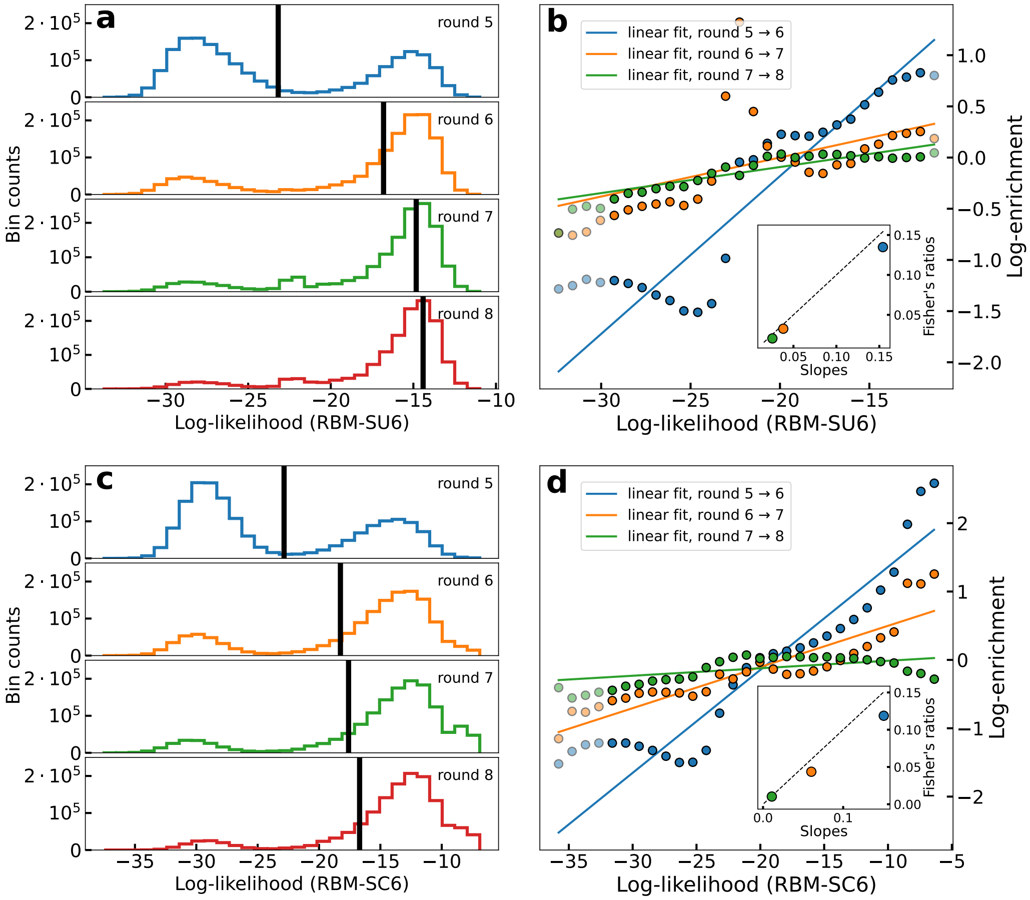


Figure 80. Relationship between log-enrichment and log-likelihoods of single-loop aptamers. Panels a, c show the histograms of log-likelihoods at each round, as computed by RBM-SU6 (panel a) and RBM-SC6 (panel c). Panels b, d show the scatter plot of log-enrichment of each bin in the left panels, and the corresponding log-likelihood. In the inset, the slope of each linear fit appearing in the main plot is compared with the same quantity estimated as a Fisher's ratio (see Sec. 4.3.3). The dashed black line is the $x = y$ line.

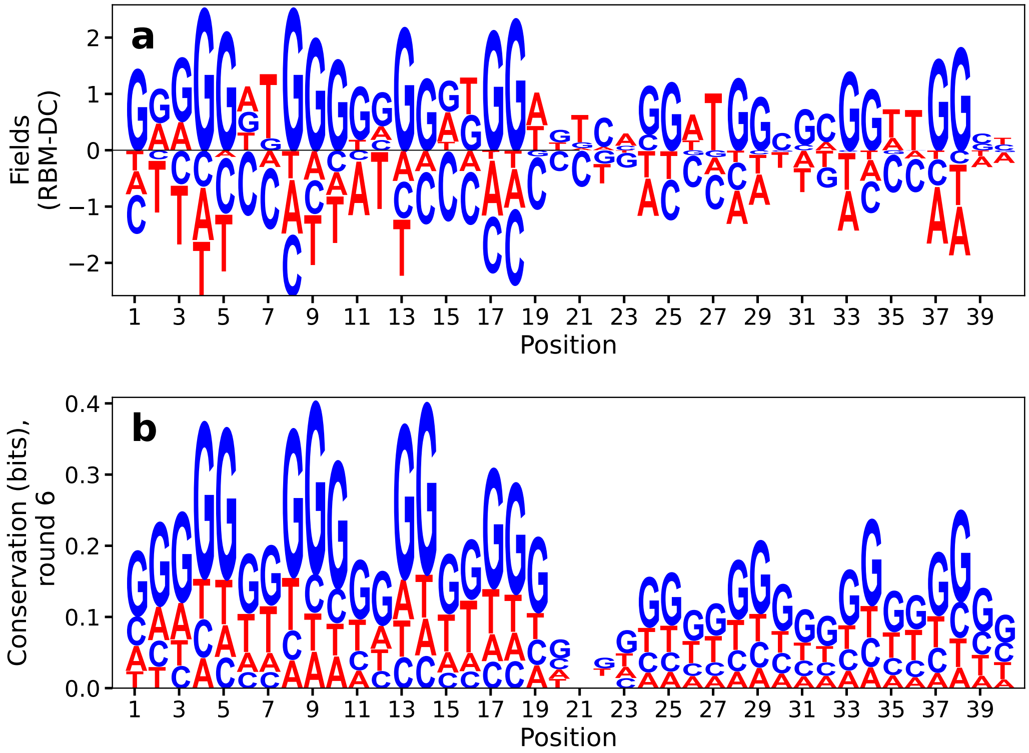


Figure 81. Local field learned by the RBM-DC6 used in Fig. 20 (panel a), compared with the conservation logo of the full dataset at round 6 (panel b).

Label	counts round 8	Dist1	Dist3	Dist10	Dist100
r1	3	0	0	1	1
r2	3	0	0	1	1
r3	0	1	1	1	1
r4	0	1	1	2	2
r5	0	1	1	2	2
r6	242	0	0	0	0
r7	341	0	0	0	0
r8	11	0	0	0	1
r9	9	0	0	1	2
r10	0	1	2	2	3
r11	0	2	2	2	4
r12	0	1	2	3	3
r13	0	2	2	3	5
r14	0	2	2	2	5
r15	0	2	2	2	4
r16	0	1	2	2	3
r17	0	1	2	3	4
r18	528	0	0	0	0
r19	139	0	0	0	0
r20	10	0	0	0	1
r21	8	0	0	1	2
r22	0	2	2	2	2
r23	0	1	1	2	4
r24	0	1	1	1	3
r25	0	1	1	2	3
r26	0	1	3	3	4
r27	0	1	1	1	3

Table 29. For each sequence generated from RBM-SU trained on unique loop sequences observed in the last round, we provide here the distance from the closest single-loop aptamer observed at round 8 (column Dist1, 382094 sequences) and the number of counts of each sequence at round 8. Since a good binder is expected to be found close to a sequence with many counts, we also provide in the other columns (Dist3, Dist10, Dist100) the distance to the closest single-loop aptamer with at least, respectively, 3, 10 or 100 counts in round 8 (respectively 74785, 22332, and 1177 sequences).

APPENDIX C

PUBLICATION PERMISSIONS

Permission was granted by all co-authors to include the published articles of Chapters 2,3, and 4.