Synthetic Molecular Machines for Active Self-Assembly: Prototype Algorithms, Designs, and Experimental Study

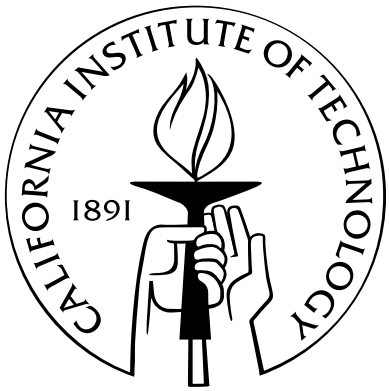
Thesis by

Nadine L. Dabby

In Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy



California Institute of Technology

Pasadena, California

2013

(Defended February 4, 2013)

This manuscript is dedicated to

Yvette and Joseph Dabby (mom and dad): Of all the brilliant minds and beautiful souls that I have stumbled upon in this great wide world, there are no two people whom I admire more, or aspire more to be like. It is hard for me to imagine ever being as strong as you, or having as much integrity, or so much to give.

*

To Peter C. Beller: you are my eyes, my soul and my home.

*

And to my thesis advisor Richard Murray: Thank you for believing in me.

Acknowledgements

Sio credesse che mia risposta fosse
A persona che mai tornasse al mondo,
Questa fiamma staria senza piu scosse.
Ma perciocche giammai di questo fondo
Non torno vivo alcun, siodo il vero,
Senza tema dinfamia ti rispondo.

"The Inferno," Dante Alighieri [Alighieri, 2013]

It is fitting to invoke my first intellectual crush here. [1]

A short while after starting at Caltech in my first year of graduate school, I went to the Millikan library to photocopy an article that had been published in 1977 and had not yet been taken up by the digitization movement [Gillespie, 1977]. I remember looking out of the windows on the 8th floor toward the mountains in the north and thinking about how lucky I was to be getting paid to learn and ask questions and think. There was a peace that came with the promise of objectivity.

I came to this place via quite the circuitous path, asking some questions along the way [Dabby, 2003]. And there were many people who encouraged me onward (specifically **Kevis Goodman** and **Lynn Hejinian**, both faculty members of the amazing UC Berkeley English Department). Back then (not unlike now), it was not clear how I should proceed. I had many questions and I did not even know how to think about them in a productive way. I somehow had not managed to

[1] The epigraph to The Love Song of J. Alfred Prufrock [Eliot, 1995]. The passage translates to: "If I thought my answer were given to anyone who would ever return to the world, this flame would stand still without moving any further. But since never from this abyss has anyone ever returned alive, if what I hear is true, without fear of infamy I answer you."

jam in everything I needed to learn in my undergraduate career [2]. So I followed my instincts and I attempted to catch up on math and learn the thought process behind computer science.

Shuki Bruck is the reason I applied to the Computation and Neural Systems Ph.D. program at Caltech. I will never be sure of what compelled the admissions committee to accept me [3]. At Caltech I was pushed to think, to work hard and to push the limits of my creativity. It was tough: truth be told, brutal at times, and at every pitfall I found some new strength within myself that I did not know I had in me. That toughness has (somehow magically over time) given me a profound sense of confidence that I consider priceless.

This dissertation would absolutely not have been possible without the support and help of many people who believed in me.

The most important of these is **Ho-Lin Chen**, my mentor, colleague and friend. You have one of the most brilliant minds I have ever encountered and yet you are so humble about it.

My thesis committee has been a source of great strength and insight.

To **Richard Murray**: Thank you for giving me the most important gift of all – a room of my own [Woolf, 1957]. A key can be a magical thing: it opens doors and eyes and thoughts and minds, and in all those things: ideas, dreams, imaginations and opportunities. A refuge, a sacred place for quiet thoughts and a home for conversations. It can awaken a spirit and a soul simply by opening up the possibility of thinking freely. In your lab I have had the most important epiphany of all. You probably think that this was a small thing for you to do, but it has made all the difference in the world to me.

To **Niles Pierce**: Thank you for doing beautiful work. Thank you for opening up your lab to me and for guiding me through my experiments.

To **Al Barr**: I am very lucky that I first met you at the CNS interview weekend. I have really appreciated all of your thoughtful advice and feedback and the I have enjoyed working with you over these last few months. Thank you for your time and your patience.

To **Milan Stojanovic**, in a parallel universe you would have been by PhD. advisor and I would have been ecstatic about it. Thank you for being my would-be advisor. I have valued every con-

[2] While my formal education is coming to an end, I imagine that there is yet so much more to learn.

[3] Although I do have it on good authority that I was "an experiment"...

versation and every piece of wisdom you have tried to impart (even if I ignored some of it).

To **Deborah Fygenson**: Thank you for mentoring me. Your encouragement and advice have been invaluable to me.

To **Shin Shimojo**: Thank you for listening and for reminding me that I am not the only alien here.

I don't think graduation would even have been a possibility without a few guardian angels that have been watching out for me–**Paul W. K. Rothemund** and **Mathieu Desbrun**.

Thank you to the Murray Lab Biocircuits Subgroup (**Emzo de los Santos, Marcella Gomez, Clare Hayes, Victoria Hsiao, Vanessa Jonsson, Jongmin Kim, Joseph Meyerowitz, Dan Siegal-Gaskins, Vipul Singhal, Zachary Sun, Anandh Swaminathan, Anu Thubagere, Zoltan Tuza, Ophelia Venturelli**, and **Enoch Yeung**), an amazing group of people that have made my last few months at Caltech fun. Special thanks to Anu and Jongmin for patience with my Spex usage, to Zach and Clare for their friendship and to Jongmin, Zoltan and Dan for last minute o.d.e. aid.

Thank you to the Pierce Lab, another amazing group of people. In particular, I received a lot of help and support from **Victor Beck, Colby Calvert, Harry Choi, Lisa Hochrein, Jennifer Padilla, Maayan Schwarzkopf, Jonathan Sternberg**, and **Jeff Vieregg**.

Also thank you for support and help to **Tanya Owen, Karolyn Yong, Lucinda Acosta** and **Anissa Scott**.

Thank you to my friends, my mentors and colleagues who I have had the great pleasure of learning from during my time at Caltech: **Catherine Beni, Elisa Franco, Ashwin Gopinath, Greg Griffin, Rizal Hariadi, Ruxandra Paun, Joseph Schaeffer, Rebecca Schulman, David Soloveichik, Merrielle Spain, Sungwook Woo**, and **David Zhang**.

A special thank you to friends **Takeshi Irie** and **Nilesh Shah** for saying just the right things at the right time.

During my time at Caltech I had the great privilege of teaching at the Art Center College of Design. I have **Ann Marie Polsenberg Thomas** to thank for the opportunity. And I thank my colleagues and students there for teaching me, with particular thanks to **Julia Hur** and **Bruce Hubbard**.

To my parents, **Yvette** and **Joseph Dabby**: Life is about luck and then about choice. I feel so

lucky to be born into your world, to have learned even just a fraction of what you have to teach. And it wasn't so much anything you ever said, but more just watching the example that you set. Kind and strong, and ballsy enough to stand behind your name, your reputation and your beliefs even in difficult circumstances. You showed me the importance of family, love, community, and that it is only through this love that one can change the world, even if the part we are changing is only our small corner of the world.

To my sisters **Sharon Naomi Dabby** and **Lisa Dabby Joyce**: You were my first mentors, and you are now my closest friends.

To my husband **Peter C. Beller**: You are my soulmate and my partner in every possible way. Thank you for sitting with me in lab on Saturdays and for feeding my soul.

Thank you to my collaborators: **Ho-Lin Chen, Scott Goodfriend, Alexander Johnson-Buck, Kyle Lund, Anthony J. Manzo, Nicole Michelotti, Jeanette Nangreave, Renjun Pei, Joseph Schaeffer, Milan N. Stojanovic, Steven Taylor, Nils G. Walter, Erik Winfree, Damien Woods, Hao Yan, Peng Yin**, and **Bernard Yurke**.

This thesis was supported by generous funding from the NSF (Graduate Research Fellowship, Molecular Programming Project and other grants), and NIH (training grants).

Abstract

Computer science and electrical engineering have been the great success story of the twentieth century. The neat modularity and mapping of a language onto circuits has led to robots on Mars, desktop computers and smartphones. But these devices are not yet able to do some of the things that life takes for granted: repair a scratch, reproduce, regenerate, or grow exponentially fast–all while remaining functional.

This thesis explores and develops algorithms, molecular implementations, and theoretical proofs in the context of "active self-assembly" of molecular systems. The long-term vision of active self-assembly is the theoretical and physical implementation of materials that are composed of reconfigurable units with the programmability and adaptability of biology's numerous molecular machines. En route to this goal, we must first find a way to overcome the memory limitations of molecular systems, and to discover the limits of complexity that can be achieved with individual molecules.

One of the main thrusts in molecular programming is to use computer science as a tool for figuring out what can be achieved. While molecular systems that are Turing-complete have been demonstrated [Winfree, 1996], these systems still cannot achieve some of the feats biology has achieved.

One might think that because a system is Turing-complete, capable of computing "anything," that it can do any arbitrary task. But while it can simulate any digital computational problem, there are many behaviors that are not "computations" in a classical sense, and cannot be directly implemented. Examples include exponential growth and molecular motion relative to a surface.

Passive self-assembly systems cannot implement these behaviors because (a) molecular motion relative to a surface requires a source of fuel that is external to the system, and (b) passive systems

are too slow to assemble exponentially-fast-growing structures. We call these behaviors "energetically incomplete" programmable behaviors. This class of behaviors includes any behavior where a passive physical system simply does not have enough physical energy to perform the specified tasks in the requisite amount of time.

As we will demonstrate and prove, a sufficiently expressive implementation of an "active" molecular self-assembly approach can achieve these behaviors. Using an external source of fuel solves part of the the problem, so the system is not "energetically incomplete." But the programmable system also needs to have sufficient expressive power to achieve the specified behaviors. Perhaps surprisingly, some of these systems do not even require Turing completeness to be sufficiently expressive.

Building on a large variety of work by other scientists in the fields of DNA nanotechnology, chemistry and reconfigurable robotics, this thesis introduces several research contributions in the context of active self-assembly.

We show that simple primitives such as insertion and deletion are able to generate complex and interesting results such as the growth of a linear polymer in logarithmic time and the ability of a linear polymer to treadmill. To this end we developed a formal model for active-self assembly that is directly implementable with DNA molecules. We show that this model is computationally equivalent to a machine capable of producing strings that are stronger than regular languages and, at most, as strong as context-free grammars. This is a great advance in the theory of active self-assembly as prior models were either entirely theoretical or only implementable in the context of macro-scale robotics.

We developed a chain reaction method for the autonomous exponential growth of a linear DNA polymer. Our method is based on the insertion of molecules into the assembly, which generates two new insertion sites for every initial one employed. The building of a line in logarithmic time is a first step toward building a shape in logarithmic time. We demonstrate the first construction of a synthetic linear polymer that grows exponentially fast via insertion. We show that monomer molecules are converted into the polymer in logarithmic time via spectrofluorimetry and gel electrophoresis experiments. We also demonstrate the division of these polymers via the addition of a single DNA complex that competes with the insertion mechanism. This shows the growth of a

population of polymers in logarithmic time. We characterize the DNA insertion mechanism that we utilize in Chapter 4. We experimentally demonstrate that we can control the kinetics of this reaction over at least seven orders of magnitude, by programming the sequences of DNA that initiate the reaction.

In addition, we review co-authored work on programming molecular robots using prescriptive landscapes of DNA origami; this was the first microscopic demonstration of programming a molecular robot to walk on a 2-dimensional surface. We developed a snapshot method for imaging these random walking molecular robots and a CAPTCHA-like analysis method for difficult-to-interpret imaging data.

Contents

List of Figures

Chapter 1

Introduction

> science is the same as poetry
> only it uses the wrong words.
>
> Science, Robert Kelly (in May Day)

A sequence of DNA or a string of code is a line of poetry. These structures share a common trait: the ability of the individual units to transcend their symbolic nature. These seemingly unrelated fields share a profound emphasis upon sequences of information. This thesis is motivated by a deep philosophical question that underlies all of biology: how is it that a biological system derives all of its complexity from a simple sequence? While this is a very large question, we might approach it by trying to reason as to how information is transformed into an active entity in organic molecules.

1.1 From Information to Activation

A human's genome consists of approximately three billion base pairs [Venter et al., 2001]. This implies that the program our cells are running, which gives rise to our very complicated bodies and brains, utilizes less than two gigabytes of information[1]. There is a long history of great minds

[1]To calculate this number, first observe that each DNA base requires two bits to encode ($00, 01, 10, 11$ map to A, C, G, T). Thus, one byte is equivalent to a sequence of four bases. The genome consists of three billion base pairs or 6×10^9 bases of DNA. When we divide this number by four bases per byte, the result is 1.5×10^9 bytes or 1.5 gigabytes. Contrast this number to the amount of space one would need to store a low resolution movie – by today's standards that requires at least a few gigabytes of storage space.

having drawn comparisons between machines and man[2], but there are some notable differences.

The computer on which this report was written has about three hundred gigabytes of storage space, but it is not capable of doing many things that we people can do. For example, it cannot grow to five times its initial mass while operating at full capacity[3] and it cannot function if I impale the motherboard with a railroad spike[4]. A human's ability to function despite injury itself pales in comparison to the robustness of other members of the animal kingdom[5]. On the other hand, computers are capable of crunching out computations that are very difficult for a human to do.

Computers are traditionally suited to particular kinds of computation that differ from those encountered in biology. One difference between computer architectures and nature's architectures is the mode of information transfer. Whereas in computers, the information encoded in the program is fundamentally separate from the hardware used to execute it[6], in biology, software and hardware are the same thing. Of interest here are the implications of the unification of "form and function" of information in biology. Every day we encounter examples of phenomena that we are as of yet unable to reproduce in electronics, and we don't yet know the full capabilities of simple organic materials.

In 1953, Watson and Crick solved the structure of DNA and proposed that the double helix suggests a replication mechanism for genetics [Watson and Crick, 1953]. The nucleic acids DNA and RNA epitomize the blurring of form and function at the molecular scale: these molecules store information in their sequences, and they can perform enzymatic reactions such as ligation (attaching two strands to form a longer strand) and cleavage (cutting of one strand into two)[7].

DNA has been used as a material to build computing devices [Adleman, 1994, Winfree, 2000],

[2]In fact, Shannon and McCarthy allude to this in their introduction to *Automata Studies*: "Among the most challenging scientific questions of our time are the corresponding analytic and synthetic problems: How does the brain function? Can we design a machine which will simulate a brain? Speculation on these problems, which can be tracked back many centuries, usually reflects in any period the characteristic machines used" [Shannon et al., 1956].

[3]An average male grows from 13 kg at two years of age to 70 kg at twenty years of age [Kuczmarski et al., 2000].

[4]I refer to the case of Phineas Gage [Harlow, 1999], the railway worker who survived an accident in which a large iron rod was driven all the way through his frontal lobe. Humans are capable of carrying on (almost normally) after sustaining a rather large blow to their motherboards.

[5]To list a few examples: a newt's ability to regenerate its tail, a flatworm's ability to regenerate its head, a starfish's ability to regenerate its entire body from a severed leg [Alvarado and Tsonis, 2006].

[6]In fact, computers can be built from water pipes, wires, vacuum tubes, transistors, and even legos.

[7]All of these abilities have led to theories that nucleic acids may have begun life on Earth [Joyce, 1989].

circuits [Lederman et al., 2006, Seelig et al., 2006, Yin et al., 2008] and self-assembled two and three-dimensional structures [Chen and Seeman, 1991, Douglas et al., 2009, Rothemund, 2006, Yin et al., 2008]. An alternate line of research has utilized nucleic acids to engineer biological circuits [Elowitz and Leibler, 2000, Hasty et al., 2002]. For the most part, these endeavors in molecular programming either translate traditional computational problems into molecules, or hijack a cell's materials to control a behavior. While DNA computers have been proven to be Turing universal [Beaver, 1996, Rothemund, 1996, Smith and Schweitzer, 1996], they cannot compete with their digital counterparts in solving classic computational problems[8]. Meanwhile, the synthetic biology approach does not leave room to discover what molecules are capable of in the absence of four billion years of evolution.

What is the limit of behavioral complexity that can be achieved with a molecule as simple as DNA? We know that in biology, there is some nondeterministic encoding [Metzger et al., 2008] and overlaying of information [Breitbart et al., 1987]. We have encountered operons and regulatory networks [Jacob et al., 1960]. We have some idea that cells are running programs, that these programs are stochastic in nature, and that these systems have evolved out of some random mixture of molecules over a billion years ago. But we do not yet understand the programs that cells are running.

The DNA nanotechnology community has addressed this question by engineering programmable assemblies and motors. DNA has been made to programmably self-assemble [Douglas et al., 2009, Liu et al., 1994, Mao et al., 2000, Rothemund, 2006, Rothemund et al., 2004, Winfree et al., 1998] and disassemble [Yin et al., 2008]. An alternate line of research inspired by cellular machinery has resulted in the construction of several nucleic acid based motors [Bath and Turberfield, 2007]. The simplest actuators made of DNA are molecular switches that toggle between two or more conformations [Mao et al., 1999, Simmel and Yurke, 2002, Yurke et al., 2000]. DNA walkers can move relative to a track [Bath et al., 2005, Omabegho et al., 2009, Pei et al., 2006, Sherman and Seeman, 2004, Shin and Pierce, 2004, Tian et al., 2005, Yin et al., 2008], and can generate

[8] As Soloveichik, et al. eloquently phrase the issue, "... shoehorning the design of synthetic chemical circuits into familiar but possibly inappropriate computing models may not capture the natural potential and limitations of the chemical substrate" [Soloveichik et al., 2010].

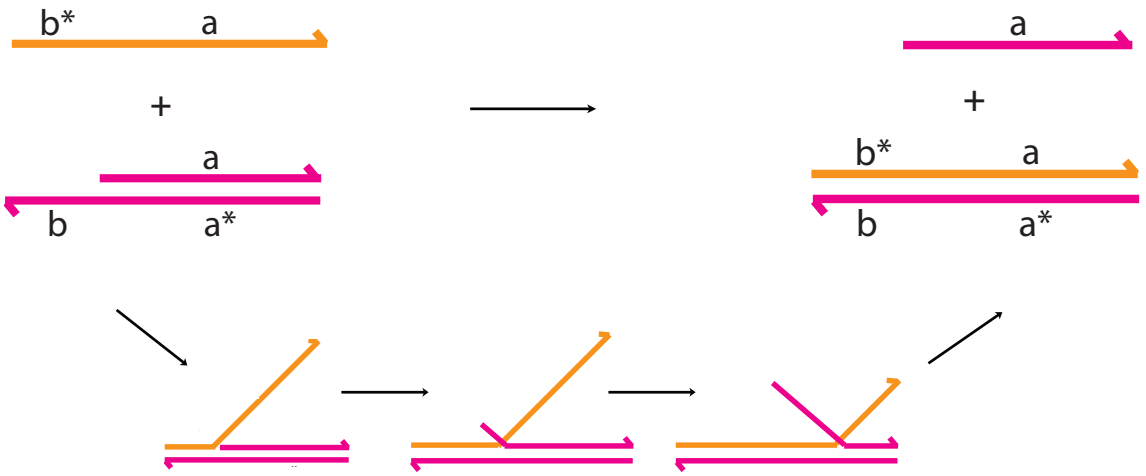


Figure 1.1: A schematic of three-way branch migration.

directional motion without tracks [Venkataraman et al., 2007]. In addition, DNA devices can coordinate two moving parts [Green et al., 2008] and change shape [Andersen et al., 2009, Lubrich et al., 2008, Yurke et al., 2000].

This thesis attempts to bridge the self-assembling and dynamic / motor trajectories of DNA nanotechnology by utilizing insertion and deletion primitives to actively grow and shrink assemblies and to further the goal of programming molecular components that can output complex behaviors like those we see in life.

1.2 Review of Fundamental Components

1.2.1 Three-way versus Four-way Branch Migration

In three-way branch migration only one strand in one duplex is traded for a single strand with the same sequence (Figure 1.1). Four-way branch migration is the process by which two double-stranded oligonucleotides that share the same stem sequence simultaneously exchange strands (Figure 1.2).

Many developments in DNA nanotechnology rely on three-way branch migration to implement

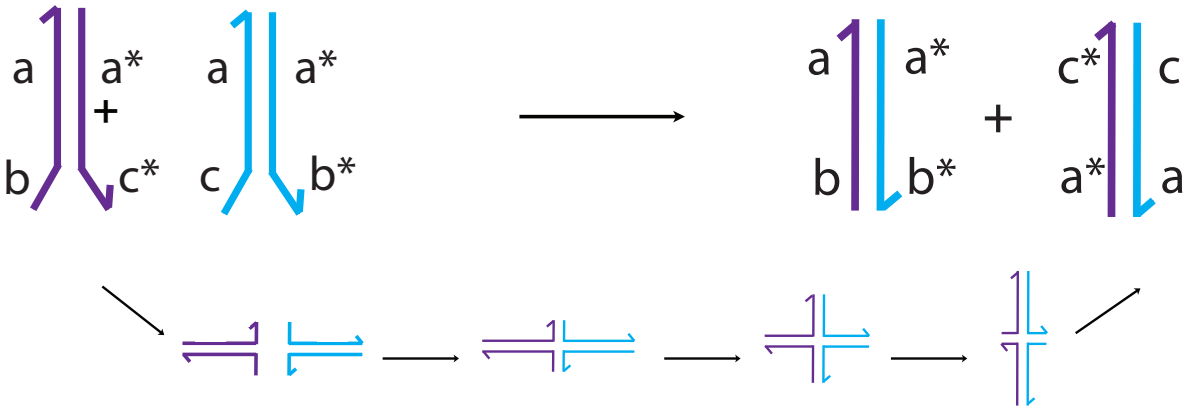


Figure 1.2: A schematic of four-way branch migration.

switches [Lubrich et al., 2008, Simmel and Yurke, 2002, Yurke et al., 2000], circuits [Seelig et al., 2006, Yin et al., 2008], motors [Omabegho et al., 2009, Shin and Pierce, 2004, Yin et al., 2008], assembly [Yin et al., 2008] and amplification [Dirks and Pierce, 2004]. Three-way branch migration is easier to initiate than four-way branch migration (owing to the entropic penalty of bringing two complexes together), and three-way branch reaction rates can be controlled over six orders of magnitude [Zhang and Winfree, 2009]. But it appears that four-way branch migration may give us finer control over this rate, and greater range (seven or more orders of magnitude) [Dabby et al., 2013]. Four-way branch migration has been used to perform directional motion via insertion [Venkataraman et al., 2007]. The capabilities of four-way branch migration have not been fully explored.

1.2.2 Hybridization Chain Reaction

Dirks and Pierce make use of three-way branch migration in their Hybridization Chain Reaction construction [Dirks and Pierce, 2004]. Their construction, which triggers the polymerization of DNA monomers, uses two single-stranded DNA hairpins that have the same 18 base-pair stem sequence and one toehold that is complementary to the other hairpin's loop sequence. These hairpins are caught in a kinetic trap that causes them to react with each other very slowly in the absence of an initiator strand. The initiator consists of a toehold, which is complementary to one hairpin's

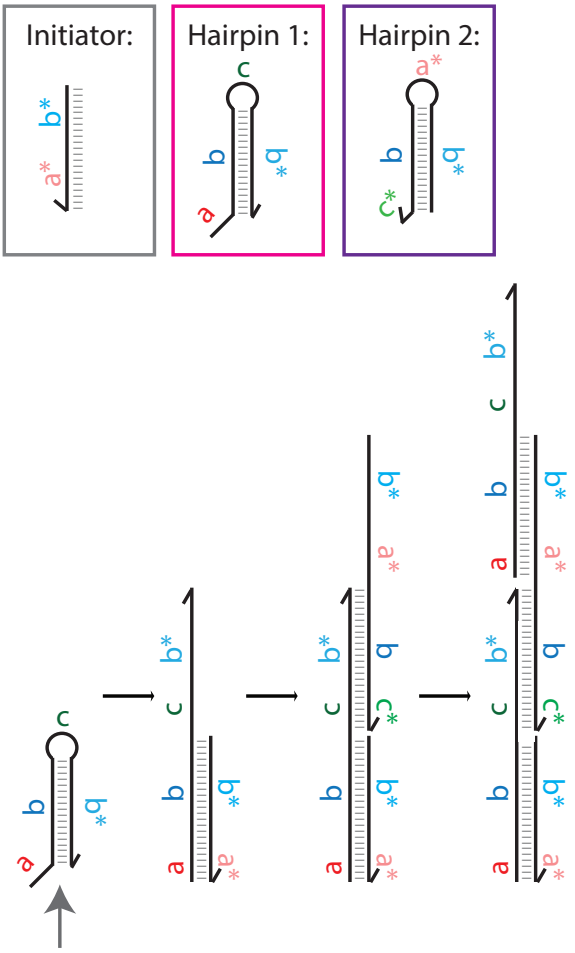


Figure 1.3: A schematic of the Hybridization Chain Reaction as described by [Dirks and Pierce, 2004].

toehold, and its adjacent stem sequence. When the initiator is added to the solution of monomers, it binds to the toehold of the first hairpin and undergoes a strand displacement reaction that opens the hairpin. The newly exposed sticky end of the hairpin can then undergo a similar reaction with the second hairpin.

The two hairpins will continue to polymerize until an equilibrium concentration of monomers is reached. While the HCR system has been applied to the amplification of nucleic acid probes [Choi, 2009], the system was modified to employ a four-way branch migration design to create an autonomous polymerization motor [Venkataraman et al., 2007]. The metastable fuel hairpins from

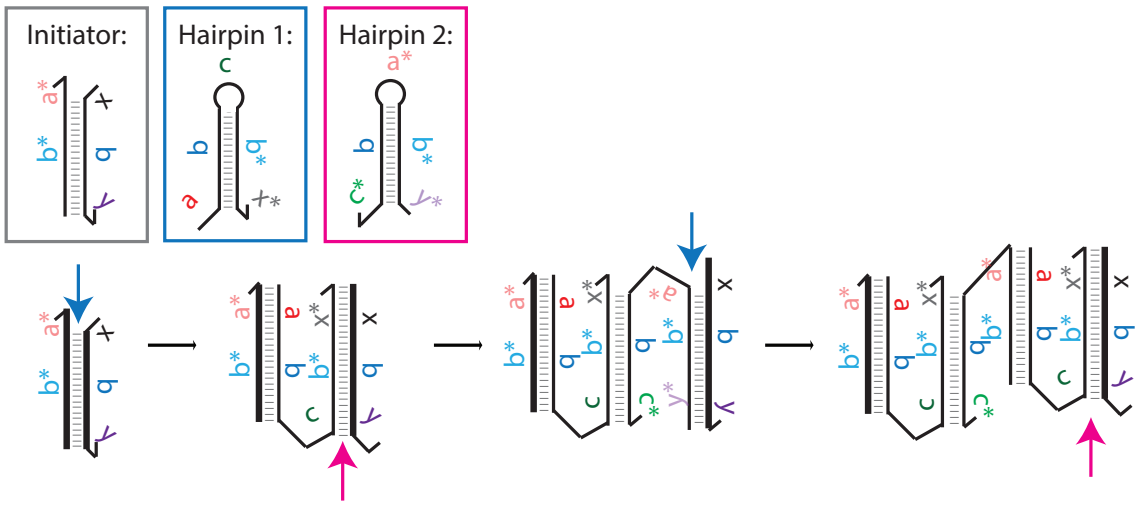


Figure 1.4: A schematic of a Rickettsia-like autonomous polymerization motor as described by [Venkataraman et al., 2007].

the HCR system were modified to include an extra toehold, and the initiator strand was replaced by an initiator complex that is composed of an "anchor" strand and a "rickettsia" strand. Upon mixing, the first hairpin binds to the sticky ends of the anchor-rickettsia complex, initiating a four-way branch migration in which the rickettsia strand is passed from the anchor to the hairpin. The second hairpin then binds to the newly exposed sticky ends and the rickettsia strand is passed to the second hairpin. The rickettsia strand continues to be passed back and forth between newly added hairpins as the polymer grows in its wake.

1.2.3 Other Four-way Branch Machines

Four-way branch migration machines make use of duplex DNA's ability to undergo genetic recombination via a Holliday Junction, a four-way branched DNA structure in which the opposite arms share common sequences. The junction can migrate along a duplex of DNA by breaking the basepairs in one pair of opposite arms and forming base pairs in the other pair [Holliday, 1964]. One of the first DNA nanomachines that made use of the Holliday Junction worked by converting torsional strain into the linear motion of a Holliday junction [Yang et al., 1998]. The device, which consisted of a Holliday junction connected on opposite arms by a closed loop of double

stranded DNA, was powered by the addition and removal of ethidium bromide (an intercalating dye that binds DNA between adjacent base pairs). Since then, the Holliday Junction has been employed in constructing a single molecule "nanometronome" [Buranachai et al., 2006], and in the design of a single molecule switch that can detect single nucleotide mismatches in RNA and DNA oligonucleotides [Buck et al., 2007].

1.2.4 Programming Biomolecular Pathways

Yin, et al. demonstrated a method for programming the pathway by which DNA self-assembles and disassembles that allows one to design a sequence of reactions that can implement dynamic functionality [Yin et al., 2008]. Molecules are initially trapped in a metastable state that allows other oligonucleotides to systematically catalyze their interactions with each other. The examples are implemented using three-way branch migration.

1.3 Related Work

1.3.1 Theory of Active Self-Assembly

One of the central questions that this work addresses is how to program global tasks through local interactions. Graph grammars [Klavins et al., 2004] allow for a systematic way to program molecular self-assembly using rule sets to synthesize a general graph. The approach taken by Klavins, et al. lacked a geometrical component, which we tried to address in our prior work on active self-assembly, and in the process we proved some very interesting theorems that highlight the power that can be gained from actively assembling units over passively assembling units.

The Nubot model [Woods et al., 2013] builds on the concept of graph grammars, by defining rule sets over two dimensional monomers (that we represent as disks of unit diameter centered on a point in a triangular grid). Two monomers can react with each other (according to a rule) to change state, make and break bonds, change relative position, appear and disappear.

Using this model, we showed that a line of length n can be constructed with $O(\log n)$ states, in $O(\log^2 n)$ time. An arbitrary two-dimensional geometric shape with n pixels can be constructed

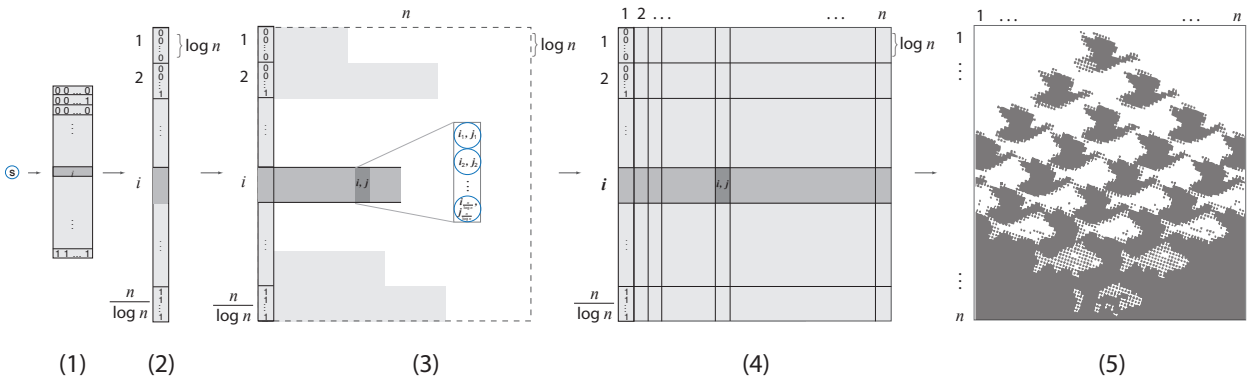


Figure 1.5: An overview of a how a pattern is computed in the nubot model: The shape begins with an initial seed that generates a vertical line of n nubots with unique states in logarithmic time. Each of these nubots then grows a horizontal line of length n, thus generating an $n \times n$ square. Each $\dfrac{1}{\log(n)} \times n$ strip of nubots acts as the input tape to a turing machine, which colors the pixels (nubots) either black or white depending on its unique address in the total assembly.

in $O(\log^2 n + \log n \times f(n))$ time using $O(\log n + g(n))$ states, where *f(n)* is the time required for a size *g(n)* Turing machine to determine if a given pixel is in the shape. It works as follows: the shape begins with an initial seed that generates a vertical line of n nubots with unique states in logarithmic time. Each of these nubots then grows a horizontal line of length n, thus generating an $n \times n$ square. Each $\dfrac{1}{\log(n)} \times n$ strip of nubots acts as the input tape to a turing machine, which colors the pixels (nubots) either black or white depending on its unique address in the total assembly. For many common shapes, *f(n)* and *g(n)* are polylogarithmic in *n*. This is exponentially faster than systems composed entirely of passive components (e.g. tiles).

The main limitation of the Nubot model is that it requires individual monomers to have Turing machine capability to perform the above complex tasks. Molecules are not individually capable of having large state spaces with large look-up tables, thus the Nubot model is not chemically implementable. In order to implement such a system today, one would require macro scale robots with relatively large onboard memories and complex actuation capability.

While the Nubot model is not chemically implementable, it motivates experimental efforts to construct systems with actively assembling components. In fact, it inspired Chapters 3 and 4 of this manuscript. In our ongoing development of a new model for active assembly, we seek to preserve the complex behaviors that our abstract system is capable of, but in a formulation that is simple

enough to implement experimentally.

1.4 Summary of Thesis

1.4.1 Programming Molecular Robots Using Prescriptive Landscapes

Taking some cues from Brooks's "Intelligence without Representation" [Brooks, 1991], we considered how we might imbue molecules with complex programs given their limited encoding space. We hypothesized that molecules might be capable of being programmed by their interaction with a surface. In this way one can program "local rules" such that the configuration and location of a molecule will determine the next step that the molecule can carry out. To this end we explored the ability of a pre-programmed surface to direct the behavior of a "molecular robot" [Lund et al., 2010, Lund, 2008] in Chapter 2.

1.4.2 A Model for Active Self-Assembly in DNA Systems

In Chapter 3 we describe a formal model for studying the complexity of self-assembled structures with active molecular components. In particular, we add an insertion primitive and we show a direct mapping of our model to a molecular implementation using DNA. We show that the expressive power of this language is stronger than regular languages, but at most as strong as context-free grammars. Here, we explore the trade-off between the complexity of the system (in terms of the number of unit types), and the behavior of the system and speed of its assembly. We find that we can grow a line of any given length n in expected time $O(log^3 n)$ using $O(log^2 n)$ monomers. If we grow a line with k insertion rules, either the expected final length is infinite or the expected length at time t is at most $(2t + 2)^{k^2}$, which is polynomial in t.

1.4.3 A Synthetic Polymer that Grows Exponentially Fast

In Chapter 4, we demonstrate the growth of a linear DNA polymer exponentially fast using a molecular insertion primitive to deterministically incorporate three hairpins into a linear structure.

We experimentally verify the exponential kinetics of the system and compare it to its linear counterpart. In addition, we implement a division primitive and show that we can initiate the exponential growth of populations of smaller sized polymers. Lastly we present a theoretical implementation of a treadmilling behavior using these two primitives from our model.

1.4.4 The Kinetics of Toehold-Mediated Four-way Branch Migration

Chapter 5 explores the kinetics of the insertion mechanism implemented in the prior chapter. We characterize the kinetics of toehold-mediated four-way branch migration. We found that by designing the toeholds that initiate the reaction, we can control the reaction rate over at least seven orders of magnitude. We propose a model for the design of the sequences of a four-way branch reaction that operates with desired kinetics. The ability to control the kinetics of these reactions should greatly facilitate the programming of dynamic behaviors mediated by four-way branch migration.

Chapter 2

Molecular Robots Guided by Prescriptive Landscapes[0]

Traditional robots [Siegwart, 2004] rely for their function on computing to store internal representations of their goals and environment and to coordinate sensing and any actuating of components required in response. Moving robotics to the single-molecule level is possible in principle, but faces the limited ability of individual molecules to store complex information and programs. One strategy to overcome this problem is to use systems that can obtain complex behavior from the interaction of simple robots with their environment [Braitenberg, 1984, Brooks, 1991, Turing, 1936]. First steps in this direction were the development of DNA walkers [Bath and Turberfield, 2007], which have transitioned from being non-autonomous [Sherman and Seeman, 2004, Shin and Pierce, 2004], to being capable of directed but brief motion on one-dimensional tracks [Bath et al., 2005, Tian et al., 2005, Yin et al., 2008, Omabegho et al., 2009]. Here, we demonstrate that previously developed random walkers [Pei et al., 2006], so-called molecular spiders that comprise a streptavidin molecule as inert body and three deoxyribozymes as catalytic legs, exhibit elementary robotic behavior when interacting with a precisely defined environment. Single-molecule

[0]This work was coauthored by Kyle Lund, Anthony J. Manzo, Nadine Dabby, Nicole Michelotti, Alexander Johnson-Buck, Jeanette Nangreave, Steven Taylor, Renjun Pei, Milan N. Stojanovic*, Nils G. Walter*, Erik Winfree*, & Hao Yan* and published in 2010 [Lund et al., 2010] with the following contributions: AFM experiments were performed by K.L. (majority), J.N., and N. D.; analysis was performed by N. D., K.L., J.N., S.T., and supervised by E.W., and H.Y. Fluorescence microscopy and particle tracking analysis were performed by A.J.M., N.M., and A.J.B, supervised by N. G. W. Spiders were synthesized, purified, and their integrity confirmed and monitored by S.T. SPR experiments were performed by R. P. Research coordination by M.N.S., material transfer coordination by S.T., J.N., and K.L. Experimental design and manuscript was done with input from all authors.

microscopy observations confirm that such walkers achieve directional movement by sensing and modifying tracks of substrate molecules laid out, much like bread crumbs, on a two-dimensional DNA origami landscape [Rothemund, 2006]. When using appropriately designed DNA origami, the molecular spider robots autonomously carry out sequences of actions such as "start", "follow", "turn", and "stop". We anticipate that this strategy can result in more complex robotic behaviour at the molecular level, by incorporating additional control mechanisms. One example might be interactions between multiple molecular robots leading to collective behaviour [Bonabeau et al., 1999, Rus et al., 2002], another the ability to read and transform secondary cues on the DNA origami landscape as a means for implementing Turing-universal algorithmic behaviour [Turing, 1936, Von Neumann and Burks, 1966, Bennett, 1982].

We previously described polycatalytic assemblies, comprised of steptavidin molecules and several attached nucleic acid catalysts, that function as walkers and are referred to as molecular spiders [Pei et al., 2006]. The molecular spiders used in this study comprise one streptavidin molecule as inert body and three catalytic "legs". Legs are adapted from DNA enzyme 8-17 that binds and cleaves oligodeoxynucleotide (henceforth "oligonucleotide") substrates with a single ribose moiety (Fig. 2.1) into two shorter products that have lower affinities for the enzyme [Santoro and Joyce, 1997]. The different substrate and product affinities ensure that a spider's interactions with a layer of immobilized substrate and/or product sites can be modeled using a simple memory principle [Antal, 2007]: each leg moves independently from sites to accessible neighboring sites, but if a leg is on a site not visited before, it will stay longer on average. Put biochemically: a deoxyribozyme attached to a site that was previously converted to a product will dissociate faster, whereas it will stick longer to substrates and eventually cleave them. Because spiders have multiple legs that prevent complete dissociation, a single dissociated leg will quickly reattach to a nearby product or substrate site. After cleaving, each leg will thus explore neighbouring sites until it finds another substrate to bind to for longer. This ensures that the body of a spider positioned at the interface between products and substrates will move toward the substrate region, and that it will move directionally along a linear track as the substrates are cleaved. Previously engineered walkers using "burnt bridge" mechanisms [Sherman and Seeman, 2004, Bath et al., 2005, Tian et al., 2005, Omabegho et al., 2009] and Brownian ratchets found in nature [Saffarian et al., 2004] ren-

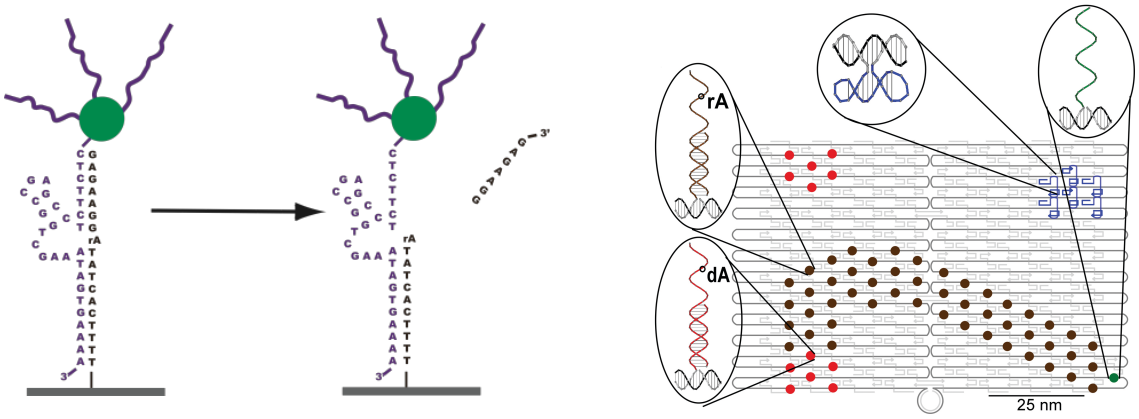


Figure 2.1: Deoxyribozyme based molecular walker and origami prescriptive landscape schematics. The NICK$_{3.4A3+1}$ spider consists of a streptavidin core that displays a 20 base ssDNA that positions the spider at the start (green), and three deoxyribozyme legs. The 8-17 deoxyribozyme cleaves its substrate at an RNA base creating two shorter products (seven and eleven bases). Dissociation from these products allows legs to associate with the next substrate. Spider actions: after release by a 27-base ssDNA trigger, the spider follows the substrate track, turns, and continues to a stop site (red). Schematic of the DNA origami landscape with positions A-E labeled; track EABD is shown. A representative origami landscape shows the start position (green), the substrate track (brown), stop and control sites (red), and a topographical marker (blue).

der revisiting the same path impossible, but our spiders will perform random Brownian walks on product sites until they again encounter a substrate track.

In analogy to the reactive planning used in simple robots [Brooks, 1991], the sensor-actuator feedback afforded when legs sense and modify nearby oligonucleotides allows us to design prescriptive landscapes that direct the spider's motion along a predefined path (Fig. 2.1). Prescriptive landscapes were constructed using the DNA origami scaffolding technique [Rothemund, 2006]. The scaffold consists of a 7249-nucleotide single-stranded DNA folded by 202 distinct staple strands into a rectangular shape roughly $65 \times 90 \times 2$ nm in size and with 6-nm feature resolution (Fig. 2.1). Each staple can be extended on its $5'$ end with probes that recruit substrates, products, goal and control DNA strands [Ke et al., 2008]. We designed pseudo-one dimensional tracks on origami of about spider width (three adjacent rows of substrates, Fig. 2.1). Tracks are coded by a sequence of points (A, B, C, D, E; i.e., on an ABD landscape the spider starts at A, and passes through B before ending at D). Staples were modified to position: (1) A START

oligonucleotide, used to position a spider at the start of the experiment, that is complementary to a TRIGGER oligonucleotide used to release the spider [Yurke et al., 2000] (the "start" action); (2) Substrate TRACK probes to capture the $5'$ extension on substrates forming the TRACK (directing the "follow" and "turn" actions); (3) STOP probes complementary to the $5'$ extension on STOP strands (non-chimeric and uncleavable analogs of the substrate) that do not influence directional movement but trap spiders to prevent them from walking backwards after completing the track (the "stop" action); (4) CONTROL probes (identical to the STOP, but disconnected from the track), used to assess the extent to which free-floating spiders are captured directly from solution; and (5) MARKER oligonucleotides based on inert dumbbell hairpins, aiding in origami classification within atomic force microscopy (AFM) images (Fig. 2.1). To position spiders at START sites, we replaced one of the four catalytic legs of the $\text{NICK}_{4.4A12}$ spider with a tethering oligonucleotide (Supplementary Information) partially complementary to the START oligonucleotide.

To estimate the efficiency of spider motion directed by the TRACK, we defined and tested four paths with no (EAC), one (ABD), or two (EABD, EABC) turns (Fig. 2.2). The basic experimental procedure involves: (1) Assembling the origami; (2) attaching spiders to START sites; (3) adding TRACK, STOP, and CONTROL strands to complete the landscape; and (4) initiating an experiment by releasing spiders through addition of TRIGGER and 1 mM Zn^{++} cofactor [Li et al., 2000].We sampled the origami solution before and after spider release, and imaged individual samples by AFM to determine the locations of spiders. We scored only "face-up" origami (substrates projected away from mica) to avoid artifacts, using procedures that minimize readout bias (see Supplementary Information for details). In all samples imaged before spider release, 30-40% of the assembled origami carry at least one spider, 80-95% of which are singly occupied, and of these 80-90% bound their spider at the START position.Upon adding trigger, all four landscapes with substrate tracks showed that the fraction of spiders at the START diminishes with a concomitant increase in spiders observed on the STOP sites (Fig. 2.2 c,g). A spider's ability to reach the STOP sites decreased with increased TRACK length and with decreased time of incubation in solution. In time-lapse experiments on a long path (EABD, spanning 90 nm) we observed a gradual increase of up to 70% of spiders on STOP sites within 60 minutes (Fig. 2.2 c,g). A short path (ABD, 48 nm) was completed to the same extent within 30 minutes. We captured one series of AFM images

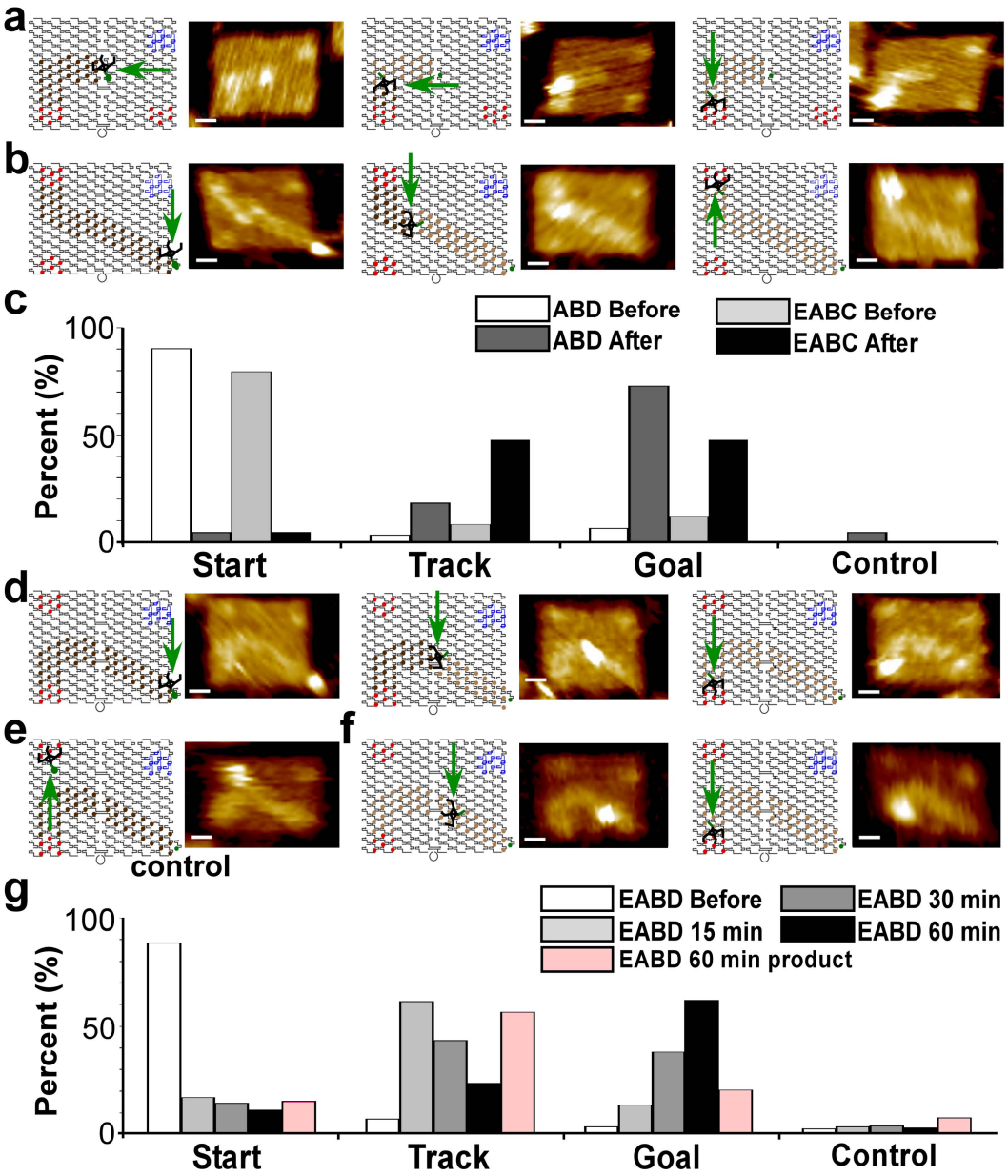


Figure 2.2: Results of spider movement along three tracks with schematics and AFM images of the spider at the start, on the track, and at the stop site. a, ABD track. b, EABC track. c, Graph of ABD and EABC spider statistics before and 30 minutes after release. d, EABD track. e, EABD track with spider on control. f, EABD product-only track. g, Graph of the EABD spider statistics before, and 15, 30 and 60 minutes after release, and 60 minutes after release on the EABD product-only track. All AFM images are 144×99.7 nm, the scale bar is 20 nm. Legend text indicates the number of origami with a single spider that were counted for the given sample.

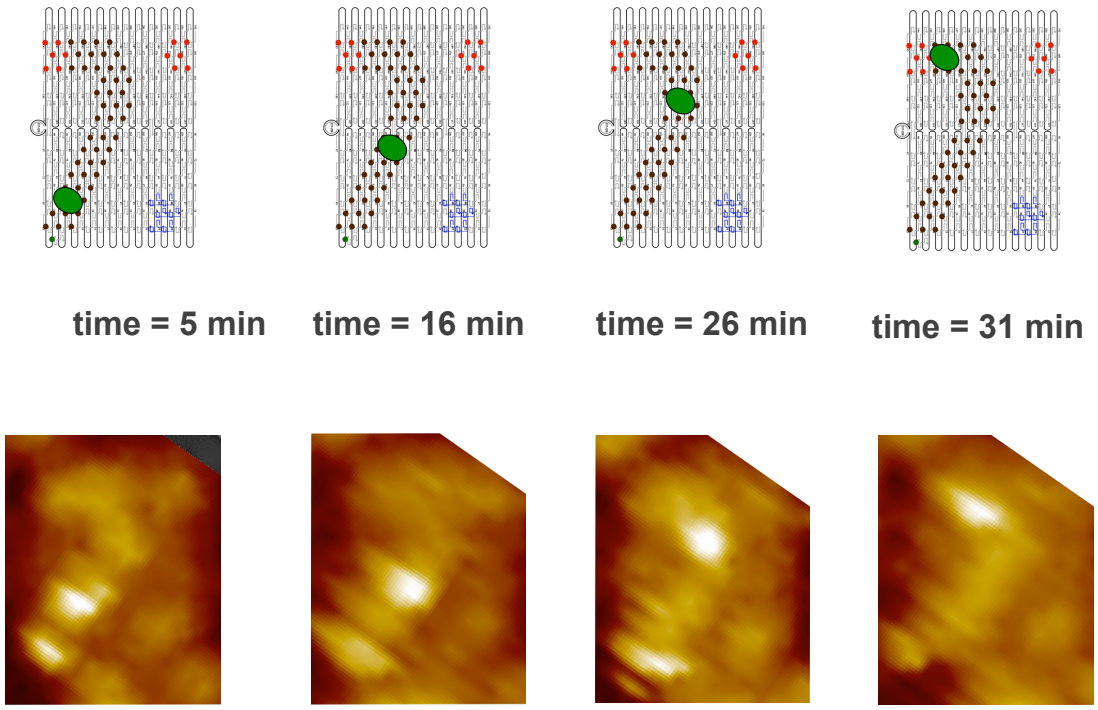


Figure 2.3: Schematics and AFM images of the spider moving along a track at 5 minutes, 16 minutes, 26 minutes, and 31 minutes after trigger was added. AFM images are 300×300 nm and the scale bar is 100 nm.

of a spider moving along an origami track (Fig. 2.3). The rate of spider movement (90 nm over 30 minutes, with approximately 6 nm per three parallel cleavage events) was consistent with the processive cleavage rates (1 min^{-1}) of spiders on a 2D surface as obtained by SPR.

More systematic sequential imaging proved difficult due to micas inhibitory effects on the spider. To test that spiders can indeed traverse product tracks by means of unbiased random walks, we challenged spiders with EABD origami in which the substrate was replaced by product on the TRACK. Spiders still reached the STOP sites albeit more slowly (Fig. 2.2 f,g), as expected from purely Brownian spider movement even if individual steps are somewhat faster [Antal, 2007]. If all

three legs simultaneously dissociate before any leg reattaches, a spider could 'jump' by completely dissociating from the origami and subsequently reattaching elsewhere at random. Evidence against frequent jumping (or an excess of spiders in solution during the initial assembly stage) comes from the low level of spider occupancy at CONTROL sites in both substrate and product track experiments (Fig. 2.2 c,e,g) and the stable proportions of unoccupied and multiply-occupied origami (both before and after the addition of trigger, 5-10% of origami displayed more than one spider on its track). In contrast, when spiders were released on ABD landscapes with no TRACK strands, after 30 minutes we observed an equal distribution between STOP and CONTROL sites, as expected for a process that involves spider dissociation from and random rebinding to the origami. In independent ensemble experiments using surface plasmon resonance to monitor spider attachment and with a constant flow passing over the surface, up to 15% of spiders dissociate from a non-origami 2D product-covered surface within 60 minutes. When using similar surfaces but covered with substrate, spiders show an average processivity of 200 substrate sites before being removed by flow. Together, the results of our control experiments rule out the possibility that spiders move predominantly by jumping; there is insufficient jumping even on product tracks to explain the $50 - 70\%$ occupation of the STOP sites after walks on ABD, EABC, and EABD substrate tracks.

To observe the movement of individual spiders in real-time, we applied particle tracking by super-resolution total internal reflection fluorescence (TIRF) video microscopy [Walter et al., 2008]. Four biotin molecules were attached to the underside of the origami for immobilization on the avidin-coated quartz slide. Spiders were covalently labeled with on average 2.3 Cy3 fluorophores, and STOP sites were labeled with 6 Cy5 fluorophores. The labeling allowed us to monitor changes in spider position relative to the STOP site by two-color fluorescent particle tracking [Churchman et al., 2005, Yildiz and Selvin, 2005]. In a typical experiment, spider-loaded tracks were incubated with TRIGGER and immobilized on the slide, then Zn^{++} was added to promote spider movement via substrate cleavage. Recognizing that the 8-17 activity depends on buffer conditions [Li et al., 2000], we obtained the best results from SSC or HEPES with increased Zn^{++} concentrations but without Mg^{++}.

Our resolution was not sufficient to reliably detect turns, so we focused on EAC landscapes. Individual particle traces showed a distribution of behaviors that may result from variations across

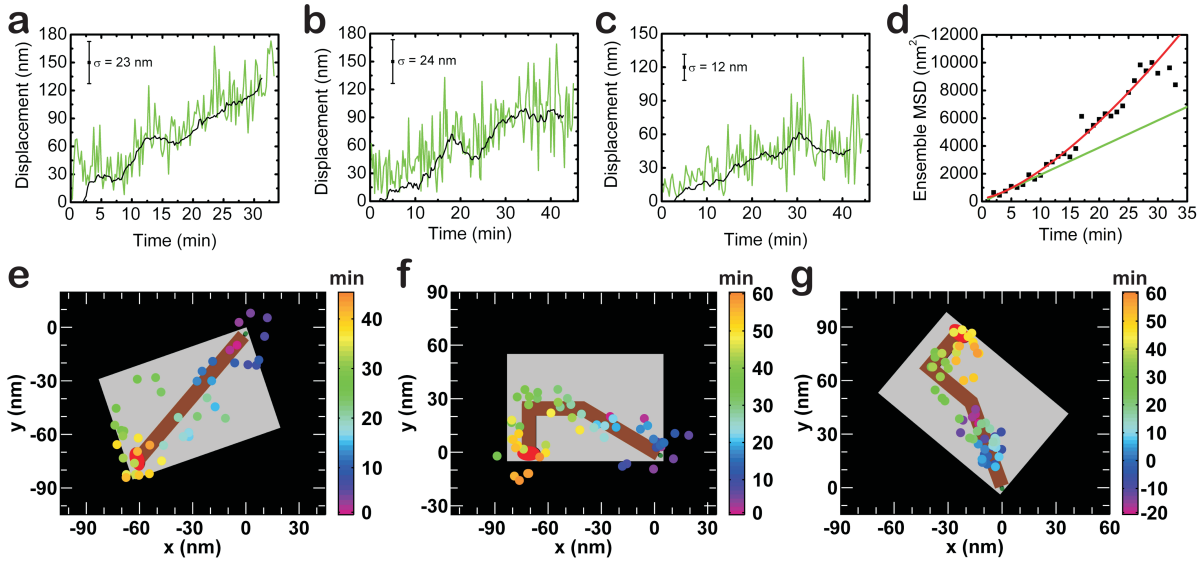


Figure 2.4: Spiders imaged on origami tracks in real-time using super-resolution TIRF microscopy. a, Position-time trajectory of a selected spider (EAC 2, Cy3-labeled) on the EAC substrate track. The position as a function of time is represented by color-coded dots. A small green dot represents the START and a large red oval represents the Cy5-labeled STOP site. ZnSO4 was added at time zero. b, Displacement of the spider trajectory in panel a from its initial position as a function of time. The green line represents displacement calculated using averaged position measurements at 1 minute intervals, and the black line represents the displacement from a rolling 4-minute average. c, Ensemble root mean square displacement (RMSD) of exemplary spiders on the EAC substrate track in the presence (red, 15 spiders) and absence (black, 7 spiders) of Zn^{++}, with the corresponding displacements used to calculate each ensemble RMSD for each buffer condition (similarly colored line graphs). d, Ensemble RMSD for spiders on EAC tracks satisfying simple filtering criteria. Curves are shown for spiders on EAC substrate track (red, 85 spiders), EAC product track with TRIGGER introduced to the sample 10-15 minutes before imaging (blue, 18 spiders), and EAC product track with TRIGGER introduced 30-60 minutes before imaging (black, 29 spiders). EAC substrate and 10-15 minute trigger product RMSD plots are fit to a power law function, and the EAC 30-60 minute trigger product RMSD is fit to a straight line. Individual displacements are shown with colors corresponding to the respective ensemble RMSD plots.

molecules, idiosyncrasies of the sample preparation, the stochastic nature of the observed process, photobleaching, and/or instrument measurement error (Fig. 2.4 a,b). Despite this variability, traces of moving particles commonly exhibited net displacements between 60 and 140 nm and mean velocities between 1 and 6 nm/min; within error, these values are consistent with track length (90 nm) and deoxyribozyme cleavage rate (1 $\text{min}^{-1}\text{leg}^{-1}$), respectively. Tests with and without Zn^{++} and/or TRIGGER, both on substrate and product tracks, yielded RMSD plots of the particle traces that in each case varied as expected based on the behavior of spiders on origami tracks, despite the inherent noise associated with single particle tracking over tens-of-nanometer length scales and tens-of-minute time scales (Fig. 2.4 c,d). For instance, RMSD plots indicated substantially more movement on substrate tracks in the presence of Zn^{++} and TRIGGER than in their individual absence (Fig. 2.4 c). On product tracks, results were consistent with an unbiased random walk independent of Zn^{++}. When product tracks were pre-incubated with TRIGGER 30-60 minutes prior to addition of Zn^{++} and onset of imaging (as were substrate tracks), little or no movement was observed (Fig. 2.4 d), consistent with spiders having been released and moved toward or to the STOP sites prior to imaging. In contrast, when TRIGGER and Zn^{++} were both added shortly prior to imaging, substantial movement was observed (Fig. 2.4 d), consistent with our AFM results for spiders on product tracks (Fig. 2.2 f,g) and with Monte Carlo simulations of spider movement.

The results of our single-molecule experiments are consistent with DNA-based random walkers guided by their landscapes over distances as far as 100 nm, for up to 50 cleavage steps, at speeds of roughly 3 nm/min. We note, however, that the distance over which a spider can move is limited by dissociation and backtracking, so any increase in processivity comes at the cost of a slower velocity [Pei et al., 2006]. Other limitations arise from the mechanism-consuming substrate that must be recharged to sustain directed movement, and from spiders being subject to the stochastic uncertainty as to whether each one can accomplish its task (cf., "faulty" behavior in robotics and "yield" in chemistry). And compared to protein-based walkers using solution phase fuels [Hess, 2006b], our walkers are not as fast, efficient, or powerful. But as candidates for molecular robots, they offer the advantages of programmability [Bath and Turberfield, 2007, Yin et al., 2008, Adleman, 1994, Stojanovic et al., 2003, Seelig et al., 2006], predictable biophysical behaviour [Bath and Turberfield, 2007], and interaction with designable landscapes [Rothemund, 2006].

The ability to obtain programmed behavior from the interaction of simple molecular robots with a complex and adjustable environment suggests that by exploiting stochastic local rules and programming the environment, we can effectively circumvent the limitations that molecular construction places on the complexity of robotic behaviour at the nanoscale.

Chapter 3

A Computational Model for Active Self-Assembly in DNA Systems[0]

3.1 Abstract

In this chapter, we define the first molecularly implementable active self-assembly model. We introduce a theoretical framework for provably knowing what actions, behaviors, and life-like qualities can emerge from a given set of simple modular units. We will use some of the theoretical approaches that computer science has for determining the complexity and difficulty of solving computational problems.

There has been a need for developing formal modular theoretical models for programming active self-assembly processes in both the reconfigurable robotics community and the nanotechnology community. With respect to materials science and nanotechnology, the formal models proposed to date are either not yet implementable with our current understanding of synthetic chemistry or those that are implementable are limited to a set of features that do not capture the power of active components. Prior implementable formal models of molecular assembly only considered the passive behaviors of attaching and detaching from a molecular complex [Winfree,

[0]This work was coauthored by Nadine Dabby & Ho-Lin Chen*, and presented at the Symposium on Discrete Algorithms 2013 [Dabby and Chen, 2013b] with the following contributions: model formulation by N. D., proofs by H-L.C. ; Manuscript was written by both authors.

1996].

3.2 Introduction

In this chapter, we first discuss some of the motivation (Section 3.2.1) and contextual background (Section 3.2.2) for the work, we next review prior self-assembly models and constructions proposed across disciplines (Section 3.2.3), we then present a new "active" self-assembly model that can be directly implemented in molecules (Section 3.3) and we provide a series of theorems and proofs about what these molecules can computationally achieve (Section 3.4). The approach arises out of the fact that molecules do certain things well and other things badly, and digital computers do other types of things well and badly.

As a starting point we note that the Winfree Tile Assembly Model is a "passive" self-assembly system that formally couples computation with shape construction. It is a computational model that can be directly implemented in DNA molecules. Winfree showed that the tiles are capable of universal computation [Winfree, 1996]. Such a system is said to be "Turing-complete".

One might think that because the Tile Assembly Model is Turing-complete, capable of computing "anything," that they can do any arbitrary task. But while they can simulate any digital computational problem, there are many behaviors that are not "computations" in a classical sense, and cannot be directly implemented. Examples include exponential growth, and molecular motion relative to a surface as was discussed in Chapter 2. The tiles cannot implement these behaviors because (a) molecular motion relative to a surface requires a source of fuel that is external to the system, (b) the tiles are too slow to assemble exponentially fast growing structures and (c) the tiles are a passive self-assembling system. We call these behaviors "energetically incomplete" programmable behaviors. This class of behaviors includes any behavior where a passive physical system simply does not have enough physical energy to perform the specified tasks in the requisite amount of time. The tile computation is finished when the system passively reaches equilibrium, which is a slow process. In order to achieve these "active" behaviors, the system will need a fuel source and a logic for how it will grow or move. We will show that it does not need Turing-completeness.

As we will demonstrate and prove in this chapter, a sufficiently expressive implementation of an "active" molecular self-assembly approach can achieve these behaviors. Using an external source of fuel solves part of the the problem, so the system is not "energetically incomplete." But the programmable system needs to have sufficient expressive power to achieve the specified behaviors. Perhaps surprisingly, some of these systems do not even require Turing completeness to be sufficiently expressive.

In this chapter, motivated by some of the ideas from Chapter 1, we present a molecularly implementable idea for "active" self-assembly that exhibits behaviors such as exponential growth. To do this, we select an implementable subset of the behaviors from the Nubot model from Chapter 1 and add a programmable fuel driven approach. In this way we derive a new type of "active" self-assembly system that can be formally defined and easily implemented in molecules.

As we explained in Chapter 1, the full version of the early model cannot be implemented in molecules because it requires small groups of molecules to have more computational power than they can provide. But the right subset of behaviors (in this case exponential growth) can be programmably implemented and abstracted into a formal model, which allows one to explore the space of what can be provably constructed without entering a lab.

Although the Nubot model itself is Turing-complete, capable of performing any digital computation, the subset we have selected is not Turing-complete. Turing machines can accept or generate recursively enumerable languages. In contrast, we prove that the subset we have selected is capable of generating, at most, context-free languages (Figure 3.1). A Turing machine can rewrite its production rules, our system cannot.

Nonetheless, it is a molecularly implementable model that exhibits some of the targeted physical behaviors like exponential growth, and we prove that it exhibits these behaviors, but our system is not Turing-complete. Instead we prove that our model can generate context-free grammars and that the computational capability of this system is at most equivalent to pushdown automata.

Specifically we will show that given any insertion system (our model) we can generate both regular and context-free languages. We explore the trade-off between the complexity of the system (in terms of the number of unit types), and the behavior of the system and speed of its assembly. We find that we can grow a line of any given length n in expected time $O(log^3 n)$ using $O(log^2 n)$

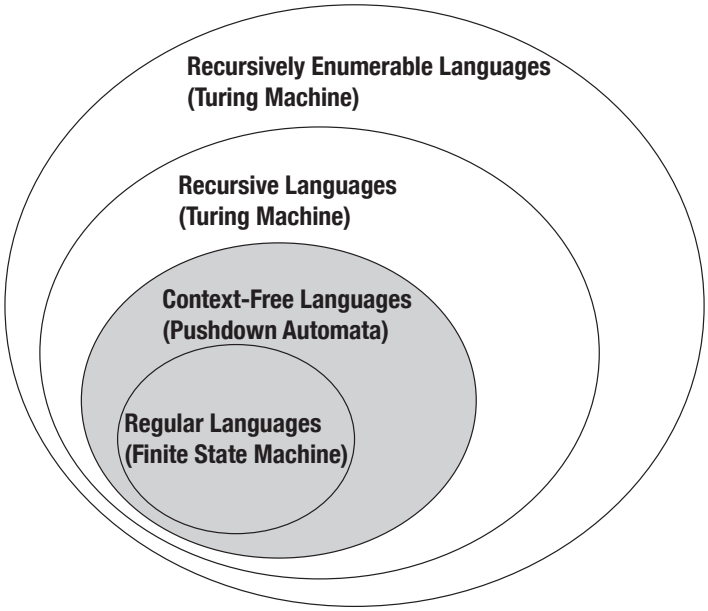


Figure 3.1: Our language is stronger than regular languages. It is within the subset of context-free languages (the shaded area in this figure) as shown in Section 3.4, but it is not as strong as recursively enumerable languages.

monomers. Finally, we show that given any insertion system with with k molecular species, either the expected final length is infinite or the expected length at time t is upper bounded by $(2t+2)^{k^2}$.

Molecular biology is missing a theoretical framework for understanding the complexity of subsets of molecules that interact with each other to generate behaviors. Computer Science has such a framework but it deals on computational complexity – thus we can say how "hard" a particular mathematical problem is by analyzing how much time and space a computer requires to solve it. On the other hand, in other parts of Biology, we can't say how computationally "hard" it is to generate behaviors like metamorphosis (the changing of one shape into another) or treadmilling (the growth of a linear polymer in one direction while it shrinks in the other direction). [1]

In the absence of biological measures of complexity we map our system onto a computational framework, by proving theorems regarding the "expressive power" of the model we define. This shows what the system is capable of doing from a computational perspective. A good primer on computer science theory can be found in [Sipser, 2006]; here we present a very simple summary.

[1] We might use the following measures to distinguish between the hardness of generating different behaviors: number of types of molecules and amount of time necessary to generate the behavior.

3.2.1 Motivation

Molecular programming, nanotechnology and synthetic biology raise the prospect of bottom-up fabrication, the manufacture of complex devices from simple components that assemble themselves. Biology sets the bar high: fabricating systems of enormous scale, defined at atomic-scale resolution, that grow quickly with small programs relative to object size and algorithmic complexity [Karsenti, 2008]. A human's genome consists of approximately three billion base pairs [Venter et al., 2001]; this implies that our cells are running a program that utilizes less than 1 gigabyte of information. Contrast this program-size efficiency with the computer on which we write this report: it has 320 gigabytes of storage disk memory, and yet it is not capable of doing many things that biology can do (e.g. it cannot grow exponentially fast like the embryos shown in Fig. 3.2, it cannot grow in mass and develop simultaneously, and it is not robust to damage). Other examples from biology prove to be even more phenomenologically interesting: a newt is able to regenerate its tail, a flatworm is capable of regenerating its head, and a starfish can regenerate its entire body from a severed leg [Alvarado and Tsonis, 2006]. Biology offers many examples of phenomena that we are as of yet unable to reproduce in computational software or hardware, but that perhaps show us what is possible. Inspired by these feats of biological efficiency, robustness and phenomena, we define a formal implementable model for active self-assembly.

Many attempts have been made to emulate biology's success across materials and disciplines. While biologists have had success reconstructing self-organized cellular systems in vitro [Liu and Fletcher, 2009], chemists have utilized self-assembly to construct films and monolayers as well as more complicated architectures constructed from nanotubes and nanoparticles [Whitesides and Grzybowski, 2002,Grzybowski et al., 2009]. These new self-assembled materials have in turn been used to construct nano-scale electronics [Lu and Lieber, 2007] and biomaterials [Stupp, 2010].

Nanotechnologists have built many examples of self-assembling two and three dimensional devices using passive subunits. The nano-components of a cell are much more "active" than passive: they sense and process environmental cues; they assemble and disassemble; upon interaction, their configurations often change, determining their future interactions; they can both diffuse and actively move. Recently, self-assembly systems using active molecular components have also

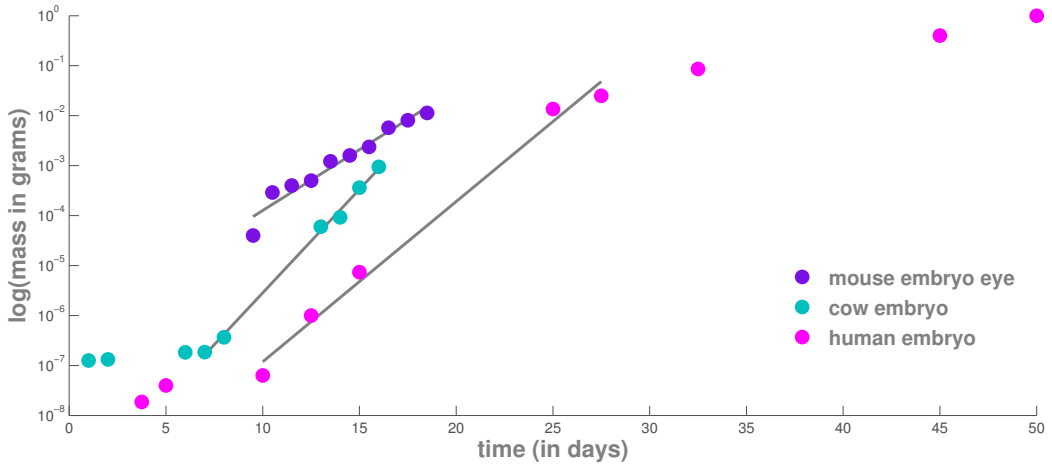


Figure 3.2: This plot is compiled from embryonic mouse [Foster et al., 2003], cow [Morris et al., 2001], and human [Luecke et al., 1999] data. The gray lines fit the periods of exponential growth in each species. Note that beginning points do not reside on these lines, because the growth rate initially increases proportionally to mass. The period of exponential growth slows down as the amount of mass necessary to sustain this type of growth becomes constrained by volume and access.

been demonstrated in various synthetic systems [Kay et al., 2007, Heuvel and Dekker, 2007, Hess, 2006a, Bath and Turberfield, 2007]. Particularly notable are the rich dynamical systems constructed out of synthetic nucleic acids, whose four-base code gives rise to a means of programming specific molecular interactions. DNA has been used to build autonomous walkers [Yin et al., 2004, Tian et al., 2005, Bath et al., 2005, Pei et al., 2006, Green et al., 2008, Omabegho et al., 2009, Yin et al., 2008, Lund et al., 2010, Muscat et al., 2011], logic and catalytic circuits [Seelig et al., 2006, Zhang et al., 2007, Yin et al., 2008, Win and Smolke, 2008], and triggered assembly of linear [Dirks and Pierce, 2004, Venkataraman et al., 2007] and dendritic structures [Yin et al., 2008].

Now that our once passive subunits can actively sense, walk and otherwise actively interact, how do these new "rules" change the prospects for what we can build from the bottom-up? This notion of actively assembling molecules is already an experimental reality, but as of yet there is no satisfying theory to guide future work. In this paper we attempt to formulate a framework for integrating these new "active" mechanisms in nanotechnology to build "programs" that can grow into a desired shape quickly and with relatively small program size to a final structure.

3.2.2 Contextual Background

In computer science, one can divide up the difficulty of a computational problem by classifying the strength of the machine necessary to solve it; these machines are associated with the types of languages they are capable of accepting as input or generating. A language is a set of strings of symbols that can be generated by a set of production rules [Sipser, 2006].

The simplest classes of machines are finite automata. For example one can think of the motion sensing doors in the supermarket– the doors are capable of being in two states, open or closed, and the doors go between these states with the various inputs to the motion sensors. The system has an extremely limited memory. We say a machine "accepts" a language if the machine can take a string from that language as input and end in one of its terminal states. A finite automata is capable of accepting regular languages as input.

The next more complex class of machines are called pushdown automata; these are essentially finite state machines that have an additional stack memory. Pushdown automata are capable of accepting context-free grammars as input to the machine.

Both finite state machines and pushdown automata are less complex than Turing machines. Turing machines have multiple states, memory and the ability to conduct if-then logic. With these key capabilities, Turing machines can accept all unrestricted grammars (grammars that have no restrictions on either the right or left side of the grammar's production rules) and they can generate arbitrary recursively enumerable languages and are capable of solving all computable problems. Our system is not as strong as a Turing machine, rather it can generate languages that are at most equivalent to context-free grammars.

Our theorems and proofs below are derived from our ability to formalize our model into a language that is, at most, as strong as a context-free grammar. A context-free grammar is a class of formal languages that can be generated by production rules. Here we use the formal definition of a context-free grammar taken from [Sipser, 2006]:

A context-free grammar is a 4-tuple (V, Σ, R, S) such that

1. V is a finite set of variables.

2. Σ is a finite set, disjoint from V, of terminals.

3. R is a finite set of rules, where each rule takes a variable and transforms it into a string of variables and terminals.

4. $S \in V$ is the start variable.

3.2.3 Review of Self-Assembly Models

The Tile Assembly Model integrates the algorithmic association of units with a defined geometry: the exposed edges of a growing crystal encode the state information of the system, and this information is modified as a new tile attaches itself to the crystal [Winfree, 1996]. This model formally couples computation with shape construction, and the shape can be viewed as the output of the tile assembly "program". Tiles are capable of universal computation [Winfree, 1996]. The system can grow an arbitrary shape (independent of scale) using a tile program whose complexity, defined as the number of distinct tile species in the program, is bounded from both above and below by the shape's descriptional (Kolmogorov) complexity [Soloveichik and Winfree, 2005]. The time required to build an $n \times n$ shape through passive self-assembly is $O(n)$ [Adleman et al., 2001]. This bound can be improved to $O(n^{4/5}log(n))$ with massive parallelism [Chen and Doty, 2012]. In this model, scale plays the same role in the self-assembly process as time plays in computability. While the Tile Assembly Model is elegant in its simplicity and ability to capture experimental reality, it is limited in its speed, its ability to be scaled up and its focus on passively assembling units.

Drawing on cellular automata and Chomsky grammars, L-systems were developed as a theoretical framework for studying development in multicellular organisms and were one of the first models used to simulate growth and development in plants [Lindenmayer, 1987]. Although they bear a resemblance to cellular automata, they differ in that arrays can grow and shrink (introducing the notions of insertion, a new cell is generated by division of a prior cell, and deletion, the elimination of a cell). L-systems differ from grammars in that they require parallel rewriting of all symbols and do not distinguish between terminal and non-terminal symbols [Lindenmayer, 1987]. While these models are well-developed for one-dimensional systems, they have also been studied in two [Siromoney, 1986] and three dimensions [Prusinkiewicz, 2004]. While L-systems have aided in the modeling of plant growth and biology, the formal work does not address theoretical

questions related to the complexity of pattern formation such as how quickly a system can generate a specific pattern.

A number of geometric models and numerous algorithms have been described for self-assembling and reconfigurable modular robotic systems [Chirikjian, 1993, Goldstein and Mowry, 2004, Rosa et al., 2006, Griffith, 2004, White et al., 2004, Jones and Mataric, 2003, Murata et al., 1994, Nagpal, 2008, Werfel and Nagpal, 2007, Arbuckle and Requicha, 2004, Rus and Vona, 2001, Butler et al., 2001, Yim et al., 1997, Yim et al., 2007, Groß and Dorigo, 2008, Walter et al., 2004]. Existing formal models have not fully captured the efficiency of active self-assembly: to assemble a prescribed shape, most of the models require a linear (to the size of the shape) number of distinct states.

One of the central questions that this work addresses is how to program global tasks through local interactions. Our approach is inspired by Klavins' work on modeling robotic self-assembly [Klavins et al., 2004] using conformational switching [Saitou, 1999] and graph grammars [Ehrig, 1979]. In Klavins' work, the state of a physical system is represented as an abstract graph, where an assembly unit is represented as a symbolic vertex labeled with its current state, and the attachment of two units is represented by an edge connecting the two corresponding vertices in the graph. Assembly proceeds following graph rewriting rules, which update the system state by updating the vertex labels and edges of a subgraph under suitable conditions. This approach nicely captures the local, asynchronous, cooperative and conditional state change logic, which is intrinsic to assembly systems with active components, and it captures disassembly in addition to assembly. However, unlike the Tile Assembly Model, the graph grammar model represents the assembly system as an abstract graph, and leaves out geometry, which is a crucial property for the assembly of physical systems.

In our prior work on active self-assembly, we constructed the "nubot" model by adding a geometric component to the graph grammar model [Woods et al., 2013]. The nubot model builds on the concept of graph grammars, by defining rule sets over two dimensional monomers, represented as disks of unit diameter centered on a point in a hexagonal grid. Two monomers can react with each other (according to a rule) to change state, make and break bonds, change relative position, appear and disappear. With this model, a line of length n can be constructed with $O(\log n)$ states, in $O(\log n)$ time. A computable shape of size $n \times n$ pixels can be constructed in time polyloga-

rithmic in n. This is exponentially faster than systems composed entirely of passive components. While the nubot model is not chemically implementable, it highlights the fundamental efficiency advantage of active self-assembly over passive self-assembly. We seek to preserve the complex behaviors that the abstract nubot model can generate, but in a formulation that is simple enough to implement experimentally.

3.3 Model

3.3.1 Formal Model Description

In our model, each construction begins with an initiator, and grows via the insertion of simple units that we call monomers. We assume that each type of monomer in the system is present in infinite amounts. Monomers can be inserted into the middle of the structure and increase the length of the structure (an abstraction of the model is shown in Fig. 3.3). Figure 3.4 shows an example system that grows exponentially fast. The detailed description of initiators, monomers, and the insertion rules follows:

1. We have two finite sets of symbols $\Gamma = \{a_1, a_2, a_3, a_4, \dots\}$ and $\Gamma^* = \{a_1^*, a_2^*, a_3^*, a_4^*, \dots\}$. Each pair a_i and a_i^* are called *complementary* to each other.

2. There are k monomers, each is described by a quadruple of symbols (a, b, c, d) and either a plus sign or a minus sign. The plus and minus sign indicate the directionality of the molecules and are used in mapping the model onto a direct DNA implementation, which requires both $5'$ and $3'$ sequences. (For example, $(a_4, a_7, a_6^*, a_1)+$ or $(a_5, a_7, a_2^*, a_3^*)-$.) Each monomer has a concentration c. We assume that the total concentration is at most 1.

3. The initial state can be described by two pairs of symbols $(a, b), (c, d)$. Either a and d are complementary to each other or b and c are complementary to each other. Each of these pairs is considered a monomer.

4. An *insertion site* can only exist between two consecutive monomers: e.g., in the initial state (a, b) and (c, d) belong to two different monomers.

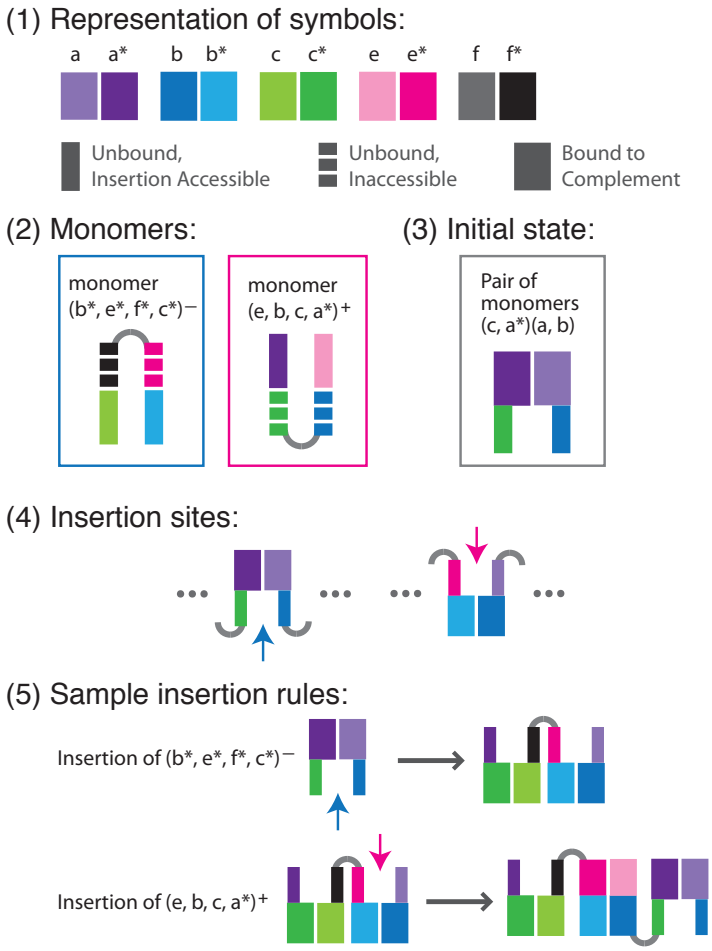


Figure 3.3: This figure depicts an abstraction of our model. (1) Each unique symbol is encoded by a color, and complementary symbols are represented by different shades of the same color. The symbols are represented as thin solid lines (Unbound, Insertion Accessible), thin dashed lines (Unbound, Inaccessible), and thick solid lines (Bound to Complement). (2) Two sample monomers are $(b^*, e^*, f^*, c^*)-$, and $(e, b, c, a^*)+$. (3) The initial state is described by the pair of doubles $(c, a^*), (a, b)$. (4) Insertion sites can only exist between two consecutive monomers connected in the structure; we use colored arrows to denote these sites. (5) Sample insertion rules show the insertion of monomer $(b^*, e^*, f^*, c^*)-$ into $(c, a^*), (a, b)$ to generate the polymer $(c, a^*), (f^*, c^*, b^*, e^*), (a, b)$, and the insertion of monomer $(e, b, c, a^*)+$ into $(c, a^*), (f^*, c^*, b^*, e^*), (a, b)$ to generate the polymer $(c, a^*), (f^*, c^*, b^*, e^*), (e, b, c, a^*), (a, b)$.

5. Only the following insertion rules are possible:

   (a) If there are two consecutive monomers connected in the structure such that the first one ends with the pair (e, a^*) and the second one starts with the pair (d^*, f), where e and f are complementary with each other, then any monomer of the form $(a, b, c, d)+$ can insert between those two groups, and add a group of symbols (a, b, c, d) in the middle. $(e, a^*), (d^*, f)$ is called an *insertion site*.

   (b) If there are two consecutive monomers connected in the structure such that such that the first one ends with (d^*, e) and the second one starts with (f, a^*), where e and f are complementary with each other, then any monomer of the form $(a, b, c, d)-$ can insert between these two groups and add a group of symbols (c, d, a, b) in the middle. $(d^*, e), (f, a^*)$ is called an *insertion site*.

6. If a particular insertion is applicable, it occurs at time x, where x is an exponential random variable with rate c, where c is the concentration of the monomer inserted.

7. A *polymer* is a sequence of tuples of symbols reachable from the initial state, where the first and last tuples are pairs of symbols and the middle tuples are monomers (as defined in rule 2). A *terminal polymer* is a polymer such that no monomers exist in the system that can be inserted at any of the insertion sites available on that polymer. The *length* of the polymer is defined as the number of monomers that it contains.

3.3.2 A Molecular Implementation

Given any system described above, there is a direct implementation of monomers into a set of DNA molecules. By encoding the order of the nucleotides in a DNA sequence, we can control the interaction of DNA strands. Subsequences of these strands are called domains and it is their binding (hybridization) and unbinding (disassociation) from complementary domains that determines what a system can do. In DNA nanotechnology, dynamic systems of DNA molecules can be controlled by toeholds, the short sequences of DNA that are complementary to single stranded

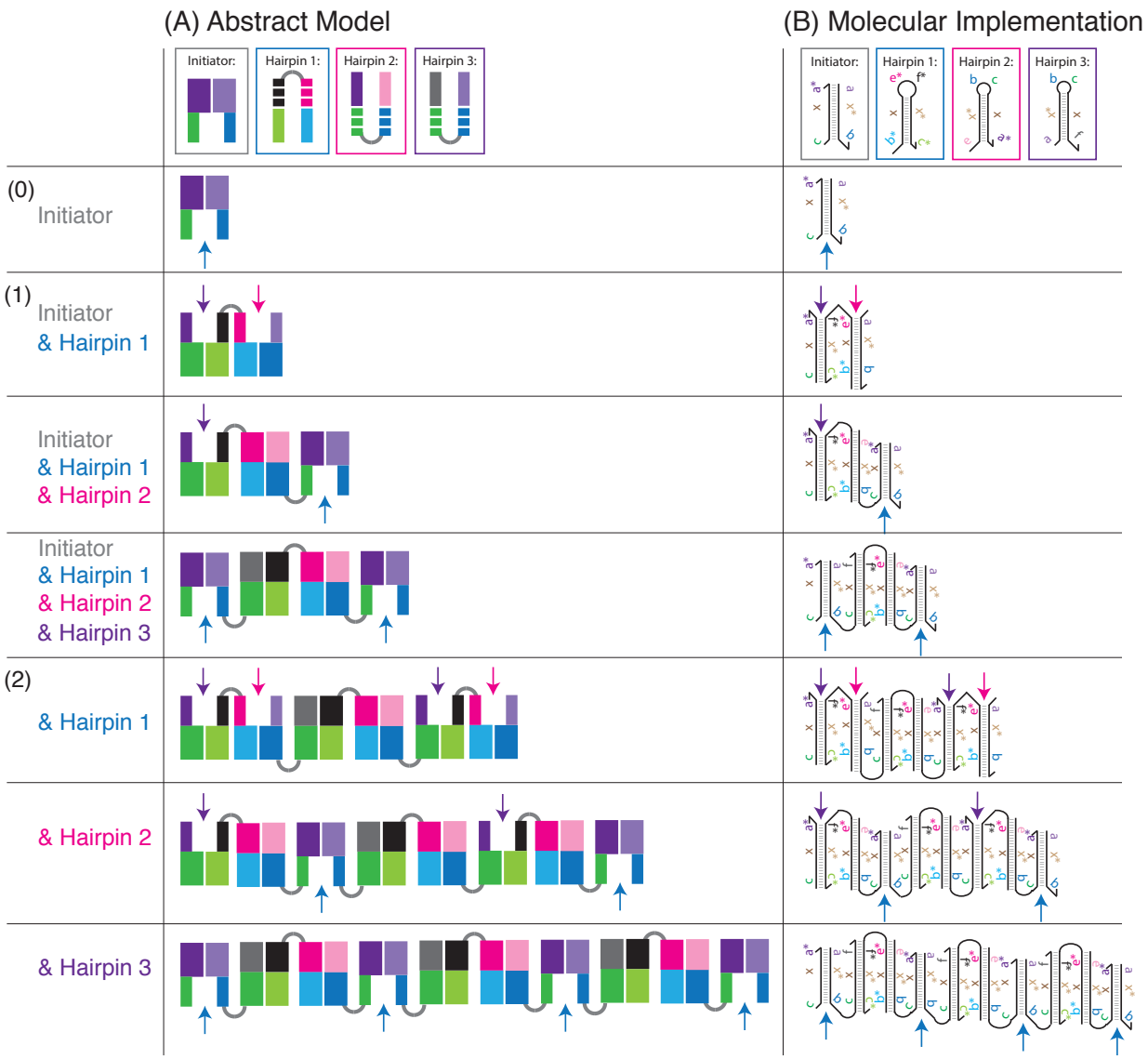


Figure 3.4: This figure depicts a system that implements insertional polymer growth in logarithmic time. The abstract representation of growth (A), is directly correlated to a molecular implementation (B). In this insertion system, the initiator is described as $(c, a^*), (a, b)$ and the three hairpins are $(b^*, e^*, f^*, c^*)-$, $(e, b, c, a^*)+$, and $(a, b, c, f)+$ After inserting hairpin 1, the polymer's description is $(c, a^*), (f^*, c^*, b^*, e^*), (a, b)$. After hairpins 2 and 3 are inserted, the polymer's description is $(c, a^*), (a, b, c, f), (f^*, c^*, b^*, e^*), (e, b, c, a^*), (a, b)$. The system will continue to grow to infinite length exponentially fast.

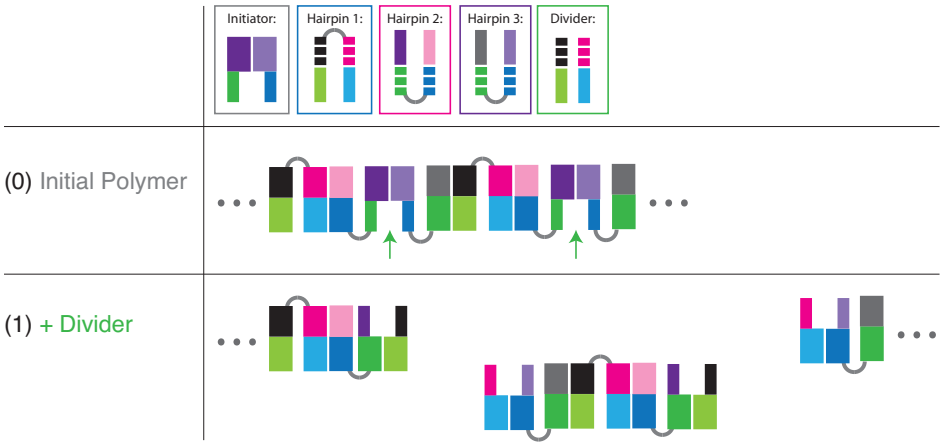


Figure 3.5: This figure depicts a system that implements division in a polymer. The reaction available for $(a, b)(c, d)+$ is exactly the same as that for $(a, b, c, d)+$, except that after $(a, b)(c, d)+$ inserts, the polymer will be cut between (a, b) and (c, d) and divided into two parts.

domains in a target molecule [Yurke et al., 2000, Zhang and Winfree, 2009]. Toeholds serve as the inputs to dynamic DNA systems and initiate branch migration processes, the random walk process of bond breaking and formation that results in the exchange of one strand in the duplex for another single strand with the same sequence. Our DNA implementation (Figure 3.4) is inspired by the Hybridization Chain Reaction system developed by Dirks and Pierce [Dirks and Pierce, 2004] and will be discussed in depth in Chapter 4.

Any system described in our model can be implemented by designing DNA hairpins and an initiator complex as follows:

For every monomer $(a, b, c, d)-$, we add a hairpin with domains (a, x, b, c, x^*, d), where x (composed of 18 bases) is the long stem of the hairpin. For every monomer $(a, b, c, d)+$, we add a hairpin with domains (a, x^*, b, c, x, d). The initiator is (a, x^*, b) binding with (c, x, d). The insertion rules defined in the model correspond to all possible reactions that can happen in the corresponding molecular system. In addition to the monomer $(a, b, c, d)+$ (or minus), we can also have a new type of monomer $(a, b)(c, d)+$ that we call a divider monomer. The reaction available for $(a, b)(c, d)+$ is exactly the same as that for $(a, b, c, d)+$, except that after $(a, b)(c, d)+$ inserts, the polymer will be cut between (a, b) and (c, d) and divided into two parts , as will be described in Chapter 4.

3.4 Proofs of the Model's Expressive Power

In this section, we first ignore the rates of insertion and show that the expressive power of this insertion system is, at most, equivalent to context-free languages. This result implies that we can simulate arbitrary tile systems that assemble a single line. From [Becker et al., 2006], we know that the insertion system can construct lines of arbitrary expected length with $O(1)$ monomers.

Theorem 1 *Given any insertion system, the set of terminal polymers that can be generated forms a context-free language.*

Proof: Given any insertion system with n symbols, the corresponding context-free language has n^4 symbols, each of which corresponds to one insertion site. The starting symbol S corresponds to $(a, b), (c, d)$, which is the initiator of the polymer. Each monomer $(e, f, g, h)+$ corresponds to $2n$ different production rules in the context-free language that starts with a symbol (insertion site) $(i, e^*), (h^*, j)$ and produces two symbols $(i, e^*), (e, f)$ and $(g, h), (h^*, j)$ for all possible choices of pairs of complementary symbols i, j in the insertion system. □

Theorem 2 *Given any regular language, there is an insertion system that generates terminal polymers corresponding to this language.*

Proof: Given any left regular grammar with nonterminal symbols $A_1, A_2, \ldots, A_n$, including the starting symbol A_1, and non-terminal symbols $\alpha_1, \alpha_2, \ldots, \alpha_m$, the following insertion system creates polymers that correspond to the given regular language:

1. $\Gamma = \{a_1, a_2, \ldots, a_n, b_1, b_2, \ldots, b_n, c_1, c_2, \ldots, c_m, d\}$.
2. The initiator is $(d^*, a_1), (d, d)$.
3. For each production rule $A_i \rightarrow \alpha_j A_k$, there are two corresponding monomers $(a_i^*, c_j, b_k, d^*)+$, $(d^*, a_k, d, b_k^*)-$.

In this system, there is always exactly one insertion site available at the end of the polymer. The insertion can only happen between two monomers (d, b_i^*, d^*, a_i) and (d, d). The insertion

site between these two monomers corresponds to the nonterminal symbol A_i. At this point, two monomers $(a_i^*, c_j, b_k, d^*)+$, $(d^*, a_k, d, b_k^*)-$ may insert, generate some inactive sequence with j encoded in the middle, and the end of the polymer becomes (d, b_i^*, d^*, a_k) and (d, d), corresponding to the nonterminal symbol A_k. □

Corollary 1 *There is a family of insertion systems that can construct polymers of expected length n with $O(1)$ monomers.*

Proof: Since insertion systems are able to simulate all regular languages, they are able to simulate all tile systems that form a linear polymer of width 1. Therefore, the proof directly follows from [Becker et al., 2006], where the result was proven on 1-dimensional tile assembly systems. □

3.4.1 Analyzing the Theoretical Growth Speed of Polymers

We also investigate the speed at which these polymers can be constructed. First, we show that arbitrarily long polymers can be constructed deterministically in expected polylogarithmic time using a polylogarithmic number of monomers.

Lemma 3 *The following insertion system deterministically constructs a line of length $n = 2^{2k} + 1$ in expected time $O(\log^3 n)$ and only uses $O(\log^2 n)$ monomers. Furthermore, the required time has an exponentially decaying tail probability.*

1. *The initiator is $(c, a_{2k}), (b_{2k}^*, c^*)$.*
2. *For every $i, j \in \{2, 4, \ldots, 2k\}, i \leq j$, there are two monomers $(a_i^*, b_{i-1}^*, a_{i-1}, b_j)+$ and $(a_j^*, b_{i-1}^*, a_{i-1}, b_i)+$. For every $i \in \{1, 3, \ldots, 2k-1\}$, there are two monomers $(b_i, a_{i-1}, b_{i-1}^*, c^*)-$ and $(c, a_{i-1}, b_{i-1}^*, a_i^*)-$.*
3. *All monomers have equal concentration $\frac{1}{2k^2}$.*

Proof: First, we show that the system deterministically constructs a line of length n. An insertion site of the form $(c, a_i), (b_j^*, c^*)$ is defined to have type $\min\{i, j\}$. Whenever a gap of type i is available, exactly one monomer of the form $(a_i^*, b_{i-1}^*, a_{i-1}, b_j)+$ and $(a_j^*, b_{i-1}^*, a_{i-1}, b_i)+$

will be able to attach. After the first monomer inserts, two monomers $(b_{i-1}, a_{i-2}, b^*_{i-2}, c^*)-$ and $(c, a_{i-2}, b^*_{i-2}, a^*_{i-1})-$ will be able to insert on the first monomer's left and right. These three insertions create four insertion sites of type $i-2$. Therefore, starting with one insertion site of type k on the initiator, there will be $2^k - 1$ total insertions, resulting in a polymer that has n insertion sites of type 0. At that time, no further insertion is available and the system halts.

Second, the system halts as soon as all $\frac{n}{2}$ insertions of $(b_1, a_0, b^*_0, c^*)-$ and $(c, a_0, b^*_0, a^*_1)-$ happen. Each of these insertions only relies on k insertions to occur before them. Therefore, for any one of these insertions, the expected time T until the insertion occurs can be described as a sum of k independent exponential random variables of expected values $2k^2$. Using Chernoff bounds for exponential random variables, it follows that

$$\text{Prob}[T > 2k^2 \cdot k(1+\delta)] \;\leq\; (\frac{1+\delta}{e^\delta})^k.$$

Although the times required for these $\frac{n}{2}$ insertions are not independent of each other, we can still use a union bound to get the following bound for the total running time T_{fin} of the system:

$$\begin{aligned}\text{Prob}[T_{fin} > 2k^2 \cdot k(1+\delta)] \;&\leq\; \frac{n}{2}(\frac{1+\delta}{e^\delta})^k \\ &\leq\; \frac{n}{2}e^{-\frac{k\delta}{2}} \;<\; (\frac{e}{2})^{-\frac{k\delta}{2}}, \text{ for all } \delta > 4.\end{aligned}$$

Therefore, the expected time is $O(k^3) = O(\log^3 n)$ with a tail probability that exponentially decays. □

Theorem 4 *There exists an insertion system that deterministically constructs a line of length n in expected time $O(\log^3 n)$ and only uses $O(\log^2 n)$ monomers for every integer n. Furthermore, the required time has an exponentially decaying tail probability.*

Proof: Lemma 3 already showed that the theorem is true for all $n = 2^{2k} + 1$. Given an arbitrary n, we can write $n - 1$ as the sum of $O(\log n)$ terms $2^{r_1} + 2^{r_2} + \cdots + 2^{r_m}$., where all r_is are even numbers. We can first construct m distinct monomers that must insert one by one at the beginning, creating m insertion sites identical to the initiator for a polymer of length $2^{r_i} + 1$. Afterwards,

the system described in Lemma 3 can make a line of length n. Since m is only $O(\log n)$, this construction works in the required $O(\log^3 n)$ time and $O(\log^2 n)$ monomers. □

In the rest of this section, the major goal is to show that for an insertion system with k different molecular species, either the expected final length is infinite, or the expected length grows polynomially with time.

Theorem 5 *Consider a context-free language L with m symbols (including terminal and nonterminal symbols) in reduced Chomsky normal form. When a production rule is applicable, the time until it is applied is a random variable of rate k. If, for any given symbol A, the rate of all production rules having A on the left side sum up to at most 1, then either the expected final length is infinite, or the expected number of symbols at time t is upper bounded by $(2t+2)^m$.*

Proof:

Assuming the expected final length is finite, we will prove inductively on m that the expected number of symbols at time t is upper bounded by $(2t+2)^m$

The general idea is that starting with S, we can't produce S too fast, otherwise the expected length will become infinite. Furthermore, since L is a context-free language, if all we want to know is the length of the string, we only need to keep track of how many copies of each symbol is currently in the string. If each time we generate S we isolate that symbol into a new sub-system, then each sub-system is essentially a system with $m-1$ different symbols and the growth speed will be bounded by the induction hypothesis.

The theorem is true when $m = 2$. Since there are only two symbols S and α, starting from S, if the rate of the production rule $S \to SS$ is higher than the rate at which $S \to \alpha$, then the expected length is infinite. Otherwise the expected length is linear in t, since the expected number of symbols S in a string is at most 1 at any given time.

Assume that the theorem is true for $m = k-1$. For $m = k$, we subdivide the sets of symbols into many subsets in the following way: initially, there is only one subset that contains S; whenever one copy of S gets produced in any subset, we move that symbol S into a new subset; when other types of symbols are produced, they stay in the same subset.

First, we show that the expected number of symbols in each subset is quite small at time t. We start by considering the subset T_1 that the initial symbol S belongs to. After applying the first production rule, the subset T_1 has at most 2 symbols and will never contain another copy of S again. Therefore, after that first production rule, only $k-1$ different symbols can appear in that subset. By the induction hypothesis, either the expected number of symbols in T_1 goes to infinity, or the expected number of symbols is upper bounded by $2 \cdot (2t+2)^{k-1}$. The exact same argument can be applied to all other subsets.

Second, we will show that the expected number of subsets at time t is at most t. Notice that the number of subsets is equal to the number of symbols S that have been generated in the process. For the expected final length to be finite, the expected number of symbols S in the system is at most 1 at any given time. (Otherwise the number of symbols S is expected to grow exponentially, a contradiction.) Furthermore, since the total rate of all rules with S on the left side is 1, the expected rate at which S is removed by applying production rules is also at most 1 at any time. Therefore, at any time t, the expected number of symbols S that have been removed by a production rule is at most t. Combining the above arguments, the expected number of symbols S that have ever appeared in the system before time t is at most $t+1$.

According to our definition, at time t, the expected number of subsets is equal to the expected total number of symbols S that have ever appeared in the system, which is at most $t+1$. Also, the expected number of symbols in each subset is at most $2 \cdot (2t+2)^{k-1}$. Using linearity of expectation, we know that the expected number of total symbols at time t is at most $(2t+2)^k$.

□

Corollary 2 *Given any insertion system with k molecular species and total concentration 1, either the expected final length is infinite, or the expected length at time t is upper bounded by $(2t+2)^{k^2}$.*

Proof: There are at most k^2 different insertion sites in a system with k species. From Theorem 1, we know that the insertion system can be described by a context-free grammar in reduced Chomsky normal form with at most k^2 symbols. Therefore, the proof follows from Theorem 5. □

3.5 Conclusions

We have defined a formal implementable model for active self-assembly utilizing an insertion primitive. We build on the concept of applying biological algorithms to the development of novel techniques in computer science to provide a method by which we can program arbitrary insertion systems whether they be reconfigurable robots, molecules or scripts of symbols. The work here is particularly relevant for the application of computer science to synthetic biology, chemistry and material science. We show a construction for building a line in polylogarithmic time using a polylogarithmic number of monomers and map it to a molecular system. To our knowledge this is a novel assembly system that has never been synthetically constructed before. We also show that with a number of monomer types the system will either grow to infinity or the expected length of the polymer grows polynomially with time. There are many interesting open questions remaining: What other behaviors can be generated by such a simple model? Are there other directly implementable simple primitives that we can add to this model to generate such behaviors? In this chapter we explored the expressive power of this language, and proved that the language is stronger than regular languages, but, at most, as strong as context-free grammars. It remains to be shown whether this system is equivalent to context-free grammars, in which case the language will prove to be even more powerful than we suggest here.

Chapter 4

Toward a Synthetic Polymer that Grows Exponentially Fast[0]

4.1 Abstract

The exponentially-fast-growing polymer introduced in the previous chapter is implemented here using DNA molecules. The result is the first synthetic linear polymer capable of growing in logarithmic time. Insertion and division are implemented by modifying the autonomous polymerization design [Venkataraman et al., 2007], whereby stable oligonucleotide complexes interact using four-way branch migration when a trigger Initiator complex is present in the solution. We experimentally verify the exponential kinetics of our system using spectrofluorimetry, we qualitatively compare the size of our polymers over time to those grown in a linear system via gel electrophoresis and we verify their shape via Atomic Force Microscopy.

[0]This work was coauthored by Nadine Dabby & Ho-Lin Chen*, and is in preparation [Dabby and Chen, 2013a] with the following contributions: experiments and analysis were performed by N.D. with supervision from H-L.C.; the manuscript was written with input from both authors.

4.2 Introduction

Material science and nanotechnology seek to achieve some of the formidable molecular tasks that biology takes for granted, such as the growth of complex structures in two or three dimensions in logarithmic time. One of the main thrusts in molecular programming is to use computer science as a tool for figuring out what can be achieved. While molecular systems that are Turing-complete have been demonstrated [Winfree, 1996], these systems still cannot achieve some of the feats biology has achieved. The need for new formalisms to describe what molecular systems [Woods et al., 2013] and macro-scale systems [Chirikjian, 1993, Goldstein and Mowry, 2004, Rosa et al., 2006, Griffith, 2004, White et al., 2004, Klavins et al., 2004, Jones and Mataric, 2003, Murata et al., 1994, Nagpal, 2008, Werfel and Nagpal, 2007, Arbuckle and Requicha, 2004, Rus and Vona, 2001, Butler et al., 2001, Yim et al., 1997, Yim et al., 2007, Groß and Dorigo, 2008, Walter et al., 2004] are capable of has spawned research in the area of active self assembly to describe the behaviors that have been and can be implemented by such systems [Yin et al., 2004, Tian et al., 2005, Bath et al., 2005, Pei et al., 2006, Green et al., 2008, Omabegho et al., 2009, Yin et al., 2008, Lund et al., 2010, Muscat et al., 2011, Dirks and Pierce, 2004, Venkataraman et al., 2007, Yin et al., 2008].

In the previous chapter, a model for active self assembly was constructed. This chapter presents the molecular implementation of two active behaviors (exponential growth and splitting or division of polymers) using DNA.

By encoding the order of the nucleotides in a DNA sequence, we can control the interaction of DNA strands. Sub-sequences of these strands are called domains and it is their binding (hybridization) and unbinding (disassociation) from complementary domains that determines what a system of strands can do. In DNA nanotechnology, dynamic systems of DNA molecules can be orchestrated by toeholds, the short sequences of DNA that are complementary to single-stranded domains in a target molecule [Yurke et al., 2000, Zhang and Winfree, 2009]. Toeholds serve as the inputs to dynamic DNA systems and initiate branch migration, a random walk process of bond breaking and formation that results in the exchange of one strand in the duplex for another single strand with the same sequence (described in Chapter 1).

Our DNA implementation is inspired by the Hybridization Chain Reaction (HCR) system de-

veloped by Dirks and Pierce [Dirks and Pierce, 2004]. Their construction, which triggers the polymerization of DNA monomers, uses two single-stranded DNA hairpins that each have the same 18 base-pair stem sequence and a toehold that is complementary to the other hairpin's loop sequence. These strands are kinetically trapped–they are unable to access their lowest energy state because of the large energy required to disrupt the hairpin conformation. As a result, they react with each other very slowly in the absence of an initiator. The initiator is a molecular trigger that consists of a domain that is complementary to one hairpin's toehold and another domain that is complementary to that hairpin's adjacent stem sequence. When the initiator is added to the solution of monomer hairpins, it binds to the toehold of the first hairpin and launches a strand displacement reaction that opens that hairpin. The newly exposed bases of the opened hairpin can then undergo a similar reaction with the second hairpin. The two hairpins will continue to polymerize until an equilibrium concentration of hairpin monomers is reached.

In a subsequent work, the HCR system was modified to employ four-way branch migration and create an autonomous polymerization motor [Venkataraman et al., 2007]. The metastable fuel hairpins from the Hybridization Chain Reaction system were modified to include an extra toehold, and the initiator strand was replaced by an initiator complex that is composed of an "anchor" strand and a "rickettsia" strand. Upon mixing, the first hairpin binds to the sticky ends of the anchor-rickettsia complex, initiating a four-way branch migration in which the rickettsia strand is passed from the anchor to the hairpin. The second hairpin then binds to the newly exposed sticky ends and the rickettsia strand is passed to the second hairpin. The rickettsia strand continues to be passed forward to newly added hairpins as the polymer grows behind it.

In Chapter 3, we explored the implications of modifying the Rickettsia system (described in Chapter 1) by adding an additional hairpin and an additional toehold within each loop. Here we first describe our molecular implementation of this modification (Section 4.3) and then describe our main results (Sections 4.4 and 4.6).

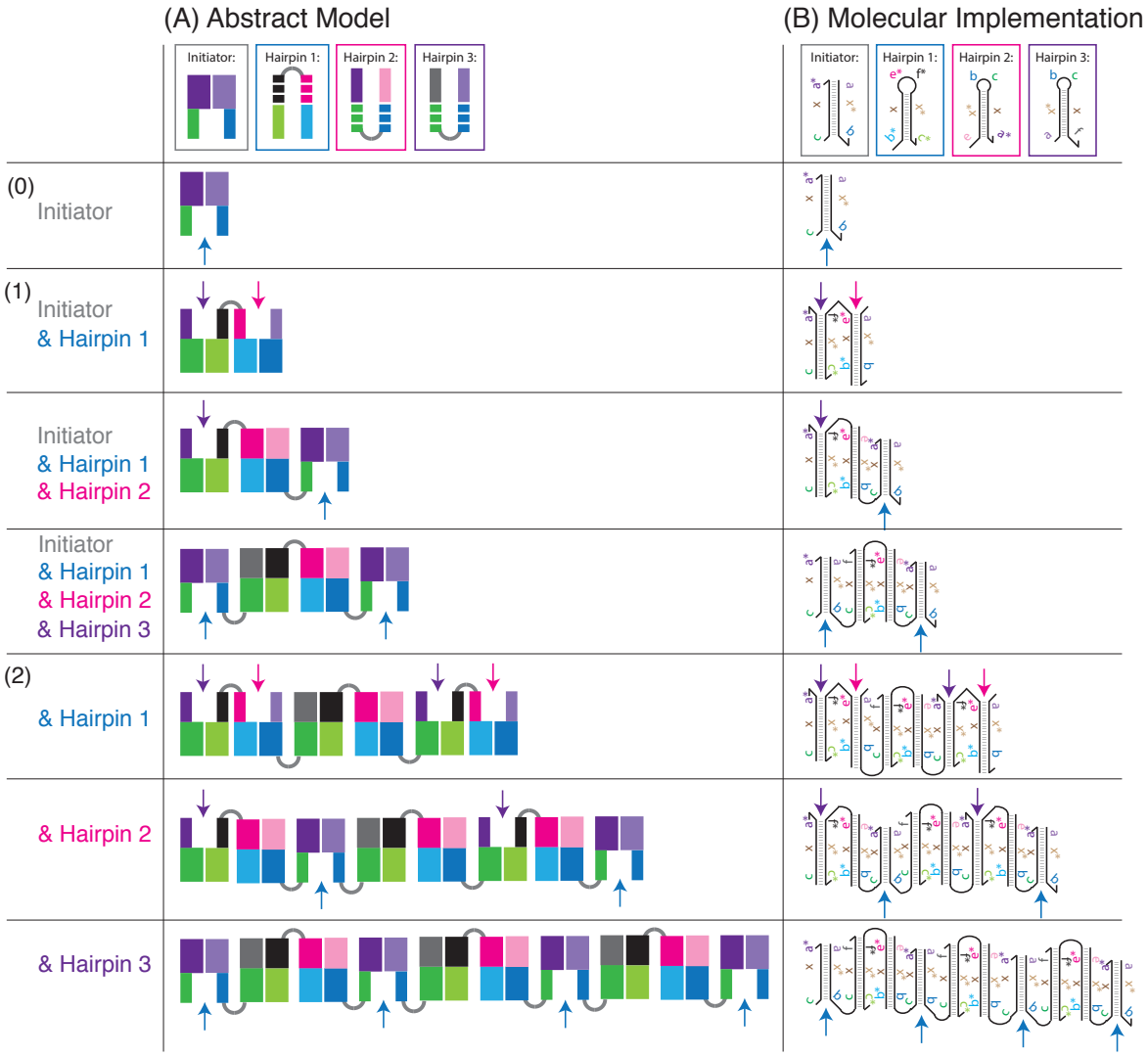


Figure 4.1: (Identical to Figure 3.4.) The schematic of our insertional polymer implementation shows the first two rounds of growth. (A) The abstract representation of our exponential growth polymer. (B) The molecular implementation of our polymer is color-coded the same way. DNA sequences in the oligonucleotides are color-coded by domain (purple, green, blue, brown, pink, and black). The boxes around each oligonucleotide in (B) correspond to the insertion arrows as follows: a blue arrow indicates an insertion site for Hairpin 1, a pink arrow indicates an insertion site for Hairpin 2, a purple arrow indicates an insertion site for Hairpin 3. Exponential growth occurs as follows: (0) The Initiator has one insertion site for Hairpin 1 (blue arrow). Insertion of Hairpin 1 is driven forward by the hybridization of 6 new base pairs. (1) After Hairpin 1 inserts into the Initiator, two new insertion sites are generated: one for Hairpin 2 (pink arrow) and one for Hairpin 3 (purple arrow). Hairpin 2 and Hairpin 3 are sequentially inserted (in solution insertion occurs asynchronously), each one generates a new insertion site for Hairpin 1 (blue arrows). After the first round of insertion, two insertion sites for Hairpin 1 are generated from what was initially (in round (0)) one site. (2) A second round of insertion is illustrated. After the second round of insertion, four new insertion sites for Hairpin 1 are generated.

4.3 Molecular Implementation

Figure 4.1 (identical to Figure 3.4 reproduced here for convenience) shows the molecular implementation of our exponential growth system from the previous chapter. Hairpin 1 (H1) and the Initator (I) react first; this results in two new insertion sites: one that is complementary to Hairpin 2 (H2), and another that is complementary to Hairpin 3 (H3). Upon insertion of H2 and H3 into the growing polymer two new insertion sites that are complementary to H1 are regenerated. Thus for every initial H1 insertion site, after each round of insertions (of H1, H2 and H3), two new H1 insertion sites are created.

The initial reaction (insertion of H1 into the Initiator complex) is driven by the hybridization of six new base pairs. After that, each new hairpin that is inserted adds nine base pairs to the system. Some of these steps become reversible as the system reaches equilibrium. The free energy and reversibility of toehold-mediated four-way branch migration is explored in depth in Chapter 5.

In addition to the insertional monomers that grow the polymer, we introduce a new type of monomer, which we call a Divide complex, that upon insertion splits the polymer into two pieces, as we will discuss in Section 4.6.

Figure 4.2 is a legend for the set of DNA molecules used in this chapter. Each oligonucleotide complex (Initiator, Hairpin1, Hairpin 2, Hairpin 3, and Divide) is shown with color-coded motifs (purple, green, blue, brown, pink, and black) that correspond to the colored DNA subsequences (Figure 4.2A). The Initiator-ROX complex is a modified Initiator complex with a single fluorophore tag for gel electrophoresis experiments (Figure 4.2B). Hairpin 2RQ (H2RQ) is a modified Hairpin 2 molecule with a quencher and fluorophore pair on opposite ends of the molecule, used in the spectrofluorimetry experiments (Figure 4.2B). Hairpin 2L (H2L) and Hairpin 3L (H3L) are inactivated versions of Hairpins 2 and 3, in which the loops are replaced with a poly-T sequence (Figure 4.2C). The boxes around each oligonucleotide correspond to the insertion arrows as follows: a blue arrow indicates an insertion site for Hairpin 1, a pink arrow indicates an insertion site for Hairpin 2, Hairpin 2RQ or Hairpin 2L, a purple arrow indicates an insertion site for Hairpin 3 or Hairpin 3L, and a green arrow indicates an insertion site for the Divide complex (Figure 4.2D).

In each diagram, we utilize a domain abstraction for referring to stretches of consecutive nu-

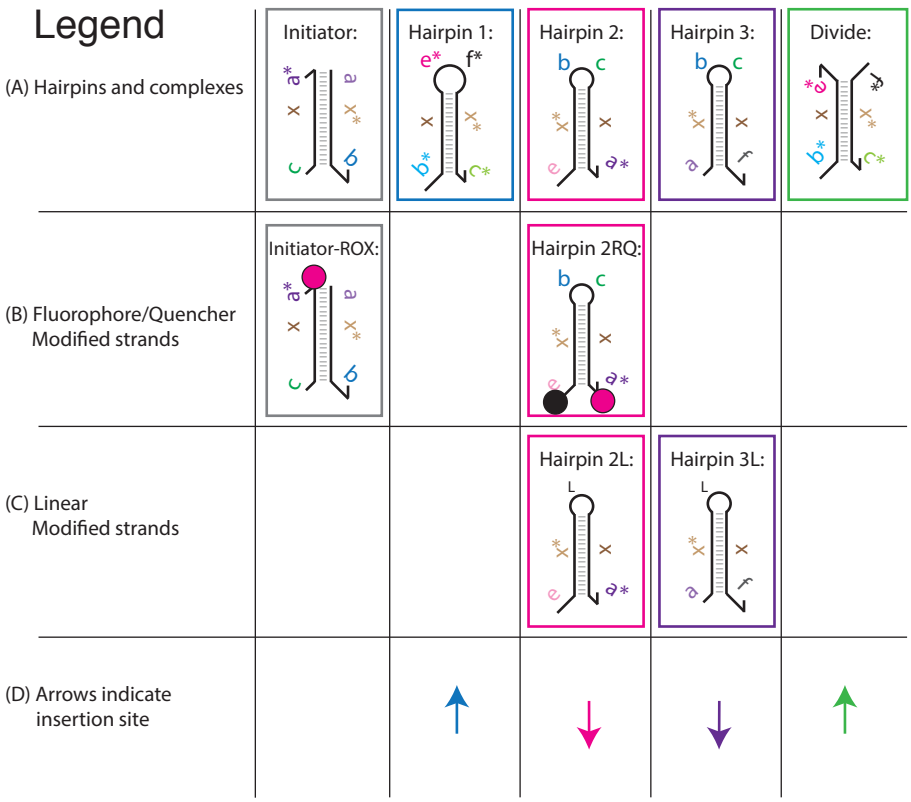


Figure 4.2: Legend of DNA hairpins and complexes. (A) Schematics of the Initiator complex, Hairpin 1, Hairpin 2, Hairpin 3, and Divide complex. Each oligonucleotide is shown with color-coded motifs that correspond to the DNA subsequences. (B) The Initiator-ROX complex is a modified Initiator complex with a single fluorophore tag for gel electrophoresis experiments. Hairpin 2RQ is a modified Hairpin 2 molecule with a quencher and fluorophore pair on opposite ends of the molecule, used in the spectrofluorimetry experiments. (C) Hairpin 2L and Hairpin 3L are inactivated versions of Hairpins 2 and 3, in which the loops are replaced with an inactive poly-T sequence. The boxes around each oligonucleotide in (A) (B) (C) correspond to the insertion arrows in (D) as follows: a blue arrow indicates an insertion site for Hairpin 1, a pink arrow indicates an insertion site for Hairpin 2, Hairpin 2RQ or Hairpin 2L, a purple arrow indicates an insertion site for Hairpin 3 or Hairpin 3L, a green arrow indicates an insertion site for the Divide complex.

cleotides that act as a unit in binding to complementary stretches of nucleotides. Domains are represented by Latin letters (Figure 4.1). Letters followed by an asterisk denote complementary domains, e.g.: **x** is complementary to **x***. Single-stranded molecules of DNA (henceforth strands) are comprised of concatenated domains. DNA complexes are composed of two or more noncovalently-bound strands. There are two types of toeholds in our system: long toeholds that indicate a stronger desired interaction (six bases in length) and short toeholds that indicate a weaker desired interaction (three bases in length).

In the next two sections, we confirm exponential growth by measuring the conversion of monomers into a product. We then qualitatively measure the size of products over time. Finally, we verify the predicted structure using Atomic Force Microscopy.

4.4 Exponential Growth Results

4.4.1 Exponential Growth Mechanism Controls

We tested each insertion step in the exponential growth mechanism by using the inactivated versions of of Hairpins 2 and 3 (Figure 4.3). Hairpin 2L and Hairpin 3L were added to the Initiator and Hairpin 1 both individually (this results in exactly one insertion event) and together with the normal version of the other hairpin, which results in linear growth. We note that there is more product in lanes 14 (I, H1, H3L) and 15 (I, H1, H3) than there is in lanes 12 (I, H1, H2L) and 13 (I, H1, H2). The reactants in lanes 12 and 14 can only proceed through two steps of the polymerization reaction due to the inactivated strands. At equilibrium (after 6 hours) there is more dimerization between the Initiator-Hairpin 1 complex and Hairpin 3L than there is between the Initiator-Hairpin 1 complex and Hairpin 2L. Thus Hairpin 3 appears to have a greater affinity to the Initiator-Hairpin 1 complex than Hairpin 2. This observation implies that the two reactions have different rate constants, Hairpin 2 is either slower to react with its insertion site or faster to dissociate from its insertion site than Hairpin 3 (or both).

When all possible combinations of the reactants are made, the leaks in the system can be assessed using gel electrophoresis. A “leak” is an undesired molecular interaction. We tested four

different sequence designs (Section B.1), and chose the one with a low leak and fast reaction time. Lanes 9 (H1, H2) and 10 (H1, H3) in Figure 4.3 show that a small leak occurs between Hairpin 1 and Hairpin 2 and between Hairpin 1 and Hairpin 3. However, no leak occurs between any of the other species.

We quantify the leak via spectrofluorimetry experiments in Figure 4.8 [Yin et al., 2008]: we adjust the Initiator concentration $[I]$ by an additional term $[I]_{leak}$ to obtain an effective Initiator concentration $[I]_{effective} = [I] + [I]_{leak}$. We then fit the $[I]_{leak}$ parameter to our data and find that in the exponential system $[I]_{leak} = 0.04\times$ and in the linear system $[I]_{leak} = 0.01\times$. Reactions were started with the addition of Hairpin 1 in order to avoid the leak.

The reader may observe the presence of faint extra bands in the lanes that contain only individual hairpins. These are dimerized hairpins that form in small amounts from individual hairpins when the strands are annealed. We minimize their presence by snap cooling. Snap cooling the hairpins results in the same amount of dimerized monomers as gel purification (data not shown). All hairpins except for the Initiator were snap cooled prior to experiments. The Initiator is a gel-purified duplex composed of two molecules of DNA.

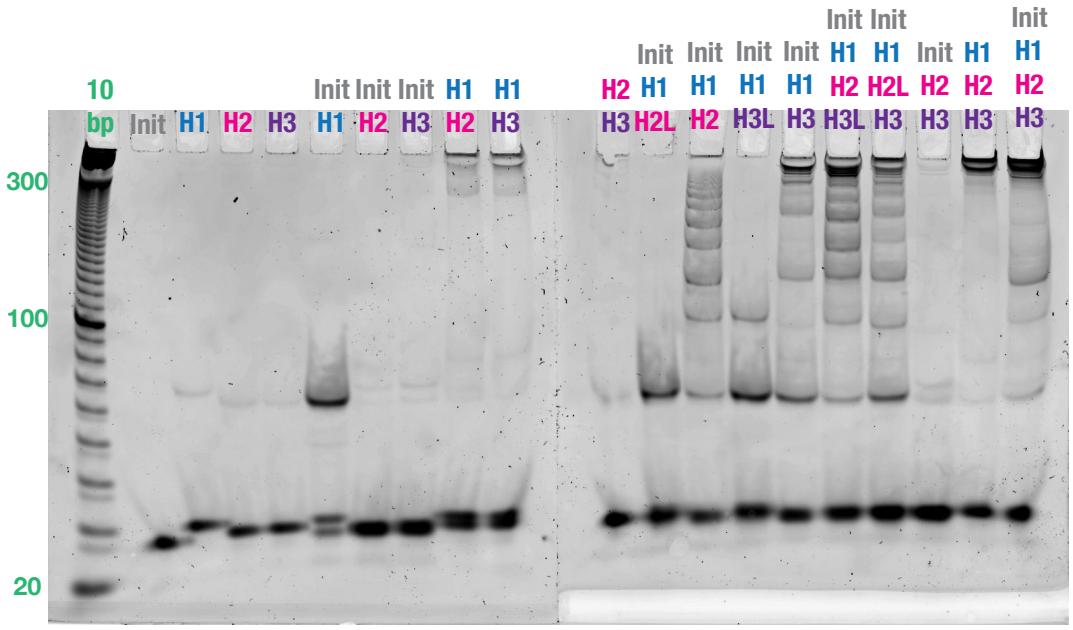


Figure 4.3: The goal of non-denaturing gel control experiments (as shown above) is to show if there are undesired interactions between each combination of each reactant. The polyacrylamide gel above shows that a small undesired leak occurs between reactants Hairpin 1 (H1) and Hairpin 2 (H2) and between reactants Hairpin 1 (H1) and Hairpin 3 (H3). However, no leak occurs between any of the other species. We fit the leak via spectrofluorimetry experiments in Figure 4.8. The leak is small enough that it doesn't interfere significantly with our experiments. All species are present at 100 nM concentration.

4.4.2 The Kinetics of Parallel Insertion.

We examined the kinetics of the conversion of monomers into the polymer by adding a fluorophore and quencher pair to the opposite ends of Hairpin 2. Before reaction, the fluorophore is quenched. Upon incorporation of the hairpin into the DNA polymer, the quencher and fluorophore pair are separated, and the fluorescence of the solution increases (Figure 4.4).

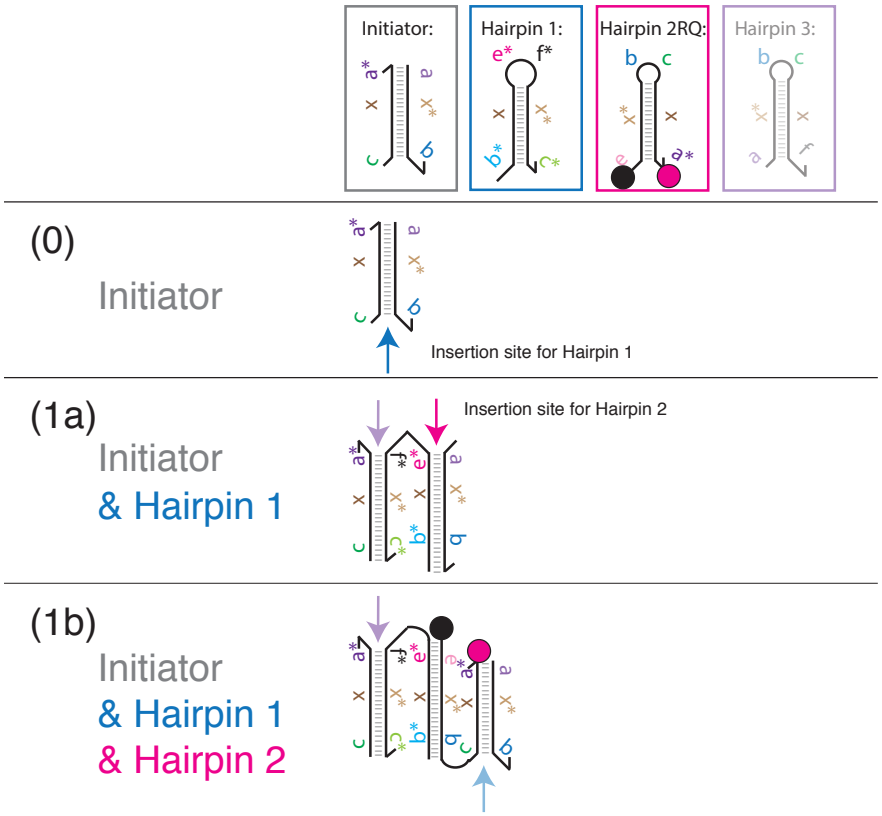


Figure 4.4: The experimental design for measuring conversion of monomer hairpins into the polymer. Above is a modified version of the schematic from Figure 4.1 The boxes around each oligonucleotide correspond to the insertion arrows in the mechanism below, which shows the incorporation of Hairpin 1 (1a) and Hairpin 2 (1b) into the polymer. Note that Hairpin 2RQ is a modified version of Hairpin 2 that includes a fluorophore-quencher pair. The fluorophore (pink circle) is quenched before Hairpin 2RQ reacts with the polymer. Upon Hairpin 2RQ's insertion into the polymer (1b), the fluorophore (pink circle) and quencher (black circle) are separated and the fluorophore emits light. We measure the kinetics of the incorporation of Hairpin 2RQ into the polymer by measuring the increase of fluorescence in the solution over time.

We probed both the linear and exponential polymerization over eight different Initiator concentration values. The time course of fluorescence intensity confirmed linear conversion of hairpins in the system with one inactivated strand (Figures 4.5, and 4.6), and exponential conversion of hairpins in the full system (Figures 4.7 and 4.8).

In a linear growth system, the total mass of polymer product, P, grows as a function of initial

Initiator concentration, I_0, and time, t, as follows:

$$P = k \times I_0 \times t. \tag{4.1}$$

The time at which 10% of monomers are consumed, $t_{10\%}$, is

$$t_{10\%} = \frac{P_{10\%}}{k \times I_0}. \tag{4.2}$$

Thus, in a linear growth system, the time to 10% completion of polymer growth (10% conversion of hairpins) is inversely proportional to initial Initiator concentration. When plotted on a logarithmic concentration scale, the time to 10% conversion exponentially decays as a function of increasing initial Initiator concentration. This model fits our linear growth system data (Figure 4.6).

In an exponential growth system, the total mass of polymer product, P, grows as a function of initial Initiator concentration, I_0, and time, t, as follows:

$$P = I_0 \times e^{(kt)}. \tag{4.3}$$

The time at which 10% of monomers are consumed, $t_{10\%}$, is

$$t_{10\%} = \frac{1}{k} \times (\ln(P_{10\%}) - \ln(I_0)). \tag{4.4}$$

Thus, in an exponential growth system, the time to 10% completion of polymer growth (10% conversion of hairpins) is a linear function of the logarithm of the initial Initiator concentration. When plotted on a logarithmic concentration scale, the time to 10% conversion linearly decreases with increasing initial Initiator concentration. This is what we observe in our exponential growth system data (Figure 4.8).

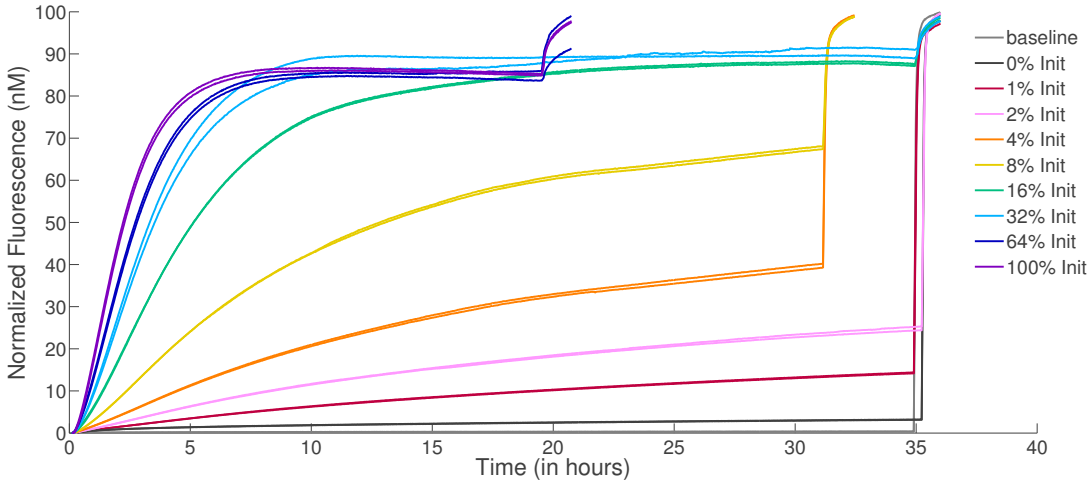


Figure 4.5: Linear polymer growth kinetics are observed in a fluorescence time course when inactivated Hairpin 3L is substituted for Hairpin 3. As Hairpin 2 is incorporated into the growing polymer, the system's fluorescence increases: this illustrates the conversion of hairpins into polymers. Plotted above are the kinetic traces of Hairpin 2RQ (all hairpins are present at 100 nM) with varying amounts of Initiator.

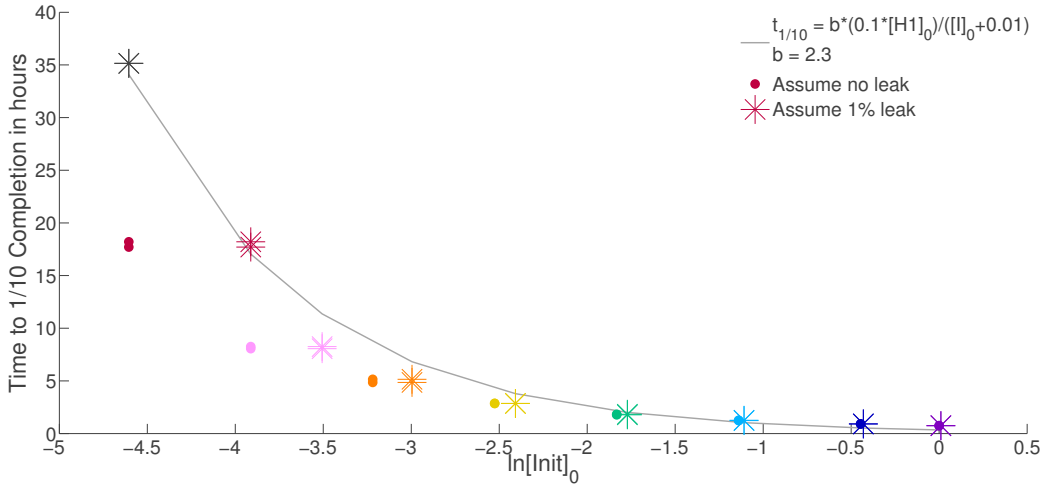


Figure 4.6: Linear polymer growth kinetics are observed when an inactivated version of Hairpin 3 is substituted for Hairpin 3. Above is the linear fit of the 10% completion time as a function of the relative concentration of Initiator to hairpins. Filled circles correspond to a system where we assume no leak. Asterisks indicate the same points but assume a leak equivalent to 1% of the Initiator concentration.

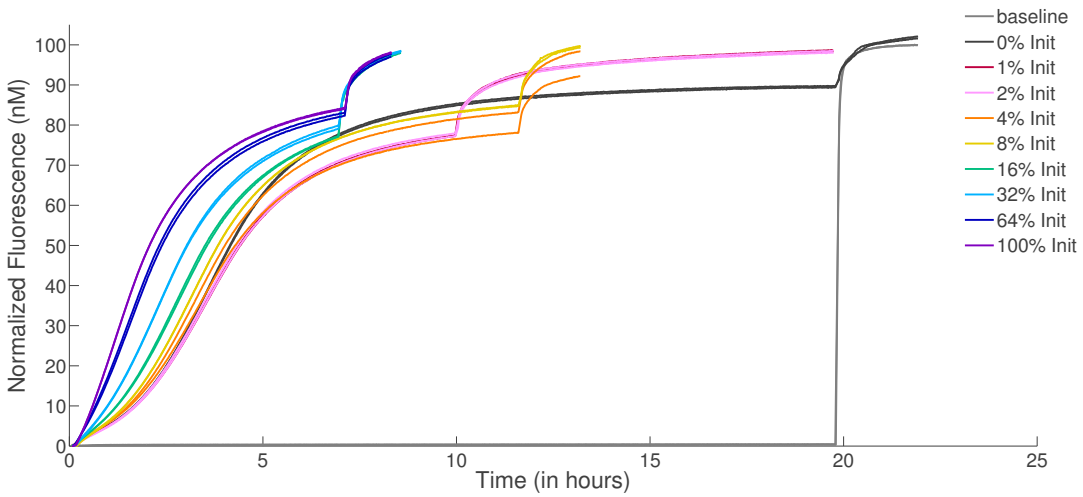


Figure 4.7: Exponential polymer growth kinetics examined via fluorescence. As Hairpin 2RQ is incorporated into the growing polymer, the system's fluorescence increases; this illustrates the conversion of hairpins into polymer. Plotted above are the kinetic traces of Hairpin 2RQ (all hairpins are present at 100 nM) with varying amounts of Initiator.

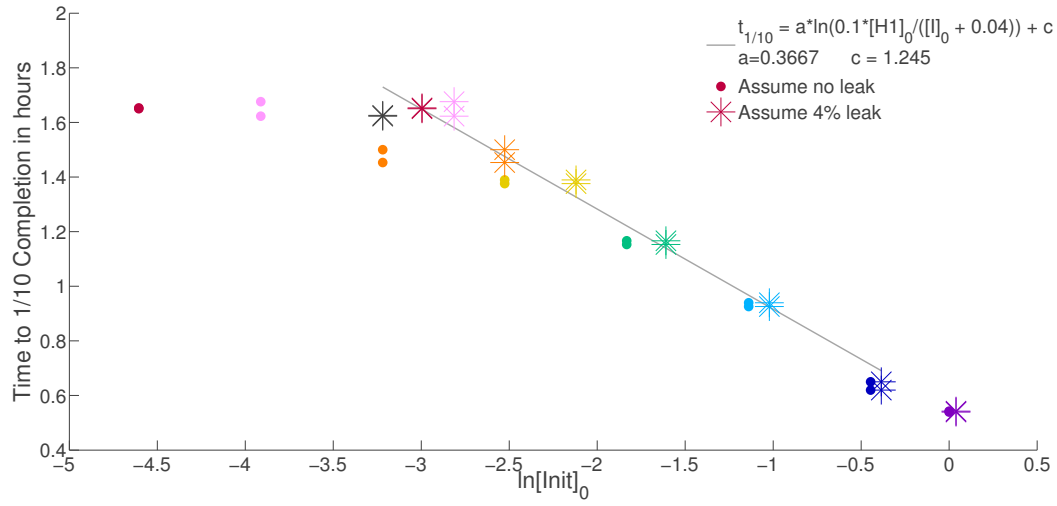


Figure 4.8: Linear fit of the 10% completion time as a function of the relative concentration of Initiator to hairpins. Filled circles correspond to a system where we assume no leak. Asterisks indicate the same points but assume a leak equivalent to 4% of the Initiator concentration. We assume leak is equivalent to added Initiator using a model from [Yin et al., 2008]. Using that assumption the data is shifted and the curves match. Thus the leak shown in Figure 4.3 does not significantly affect our data.

The polymers formed at each Initiator concentration were examined by gel electrophoresis in order to characterize their length distribution. Each Initiator molecule was tagged with one ROX

fluorophore. As the hairpins are successively added to a polymer, each polymer that is "properly initiated" retains exactly one fluorophore, thus the ROX fluorescence signal directly correlates to the number of polymers at a given size. The sizes were binned after post-staining with SYBR Gold, which allowed the DNA ladder to be visualized.

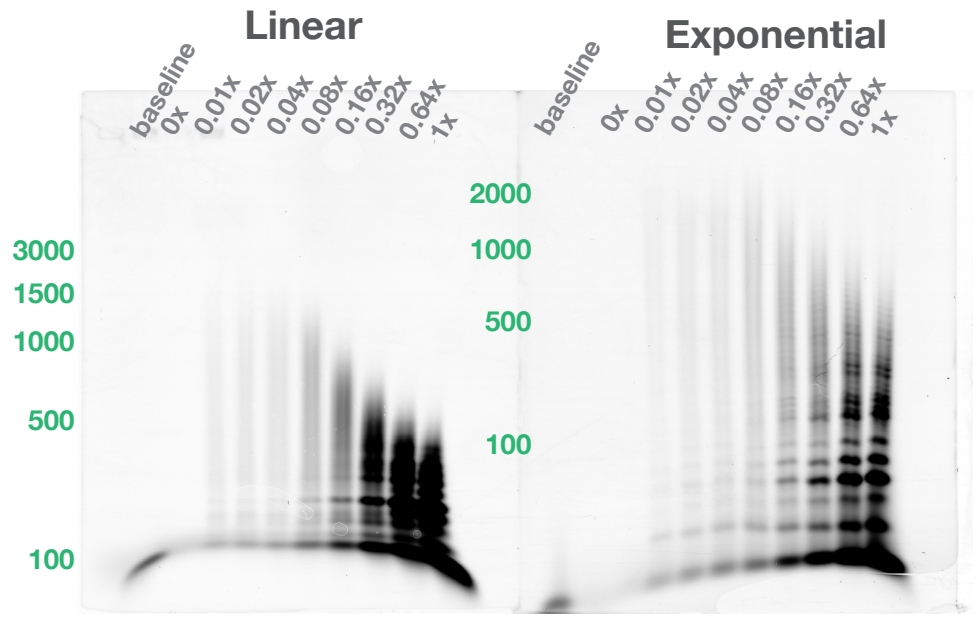


Figure 4.9: The average length of the polymer shrinks with increasing Initiator concentration (left to right) in both the linear and exponential systems. Rox fluorescence intensity imaged in this Super Fine Resolution Agarose gel shows the distribution of polymer lengths generated in the presence of Initiator concentrations $[I]_0 = \{0\%, 1\%, 2\%, 4\%, 8\%, 16\%, 32\%, 64\%, 100\%\}$ relative to hairpin concentrations after 6 hours of reaction with 1μM Hairpins. Lanes with Initiator concentrations smaller than 8% are difficult to resolve by eye in this gel as they contain less ROX-labeled Initiator. Figure B.5 shows an image of this gel after staining with SYBR Gold.

The mean length (in base pairs) of polymers decreases with increasing Initiator concentration above 4% of relative hairpin concentrations. (See Figure 4.9, 4.10, B.5 for gels and binned data of both linear and exponential systems). This is expected because high concentrations of Initiator outcompete existing insertion sites for free hairpins. In the case of Initiator concentration below 4% of relative Hairpin concentrations, the different amounts of leak in the systems are presumably responsible for the different distributions of polymer length between the linear and exponential

system. The smaller leak in the linear system (1%) would explain why the linear system produces longer polymers than the exponential system (which has a 4% leak).

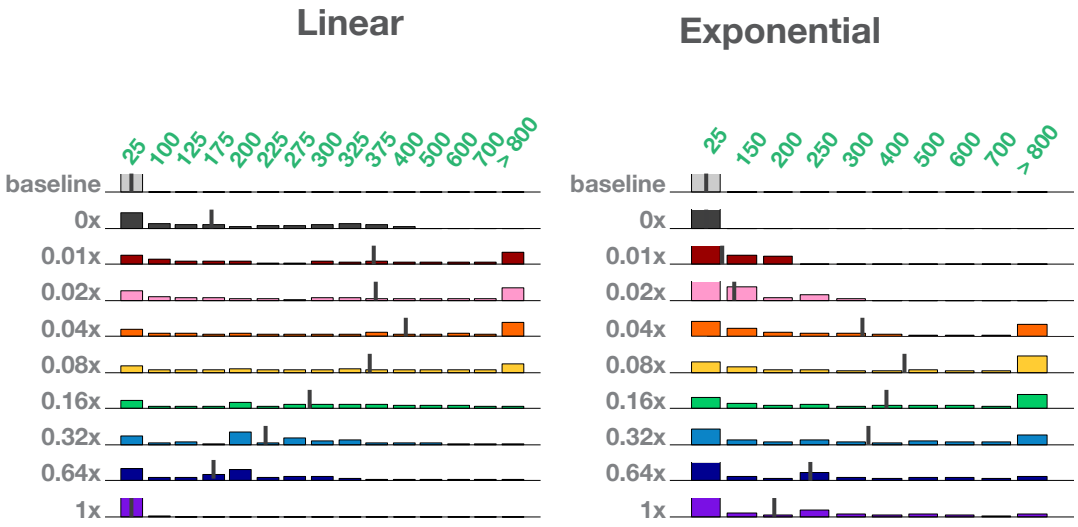


Figure 4.10: Normalized distributions of polymer length from the data in Figure 4.9 show a decrease in the mean length of polymer with increased Initiator concentration in both the linear and exponential systems. The upper limit of the y-axis for each distribution is 40% of the total concentration to allow the reader to clearly see the change in distribution. Vertical lines indicate the lower bound on the mean of the distribution, as calculated with all polymers larger than 800 base pairs being assigned a length of 800 base pairs. The mean of each distribution in which more than 8% Initiator was utilized decreases with increasing amounts of Initiator.

4.4.3 Imaging with Atomic Force Microscopy

Atomic Force Microscopy of the reaction product confirms the formation of unbranched polymers in the expoential system (Figure 4.11). In comparing images of both the polymer and the leak product, we find that the leak product is capable of growing much larger than the intended polymer, but the polymer grows faster. Others have shown that polymer growth in the absence of Initiator can provide an upper bound for how big the polymer can grow [Beck, 2011]. It is unclear whether the leak product is a linear polymer. It may be a highly pseudo-knotted structure.

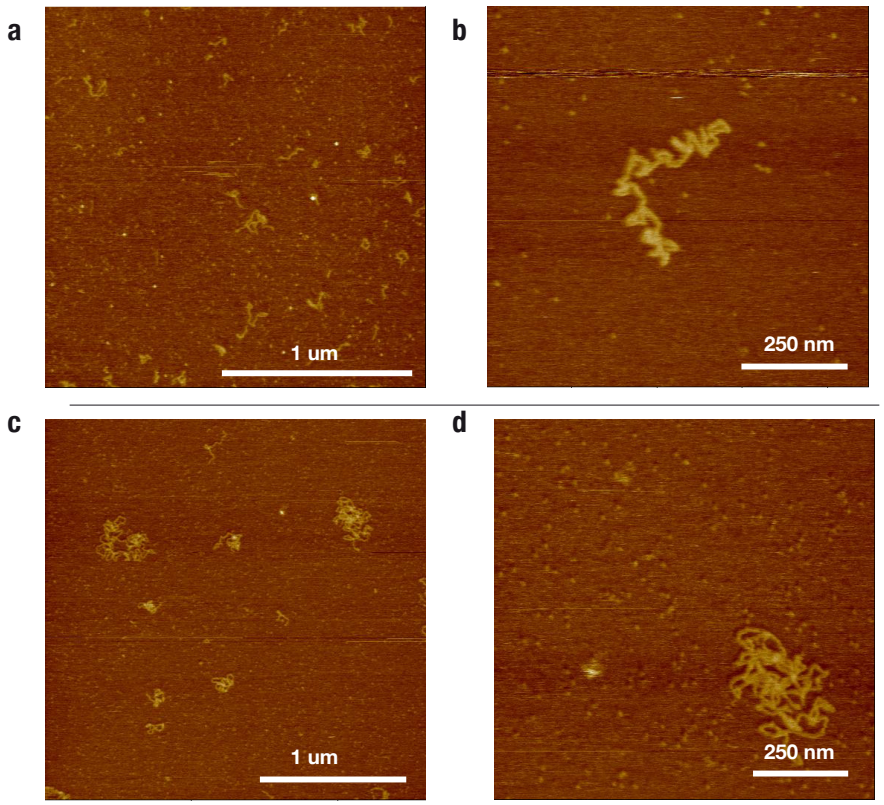


Figure 4.11: Atomic Force Microscopy images of exponentially grown polymer and leak product. a, Wide field image of polymer with 10% Initiator strand relative to the other hairpins after 5 hours (scale bar is 1 um). b, One of these polymers (scale bar is 250 nm). c, Wide field image of the leak formed by hairpins in the absence of Initiator (scale bar is 1 um). d, One of these leak products (scale bar is 250 nm).

4.4.4 Time Lapse Experiments

A qualitative difference between the exponential and linear systems is also observed when examining polymer size over time in Figures 4.12 and 4.13. (See Figures B.6 and B.8 for two additional exponential system time lapse gels and see Figures B.7 and B.9 for the SYBR Gold stained versions of all of these gels).

According to this preliminary gel data, the exponential system (Figure 4.13) generates longer polymer products sooner than the linear system (Figure 4.12). While it takes the linear system 25 min to produce a polymer of length 500 base pairs, the exponential system produces a 500-base pair

polymer within 10 min. The exponential system produces a detectable amount of 1000-base pair polymer within 20 minutes, at least four times faster than the linear system, which takes between 90 and 120 min to produce a 1000-base pair polymer. Although this is not proof of exponential growth, it is consistent with the expectation that exponential growth progresses more quickly than linear growth.

Figure 4.13 is particularly rich in data. In addition to showing that the polymers produced in the exponential system grow large quickly, the gel clearly shows that polymer growth occurs in quantized chunks of approximately 25 base pairs at a time. This is expected, as each hairpin contains between 54 and 57 nucleotides. The bands generated by the polymerization alternate between faint and dark within each lane. This corroborates our earlier claim that Hairpin 2 is slower to react with its insertion site than Hairpin 3. If the backward reaction rates for both of these reactions are equivalent, then this implies that the reaction between H2 and its insertion site is a slower step in the formation of polymers.

The exponential time lapse gel in Figure 4.13 and the replicate in Figure B.8 expose an issue. The signal of the bands relative to background fades from left to right. In the SYBR Gold-stained versions of these gels, as shown in Figure B.9, the lanes to the right show noticeably less total stained DNA than the other lanes. We suspect that this behavior is a result of the complexity of loading the gel: in order to ensure that the experiments are initiated and the gel is run exactly on time, the right half of the gel (higher time point reactions) is loaded approximately 30 minutes in advance of the shorter time lapse reactions. This may allow for the DNA in these wells to diffuse out of the wells in advance of running. Another concern is the fading of the bands at the top of the gel in the longer time lapse reactions. It is unclear why this fading occurs. More repetitions of these results will be necessary.

We hypothesize that the fluorescent loading dye bromophenol blue interferes with the fluorescence read-out of our properly initiated polymers. The gel in Figure 4.13 has a dark band in all lanes across the bottom of the gel. By comparison, this band becomes faint at intermediate times for the replicate in Figure B.8 and disappears at long time points in the replicate in Figure B.6. In the SYBR Gold-stained versions of the gels in Figure 4.13 and Figure B.8, as shown in Figure B.9, this band fades significantly. Since bromophenol blue does not fluoresce at the excitation spectra

of SYBR Gold, we can assume that only stained DNA is visible, and that if the dark lower bands in the gels were unused initiator, then there would be a larger amount of DNA at these lengths. Thus, a more complete analysis of these gels was precluded due to the interference of the fluorescent loading dye bromophenol blue and an improperly stained ladder in the linear system time lapse gel that makes it difficult to resolve at molecular weights above 1000 base pairs (Figure B.7).

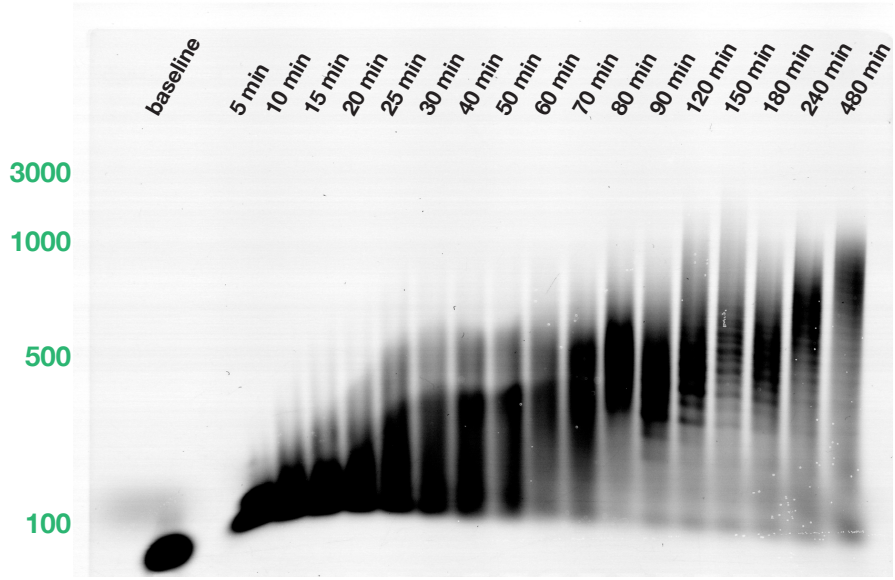


Figure 4.12: Gel time-lapse studies of linear polymer growth. Super Fine Resolution Agarose non-denaturing gels of the product of a polymerization reaction with 80 nM ROX-labeled Initiator, 1.5 μM Hairpin 1, and 1 μM of Hairpin 2 and Hairpin 3. ROX fluorescence was imaged prior to staining with SYBR Gold. (The SYBR Gold stained gel can be found in Figure B.7). A more complete analysis of this gel was precluded due to the interference of the fluorescent loading dye bromophenol blue as discussed in Section 4.4.4.

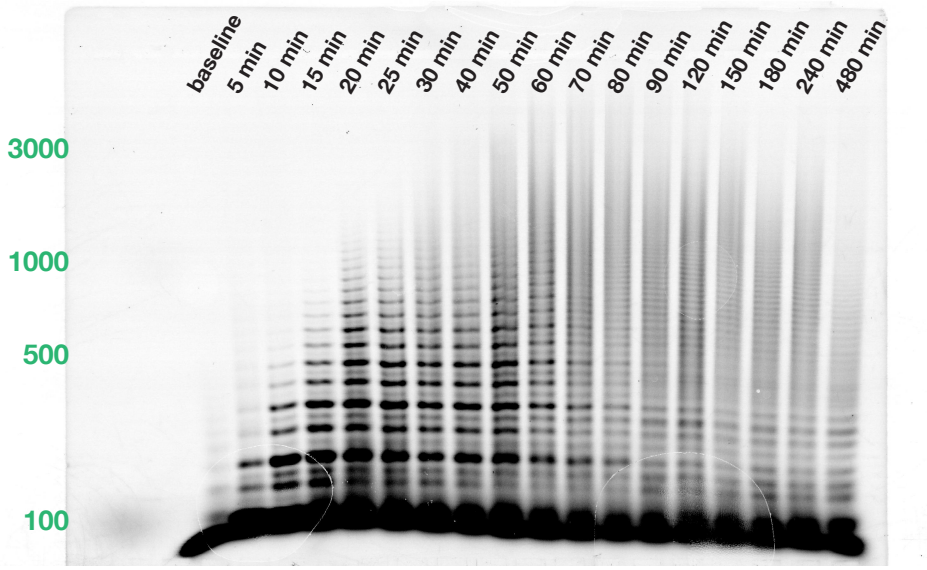


Figure 4.13: Gel time-lapse studies of exponential polymer growth. Super Fine Resolution Agarose non-denaturing gels of the product of a polymerization reaction with 80 nM ROX-labeled Initiator, 1.5 μM Hairpin 1, and 1 μM of Hairpin 2 and Hairpin 3. ROX fluorescence was imaged prior to staining with SYBR Gold. (The SYBR Gold stained gel can be found in Figure B.9). Two additional experimental runs of this experiment can be found in Figures B.6 and B.8. A more complete analysis of this gel was precluded due to the interference of the fluorescent loading dye bromophenol blue as discussed in Section 4.4.4.

4.5 Analysis

The exponential growth system described here can be modeled with the following chemical reactions:

$$I1 + H1 \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} S2 + S3, \tag{4.5}$$

$$S2 + H2 \underset{k_{-2}}{\overset{k_2}{\rightleftharpoons}} I2 + P2, \tag{4.6}$$

$$S3 + H3 \underset{k_{-3}}{\overset{k_3}{\rightleftharpoons}} I3 + P3, \tag{4.7}$$

where S2 and S3 are the insertion sites for H2 and H3 respectively, and P2 and P3 are double-stranded sections of polymer that are henceforth unreactive. The chemical reactions can be further

simplified by the following three assumptions:

1. Each Initiation site is equivalent.

2. The forward rates are the same for all three reactions. Thus, $k_1 = k_2 = k_3 = k$.

3. The reactions are irreversible. Thus, $k_{-1} = k_{-2} = k_{-3} = 0$.

The first assumption makes the set of reactions tractable. The second assumption comes from the next chapter, where we show that the forward rate of a four-way branch migration reaction is dependent on the length and sequences of interacting toeholds and consequently on the overall free energy changes in the system. In our implementation all toehold pairs share the property of being nine bases long with approximately equivalent GC content, making their free energy changes roughly equivalent. The final assumption is justified by the decreasing free energy of the system at each step. We note that this assumption may not hold as reactants are consumed by the system.

With the above assumptions the set of reactions can be reduced to:

$$I + H1 \xrightarrow{k} S2 + S3, \tag{4.8}$$

$$S2 + H2 \xrightarrow{k} I + P2, \tag{4.9}$$

$$S3 + H3 \xrightarrow{k} I + P3. \tag{4.10}$$

We next require that [S2] = [S3], henceforth they will both be replaced by a variable called [S]. Then we can simulate the rate of change of [I], [H1], [H2], [H3], and [S] with the following differential equations:

$$\frac{d[I]}{dt} = -k[I][H1] + k[S][H2] + k[S][H3], \tag{4.11}$$

$$\frac{d[H1]}{dt} = -k[I][H1], \tag{4.12}$$

$$\frac{d[S]}{dt} = 2k[H1][I] - k[S][H2] - k[S][H3], \tag{4.13}$$

$$\frac{d[H2]}{dt} = -k[S][H2], \tag{4.14}$$

$$\frac{d[H3]}{dt} = -k[S][H3]. \tag{4.15}$$

The terms in the above equations (4.11, 4.13, 4.14 and 4.15) can be added together to establish

$$0 = 2 \times \frac{d[I]}{dt} + \frac{d[S]}{dt} + \frac{d[H2]}{dt} + \frac{d[H3]}{dt}, \tag{4.16}$$

or, equivalently,

$$\frac{d[S]}{dt} = -2 \times \frac{d[I]}{dt} - \frac{d[H2]}{dt} - \frac{d[H3]}{dt}. \tag{4.17}$$

Taking the integral of Equation 4.17 results in the solution $[S] = -2[I] - [H2] - [H3] + C$. Since the sum of [S], [I], [H2], and [H3] is a constant, we get:

$$[S] + 2[I] + [H2] + [H3] = C = 2[I]_0 + [H2]_0 + [H3]_0, \tag{4.18}$$

$$[S] = 2[I]_0 + [H2]_0 + [H3]_0 - 2[I] - [H2] - [H3]. \tag{4.19}$$

Assuming that [H2] = [H3], and $[H2]_0 = [H3]_0$, this results in:

$$[S] = 2 \times ([I]_0 - [I] + [H2]_0 - [H2]), \tag{4.20}$$

and

$$[S] = 2 \times ([I]_0 - [I] + [H3]_0 - [H3]). \tag{4.21}$$

Finally, we use Equations 4.20 and 4.21 to simplify the set of differential equations (4.11–4.15) to:

$$\frac{d[I]}{dt} = -k[I][H1] + k[S][H2] + k[S][H3], \tag{4.22}$$

$$\frac{d[H1]}{dt} = -k[I][H1], \tag{4.23}$$

$$\frac{d[S]}{dt} = 2k[H1][I] - k[S][H2] - k[S][H3], \tag{4.24}$$

$$\frac{d[H2]}{dt} = -2k[H2]([I]_0 - [I] + [H2]_0 - [H2])), \tag{4.25}$$

$$\frac{d[H3]}{dt} = -2k[H3]([I]_0 - [I] + [H3]_0 - [H3])). \tag{4.26}$$

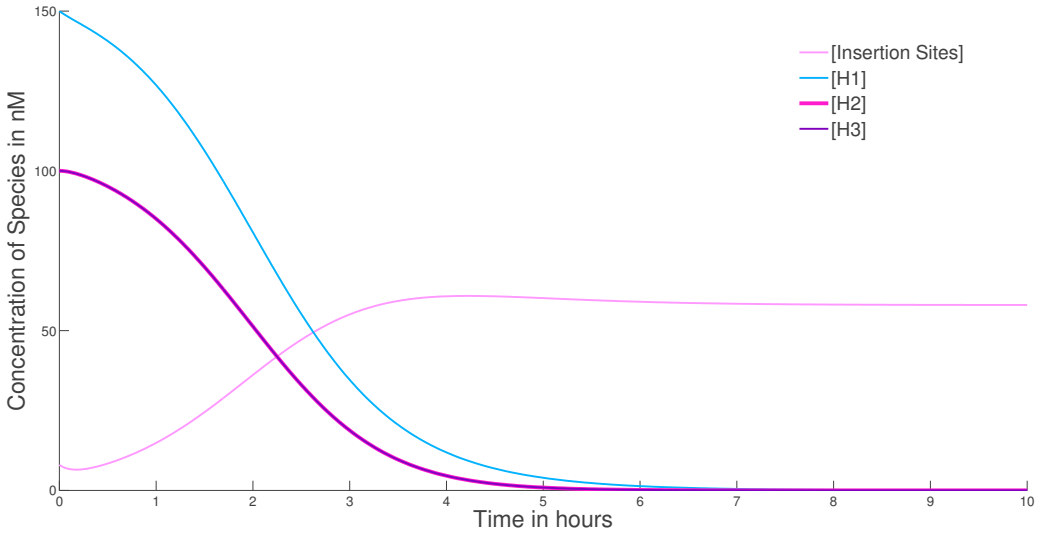


Figure 4.14: The ordinary differential equation model of the exponential growth system discussed in section 4.5 is simulated. The number of insertion sites initially decreases, as the sites interact with Hairpin 1, and then the sites begin to increase exponentially fast until a large fraction of the hairpins are consumed and the system reaches an equilibrium.

When simulated in Matlab, these equations result in a plot of the concentrations of I, H1, H2, and H3 over time. Figure 4.14 shows the simulation. The simulation starts with initial concentrations: $[I]_0 = 8$ nM, $[H1]_0 = 150$ nM, $[H2]_0 = [H3]_0 = 100$ nM as in our spectrofluorimetry experiments. We set $k = 5050 M^{-1} sec^{-1}$, as derived from the spectrofluorimetry data shown in Figure 4.8 and described in the next section.

The simulated concentration of H2 over time shows $[H2]_{total} - [H2]_{incorporated}$. The yellow trace in Figure 4.7 shows $[H2]_{incorporated}$ at the concentrations of molecules simulated. The simulated concentration of H2 is consistent with our measurements of H2 incorporation in the polymer. In the simulation, the number of insertion sites initially decreases, as the sites interact with Hairpin 1, and then the number of insertion sites increases exponentially fast until a large fraction of the hairpins are consumed and the system reaches an equilibrium. The growing number of insertion sites may serve as a proxy for the total concentration of polymer in our system, which we have not measured.

4.5.1 Parameter Fitting

The rate at which the number of H1 insertion sites [I] increases is

$$\frac{d[I]}{dt} = k[I][H1]. \tag{4.27}$$

We examine this rate at 10% completion time, because at that time the concentration of H1 is roughly constant, and the number of Insertion sites [I] is approximately 10% of $[H1]_0$. We substitute C for $k \times [H1]_0$ and $I(t_{10\%})$ for $0.1[H1]_0$ to get

$$\frac{d[I]}{dt} = C[I]. \tag{4.28}$$

Integrating this equation on both sides gives:

$$[I] = A \times e^{C \times t}, \tag{4.29}$$

where A is a constant determined by $[I]_0$. Thus

$$[I] = [I]_0 \times e^{C \times t}. \tag{4.30}$$

When t = 0, $[A] = [I]_0$. C is a measure of how quickly the number of H1 insertion sites double ($\frac{1}{C}$ is the slope of the plot comparing relative Initiator concentration to $ln(t_{10\%})$). This value is derived as follows:

$$[I](t_{10\%}) = [I]_0 \times e^{(C \times t_{10\%})}, \tag{4.31}$$

and

$$0.1[H1]_0 = [I]_0 \times e^{(k \times [H1]_0 \times t_{10\%})}. \tag{4.32}$$

Now we can divide both sides by $[I]_0$ and take the natural log of both sides to get:

$$\ln(\frac{0.1[H1]_0}{I_0}) = (k \times [H1]_0 \times t_{10\%}). \tag{4.33}$$

We performed a linear fit on $\ln(\frac{0.1[H1]_0}{I_0})$ and the 10% completion time on our spectrofluorimetry data (Figure 4.8). The slope of this line is $\frac{1}{C}$ or $\frac{1}{k \times [H1]_0}$ where k is the reaction rate constant. The slope of this line is 0.3667 hours or 1320 seconds. [H1]$_0$ in these experiments is 150 nM. Therefore:

$$k = \frac{1}{slope \times [H1]_0}, \tag{4.34}$$

$$k = \frac{1}{1320 \times 1.5 \times 10^{-7}}, \tag{4.35}$$

$$k = 5050 M^{-1} sec^{-1}. \tag{4.36}$$

This is the value used in our ordinary differential equation simulation discussed in Section 4.5.

4.6 Methods to Generate Other Behaviors

4.6.1 Division

Just as a polymer can grow in logarithmic time via parallel insertion, a population of polymers can be generated in logarithmic time using insertional division. Division is implemented by a complex that is identical in sequence to Hairpin 1 except that its loop has a break in it. When this complex inserts itself into a chain, the polymer splits into two. Figure 4.15 illustrates the general scheme and its implementation in DNA sequences.

We confirmed that monomer conversion is logarithmic in time at two different concentrations of Divide complexes (Figures B.13, B.15). We observe division of polymers when the Divide complex is added to the reactions six hours after initiation. We also observe short polymers when the Divide complexes are added to the solution at the beginning of the reaction, in which case they directly compete with exponential growth (Figure 4.16, 4.17, 4.18).

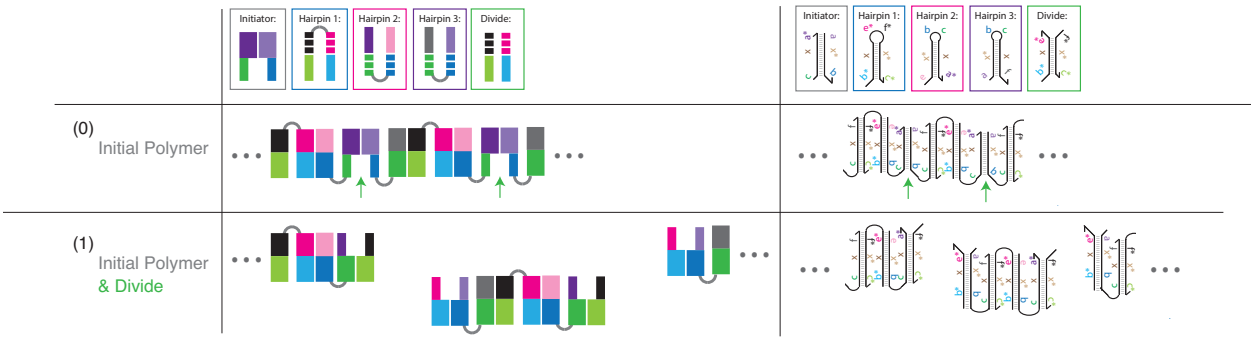


Figure 4.15: This figure depicts a system that implements division in a polymer. Each oligonucleotide is shown with color-coded motifs that correspond to the colored subsequences above. The boxes around each oligonucleotide correspond to the insertion arrows in the mechanism below, which shows the insertion of two Divide complexes. The Divide complex is identical to Hairpin 1, except that the hairpin is split between domains **e*** and **f***. Note that the rest of the hairpins are the same as in Figure 4.1.

4.6.2 Treadmilling

When linear insertion is combined with end-point division, one behavior that emerges is "treadmilling". Treadmilling is the condition in which there is growth at one end of a polymer while the other end is shrinking. Figure 4.19 shows a mechanism for treadmilling using the insertion system presented here. Note that we have not experimentally verified treadmilling. A successful implementation of this mechanism would require careful kinetic control over the insertion and division primitives. The next chapter addresses how such kinetics may be controlled via DNA sequence design.

4.7 Discussion

This work presents an advance in our ability to manipulate matter. It is part of a growing push in nanotechnology and material science toward fabricating smart materials that can be programmed to interact via molecular reactions, thus rendering them capable of being interfaced with biological compounds.

We have used molecular insertion to demonstrate the first synthetic linear polymer that grows in logarithmic time. We have presented a model in Chapter 3 that maps directly onto our molecular

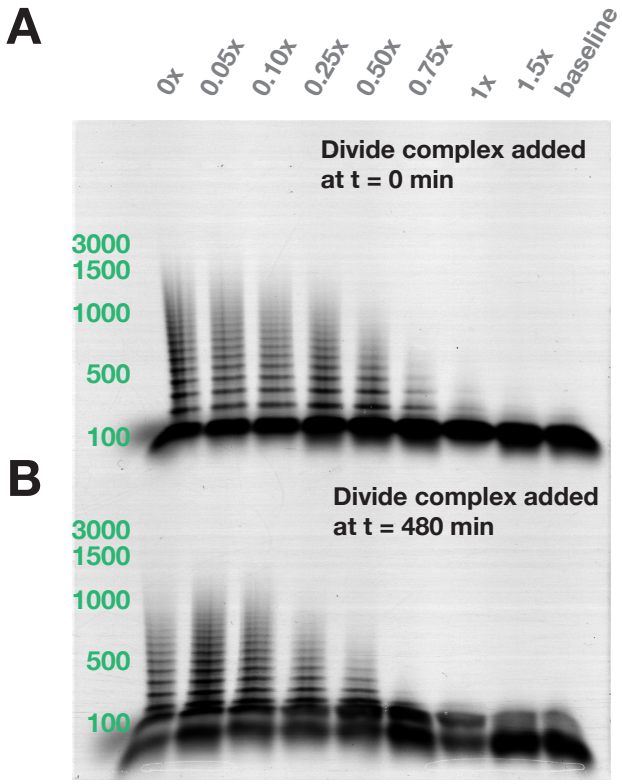


Figure 4.16: Polymer division. Super Fine Resolution Agarose non-denaturing gels of the product of a polymerization reaction with 80 nM ROX-labeled Initiator and 1 μM Hairpin 1, Hairpin 2, and Hairpin 3, to which Divide complex was added at concentrations $[D]_0 = \{0\%, 5\%, 10\%, 25\%, 50\%, 75\%, 100\%, 150\%\}$ relative to hairpin concentrations. (A) Divide complex was added with the hairpins at $t = 0$ min. (B) Divide complex was added after 6 hours of reaction.The size of polymers decreases with increased concentrations of Divide complex. See Figure B.11 for gels after staining with SYBR Gold.

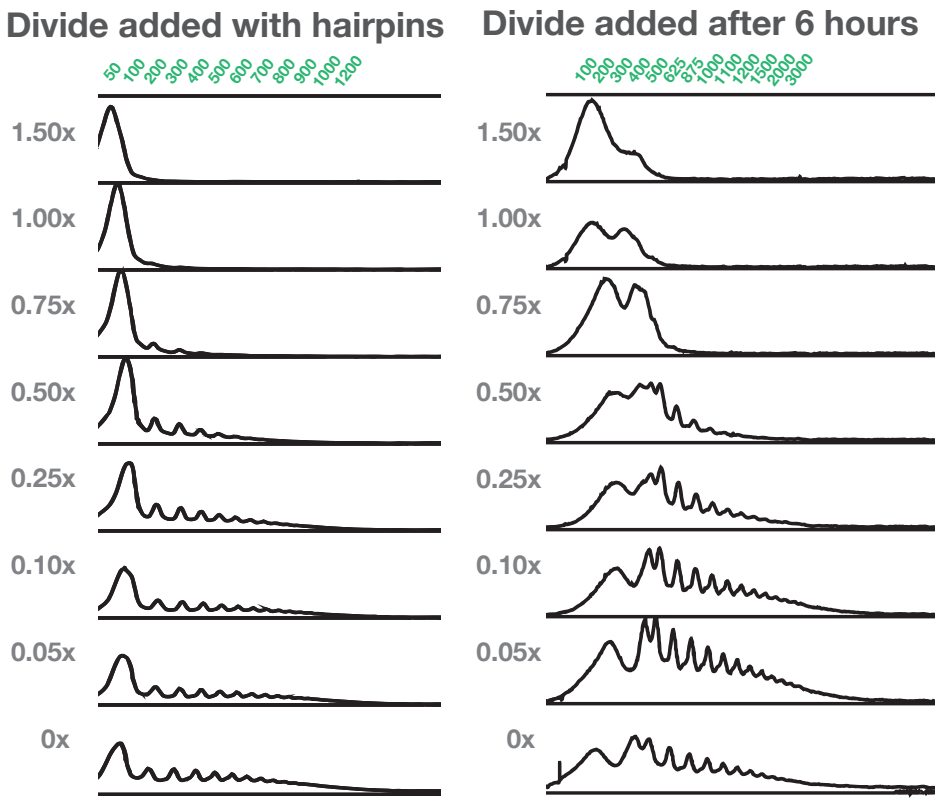


Figure 4.17: Division gel traces of the polymerization system with Divide complex added with hairpins (left) and six hours after hairpins are added (right) from Figure 4.16. Green numbers specify the size of each band, which are indicated as bumps in the gel traces.

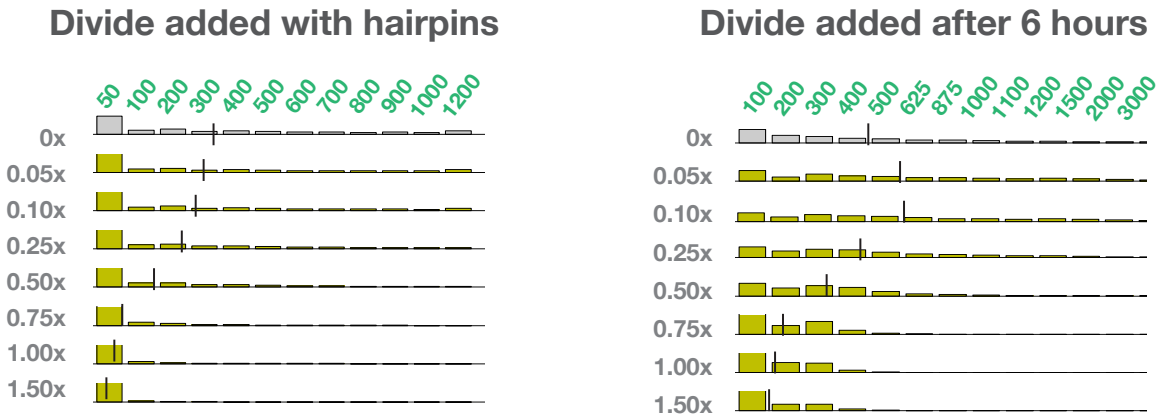


Figure 4.18: Division gel analysis of systems with Divide complex added with hairpins (left) and six hours after hairpins are added (right). The traces from Figure 4.17 were binned such that each bump in the trace was allotted to one bin. The y-axis for each distribution is 40% of the total concentration to allow the reader to clearly see the change in distribution. Vertical lines indicate the mean of the distribution. The mean of each distribution decreases with increased amounts of the Divide complex.

insertion system. This is a very powerful one-dimensional tool. It allows engineers to change the interconnections of molecules after a shape has been assembled. This is an important step toward fully reprogrammable molecular assembly. We have demonstrated three different types of behavior using a simple insertion primitive. We expect that different combinations of these actions can generate more behaviors.

Ours is not, however, the first exponentially fast growing structure ever synthesized. [Yin et al., 2008] constructed a binary molecular tree out of DNA. Their reaction begins with a root node, each node generates two child nodes in each generation of growth. [Yin et al., 2008] point out that, in the absence of steric effects, a linear increase in the number of node species will yield an exponential increase in the size of the binary tree. In practice, steric effects are always present. Our system is the first to implement parallel insertion and does not rely on adding layers to external edges for growth. This feature of our system allows the exponential growth phase of our system to last longer, as our system is not limited by cubic volume.

The next challenge will be to build reprogrammable molecular shapes in two and three dimen-

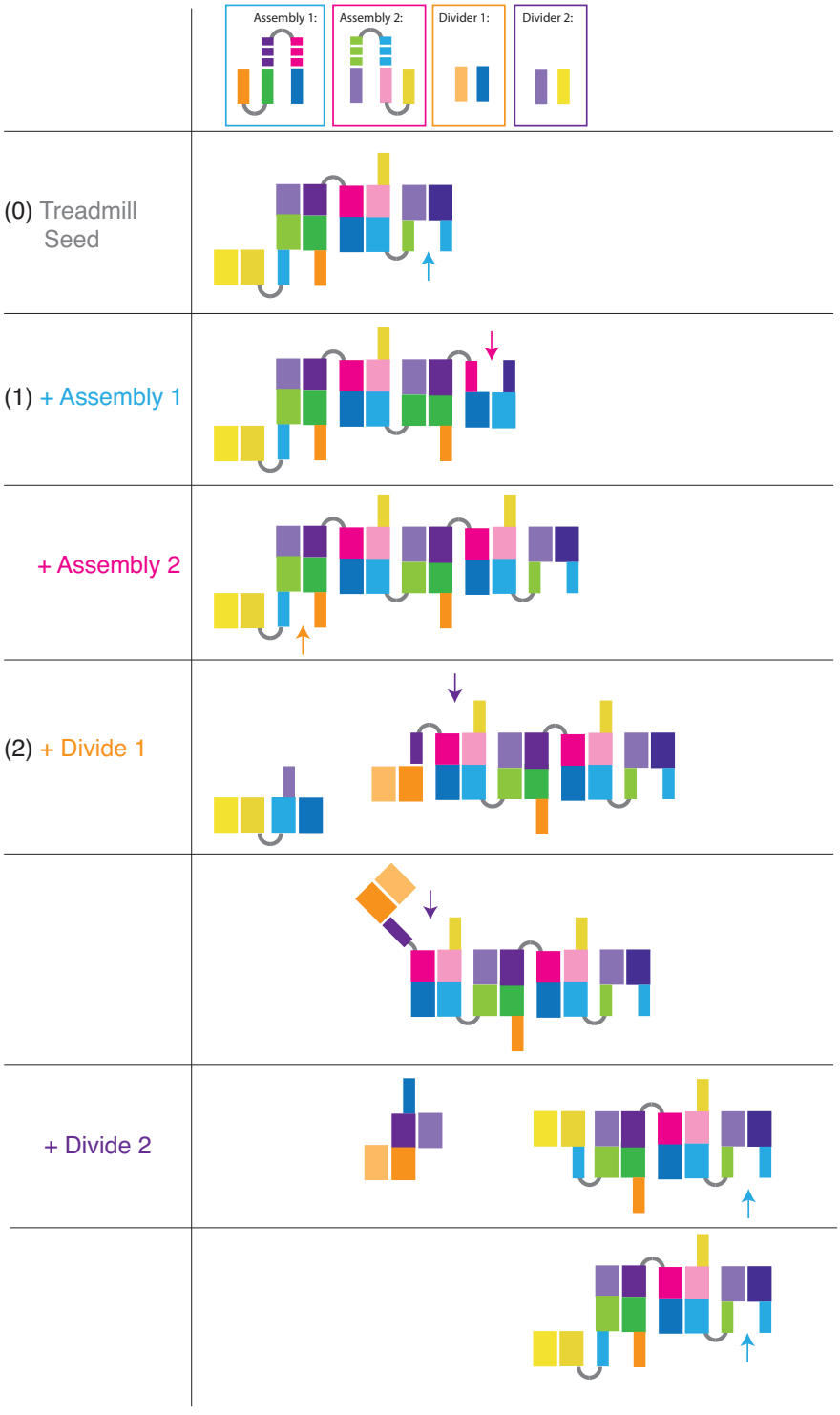


Figure 4.19: Combining the insertion and division behaviors can result in treadmilling, the growth of one end of a polymer while the other end is shrinking.

sions. Difficulties are likely to arise when scaling our current molecular system to these dimensions. Until we can precisely control the kinetics of hairpin insertion (to be discussed in the next chapter), we cannot guarantee the proper exponential growth of a shape in these higher dimensions. This is because our polymer is too flexible to accommodate insertions in multiple dimensions without the possibility of self-interactions forming a mis-shaped object. In order to generate a well-formed object using an elaboration of our system will require a more rigid structure.

A second limitation of our construction is the repeating DNA sequence utilized in the insertion and division primitives. In theory these structures can be programmed just like tiles in the tile assembly model, but in practice the repeating DNA sequence places a constraint on how many different actions can take place at a given site. The power of our system lies in its ability to grow a structure very quickly with only a few types of monomers by allowing subsets of molecules to move relative to each other. When a system like ours is scaled up its power would be limited, because Brownian motion drives these translocations only on small scales.

One may be able to extend this system by adding more complexity into the hairpin loops– additional structures or sequences that might accommodate other functionalities.

4.8 Materials and Methods

Experimental System. A typical fluorescence kinetics experiment contains Hairpin 2 labeled with a fluorophore and quencher pair on the 3' and 5' ends of the strand, respectively. Mixed together with H2 are I, H1 and either the inactivated or regular version of H3 for the linear and exponential systems respectively. H1 is added last to trigger the reaction. As H2 is integrated into the polymer, the quencher and fluorophore pair are separated from each other, yielding an increased fluorescence signal in the solution. At the end of the experiment, another strand of DNA is added into the solution in order to fully displace all unreacted hairpins (Figure 4.4). This "displacement" strand was added in $> 50\times$ excess to the concentration of H2RQ to ensure that the reaction quickly goes to completion. We use the final fluorescence level to normalize our fluorescence signals. Baseline reactions contain only I, H2 and H3, until the end of the experiment at which point the displace strand is added.

DNA Sequences and Design. The sequences presented in Supplementary Tables B.1 are based on those used in a previous insertional polymerization motor [Venkataraman et al., 2007]. These sequences were designed using the NUPACK web application [Zadeh et al., 2010, Zadeh et al., 2011] and our in-house DNA Design software package [Winfree, 2012] to minimize the presence of any unanticipated secondary structures that might interfere with the kinetics under investigation.

Buffer Conditions. DNA oligonucleotides were stored in $1\times$ SPSC buffer (50 mM Na2HPO4 pH 6.5, 1 M NaCl) at $4°$C directly preceding experiments. All experiments and purifications were performed at $25°$C.

Annealing. All annealing processes were performed with an Eppendorf Mastercycler Gradient thermocycler. The samples were brought down from $95°$C to $16°$C at a constant rate over the course of 90 min.

Snap Cooling. All Hairpins were snap cooled prior to experiments. This protocol entails heating the strand solution to $90°$C for 5 min, then immediately putting solutions on ice for 45 min. This protocol encourages intramolecular hydrogen bonding of the hairpins.

Substrate Purification. DNA oligonucleotides used in this study were purchased from Integrated DNA Technologies (IDT), with standard desalting purification, except for strands with a quencher, fluorophore or a $5'$ toehold involved in the four-way branch migration, which were purchased with HPLC purification.

Concentrations of individual strand stocks were determined from the measured absorbance at 260 nM using a Nanodrop Biophotometer and using calculated extinction coefficients that account for hypochromicity effects in double-stranded DNA [Tataurov et al., 2008].

Initiator and Divide complexes were further purified by nondenaturing (ND) polyacrylamide gel electrophoresis (PAGE) as follows: Strands for each sample were prepared with nominally correct stoichiometry at 10 nM and annealed. The acrylamide (19:1 acrylamide:bis) was diluted from 40% acrylamide stock (Ambion). ND loading dye (containing Bromphenol Blue in 50% glycerol) was added to all samples, achieving a final gycerol concentration of 10% by volume. The samples were then run on 12% ND PAGE at 150 V for 6 hours. Gels were run at room temperature ($\approx 25°$C). The band corresponding to the Initiator size was cut out and eluted in 1 mL of $1\times$ SPSC buffer for 2 days. Purified complexes were quantitated by measurement of absorbance at 260 nm

using an Eppendorf Biophotometer and calculated extinction coefficients as above.

Gel Assays. Combinatorial gels were run using 12% polyacrylamide and concentrations of all species at 100 nM. Solutions were left to react for 6 hours, then run in an XCell SureLock Mini-Cell Electrophoresis vertical gel box at 150V for 1 hour in TBE running buffer. After a gel was run, it was stained with SYBER Gold dye and imaged using an FLA-5100 fluorescent scanner (Fujifilm Life Science). Time Lapse, Final Value and Divide gels were run in 2% Super Fine Resolution Agarose (from AMRESCO) on a Thermo Scientific Owl Horizontal Gel box. In these experiments, the Initiator is tagged with a 3' ROX fluorophore on one strand. Thus each properly-initiated polymer has a single ROX tag. Time Lapse reactions contained the following concentrations of species [I] = 80nM, [H1] = 1.5 μM, [H2] = 1 μM, [H3] = 1 μM. Final Values reactions contained the following concentrations of species [I] = 0 nM, 10 nM, 20 nM, 40 nM, 80 nM, 160 nM, 320 nM, 640 nM, 1μM; [H1] = 1.50 μM; [H2] = 1 μM; [H3] = 1 μM. Divide reactions contained the following concentrations of species [I] = 80nM; [D] = 0 nM, 50 nM, 100 nM, 205 nM, 500nM, 750nM, 1 μM, 1.5 μM; [H1] = 1 μM; [H2] = 1 μM; [H3] = 1 μM.

Atomic Force Microscopy. Atomic Force Microscopy images of polymer taken with 10% Initiator (10 nM) relative to Hairpin (100 nM). 50μL of 1$\times$ TAE 12.5 mM Mg^{++} was deposited on mica (from Ted Pella), followed by 1 μ L of 5 mM Nickel Acetate and 2uL of 500 nM polymer sample after 5 hours of reaction. The sample was then imaged using a VEECO Nanoscope III with a vertical engage J-scanner.

Spectrofluorimetry Studies. Spectrofluorimetry studies were done using a SPEX Fluorolog-3 (Horiba) with external water bath and 1.6 mL synthetic quartz cells (Hellma 119-004F). The excitation was at 584 nm, while emission was at 604 nm. In all spectrofluorimetry experiments, the total reaction volume was 1.5 mL, the temperature was 25°C, and 2 nm band-pass slits were used for both excitation and emission monochrometers. Experiments were conducted with an integration time of 10 seconds at 60 second intervals. Prior to each experiment, all cuvettes were cleaned as follows: each cuvette was rinsed 15 times in Milli-Q water, 5 times in 70% ethanol, another 15 times in Milli-Q water, and finally once more in 70% ethanol and then Milli-Q water. For the slit size, concentrations, and integration times used, no measurable photobleaching was observed. Exponential and linear reactions contained the following concentrations of species [I] = 0 nM, 1

nM, 2 nM, 4 nM, 8nM, 16nM, 32 nM, 64 nM, 100 nM; [H1] = 150 nM; [H2] = 100 nM; [H3] = 100 nM. Divide reactions contained the following concentrations of species [I] = 10nM, 25 nM; [D] = 0 nM, 1 nM, 16 nM, 100 nM; [H1] = 150 nM; [H2] = 100 nM; [H3] = 100 nM.

Fluorescence Normalization. Fluorescence is normalized so that one normalized unit of fluorescence corresponds to 1 nM of unquenched fluorophore-labeled strand reporter 2. This normalization is based on the fluorescence levels of annealed samples with a minimal fluorescence measurement taken of the diluted Reporter complex before the experiment was initiated, and a maximal fluorescence value that is extracted from a biexponential fit of the data taken at the end of the experiment, after the displacement strand is added to displace all unreacted fluorophore-quencher pairs.

Chapter 5

The Kinetics of Toehold-Mediated Four-way Branch Migration[0]

5.1 Abstract

DNA nanotechnology has enabled the implementation of switches, circuits, motors, assembly and amplification using three simple mechanisms: hybridization, three-way branch migration and four-way branch migration. In particular, four-way branch migration, the process by which two double-stranded molecules of DNA that share the same stem sequence simultaneously exchange strands, demonstrates novel capabilities that expand the design-space of what can be engineered using DNA. This mechanism allows molecules to rearrange or insert into a chain without dissociating, and can be initiated conditionally by two matching toeholds (short sequences of DNA that are complementary to single-stranded domains in a target molecule). Because sequences are sequestered by their complement, toehold-mediated four-way branch migration should enable the design of circuits with less crosstalk between strands that are not supposed to interact, as compared with toehold-mediated three-way branch migration. Four-way branch migration has been used to perform directional motion, program molecular walkers and to design efficient molecular probes. We

[0]This work was coauthored by Nadine Dabby, Ho-Lin Chen, Joseph Schaeffer, & Erik Winfree* and is currently in submission [Dabby et al., 2013] with the following contributions: all experiments were performed by N. D.; trajectory simulations were performed by J.S., analysis was performed by N. D., H-L.C., and J.S with supervision by E.W. Experimental design and manuscript was done with input from all authors.

have found that by designing the toeholds involved in a four-way branch migration reaction, we can control the effective reaction rate over at least seven orders of magnitude. We characterize the kinetics of DNA toehold-mediated four-way branch migration using fluorescence spectroscopy, and derive a mechanistic model that can be used in the design of four-way branch migration reactions. The ability to control the kinetics of these reactions will greatly facilitate the programming of dynamic behaviors mediated by four-way branch migration.

5.2 Introduction

DNA enables the construction of dynamic technologies with very simple chemical mechanisms: hybridization and disassociation [Wetmur, 1976, SantaLucia, 1998], strand displacement [Yurke et al., 2000] and four-way branch migration [Panyutin and Hsieh, 1994]. We seek to understand the biophysics that underlies these fundamental mechanisms. By encoding the order of the nucleotides in a sequence, we can control the hydrogen bonding and thus the interaction of DNA strands. Hybridization and disassociation have been well characterized [Wetmur, 1976, SantaLucia, 1998]. Branch migration is the process by which a duplex of DNA exchanges one or two of its strands for new strands with identical sequences; three-way and four-way branch migration are named according to the number of strands involved in the mechanism. Nature uses four-way branch migration powered by proteins to generate genetically varied DNA from one generation to the next.

Dynamic systems of DNA molecules can be controlled by toeholds, the short sequences of DNA that are complementary to single stranded domains in a target molecule [Yurke et al., 2000, Zhang and Winfree, 2009]. We call systems controlled in this manner "toehold-mediated". Many developments in DNA nanotechnology rely on toehold-mediated three-way branch migration to implement switches [Lubrich et al., 2008, Simmel and Yurke, 2002, Yan et al., 2002, Yurke et al., 2000], circuits [Seelig et al., 2006, Yin et al., 2008, Zhang et al., 2007], motors [Gu et al., 2010, Omabegho et al., 2009, Shin and Pierce, 2004, Yin et al., 2008], assembly [Dirks and Pierce, 2004, Lubrich et al., 2009, Yin et al., 2008] and amplification [Dirks and Pierce, 2004, Zhang et al., 2007].

The kinetics and mechanism of three-way branch migration [Green and Tibbetts, 1981, Pa-

nyutin and Hsieh, 1994] and toehold-mediated three-way branch migration [Yurke and Mills, 2003, Zhang and Winfree, 2009] have been thoroughly characterized. The use of toeholds to mediate and control the process of strand displacement allows for sequence specific targeting of DNA fuels to DNA nanodevices that can be cycled through multiple states [Yurke et al., 2000]. The kinetics and mechanism of three-way branch migration has been thoroughly characterized [Green and Tibbetts, 1981, Zhang and Winfree, 2009]. Three-way branch migration reaction rates can be controlled over six orders of magnitude by changing the length [Yurke and Mills, 2003] and strength [Zhang and Winfree, 2009] of the toeholds that initiate the mechanism. One proposed model for toehold-mediated three way branch migration breaks down the mechanism into three steps: the hybridization of the free strand to the toehold domain, the branch migration process, and a final step in which the invading strand completely displaces the incumbent strand to create two separate molecules [Zhang and Winfree, 2009] .

Four-way branch migration is the process by which two double-stranded oligonucleotides that share the same stem sequence simultaneously exchange strands. Four-way branch migration has enabled the implementation of a DNA actuator [Zhang and Seelig, 2011]. Toehold-mediated four-way branch migration, initiated by unpaired toeholds that bind together to form an intermediate structure called a Holliday junction, has been used to implement molecular robots [Muscat et al., 2011], directional motors [Venkataraman et al., 2007] and molecular probes [Duose et al., 2012].

Four-way branch migration differs from three-way branch migration in its ability to implement mechanisms via two input toeholds as opposed to only one. The structure of complexes capable of undergoing four-way branch migration subserves an implicit AND function. [Duose et al., 2012] demonstrate the utility of this feature by implementing a molecular probe with a toehold-mediated four-way branch migration design that leaves all waste products double-stranded, or inert. The four-way construction is an efficient alternative to displacing multiple strands on a single probe via toehold-mediated three-way branch migration, because it reduces the number of complexes that must be added. The sequestering of DNA sequences in four-way branch migration systems also result in less cross-talk between single-stranded domains thereby making possible a slew of less noisy designs. The physical process of four-way branch migration enables the insertion or rearrangement of DNA molecules without requiring the complete disassociation of a single strand.

The mechanism is thus capable of allowing the implementation of some novel nanotechnology tools that expand the space of what can be engineered using DNA; such as the ability to implement insertion [Venkataraman et al., 2007] which has not been accomplished to date using three-way branch migration.

The full capabilities of four-way branch migration, and the means for their kinetic control have yet to be fully explored. [Panyutin and Hsieh, 1994, Thompson et al., 1976] have characterized the kinetics of the individual steps within four-way branch migration independent of toehold length, as a function of temperature and ionic conditions. The structural dynamics [McKinney et al., 2002, McKinney et al., 2005, Karymov et al., 2008] and thermodynamics [Seeman and Kallenbach, 1994] of the Holliday junction, have also been studied. As of yet, there has been no characterization of toehold-mediated control of four-way branch migration.

In pursuit of more scalable and robust DNA systems, we seek better control over the kinetics of strand displacement, to minimize leaks and maximize the ability to multiplex inputs into our DNA systems. We experimentally characterize how toehold length and strength can be used to control the speed of toehold-mediated four-way branch migration. We have found that by designing the toeholds involved in four-way branch migration, we can control the effective reaction rate over seven orders of magnitude. We used both thermodynamic and kinetic modeling to arrive at a mechanistic model for toehold-mediated four-way branch migration with four fit parameters. Our model assumes two main phases in the reaction: a bimolecular interaction step in which the two molecules hybridize to form one complex, and a unimolecular phase in which the complex undergoes branch migration. This work aims to improve our general understanding of the mechanism of four-way branch migration by providing a model that predicts how toeholds can be designed to control the kinetics of these reactions.

5.3 System Description

Single-stranded molecules of DNA (henceforth strands) are comprised of concatenated domains, stretches of consecutive nucleotides that act as a unit in binding to complementary stretches of nucleotides. Domains are represented by Latin letters; an asterisk denotes complementary do-

mains, e.g.: x is complementary to x* (Figure 5.1). DNA complexes are composed of two or more noncovalently-bound strands. The experimental reaction undergoes a bimolecular phase during which two double stranded molecules, the "Reporter" and the "Complex", join together to form the "Reporter-Complex" intermediate via hybridization of their toeholds, followed by a unimolecular phase during which branch migration is completed and the intermediate separates into two products, the "m-product" and the "n-product" (Figure 5.1A).

We characterized how toehold length and strength can be used to control the kinetics of a toehold-mediated four-way branch migration. We performed experiments with a series of DNA molecules differing from each other by only a few bases in their toehold regions. The two toeholds in the Complex are labeled **m** (for the 5' toehold on the complex) and **n** (the 3' toehold on the complex), in Figure 5.1A. The toeholds **m** and **n** bind to Reporter toeholds **m*** and **n*** respectively. The binding of one or both toeholds initiates a four-way branch migration between the two complexes; we call this process a toehold-mediated reaction.

For long enough toeholds every molecular collision will lead to complete branch migration, but for shorter toeholds only a fraction of collisions will be effective. The Reporter has toeholds of length **n*** = 6 and **m*** = either 6 or 16 in all experiments (we use a bracket notation to indicate the alternative lengths of each toehold, e.g. **m*** = $\{6; 16\}$). The Reporter is labeled with a fluorophore-quencher pair on opposite strands that are separated after the branch migration reaction with the Complex completes. This allows us to trace the kinetics of the experiment as the fluorescence of the bulk reaction increases over the course of the reaction. The Complex has toeholds of variable lengths (**m** = $\{0; 2; 4; 6; 16\}$ and **n** = $\{0; 2; 4; 6\}$. All sequences can be found in Figure 5.1 and Supplementary Table C.1.

5.4 Experimental Data

We fit experimental data in the toehold-mediated four-way branch migration process to the phenomenological model shown in Figure 5.1A. The rate constant $k_1(m, n)$ denotes the bimolecular rate of intermediate formation between the Reporter and Complex for ultimately successful reactions. The rate constant $k_2(m, n)$ denotes the unimolecular rate for completing the branch

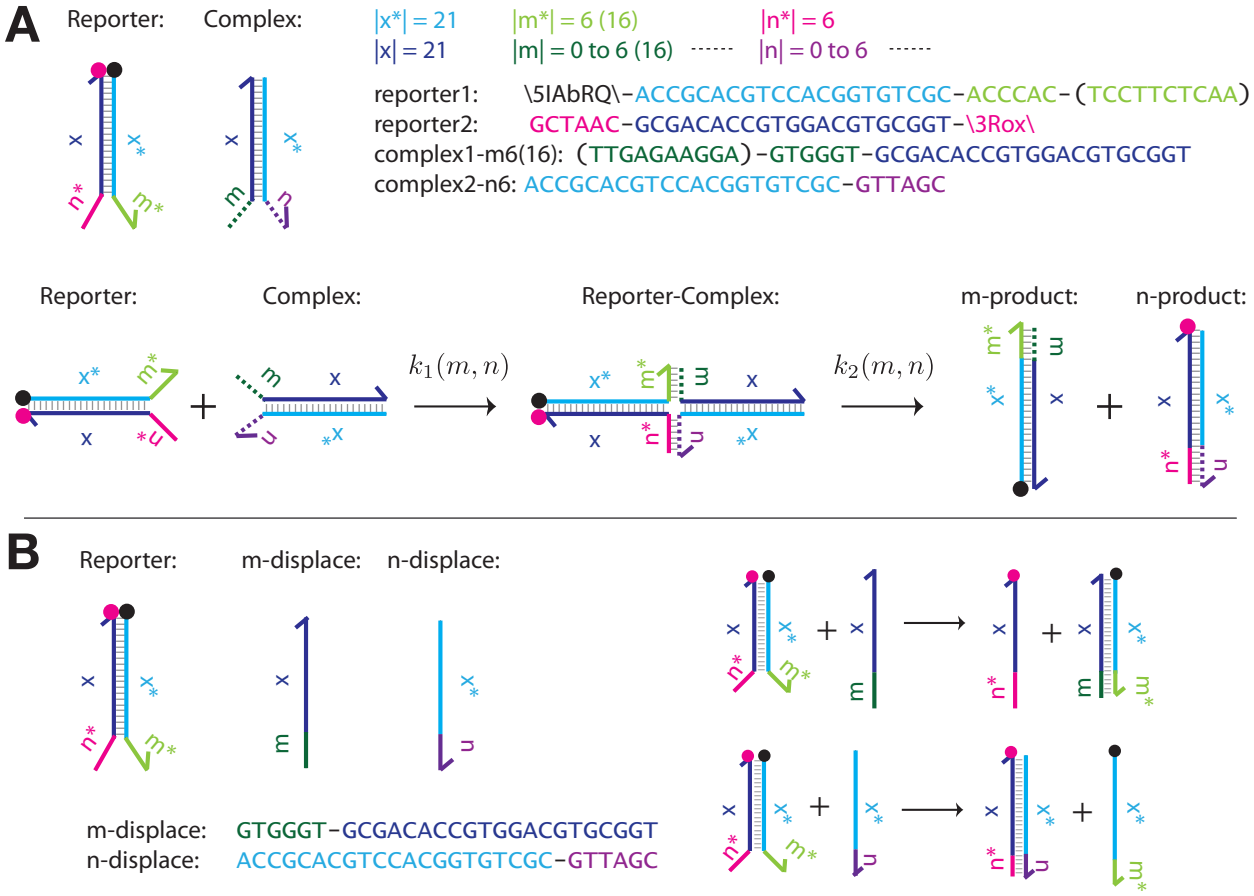


Figure 5.1: (A) A typical fluorescence kinetics experiment contains a Complex with toeholds (m, n), and a Reporter (labeled with a fluorophore and quencher pair on opposite strands) mixed together in solution. As the Complex and and Reporter exchange strands, the fluorophore and quencher pair on the Reporter are separated from each other yielding an increased fluorescence signal in the solution. Sequences for DNA strands are color-coded by domain, \5IAbRQ\ indicates a 5' Iowa Black red quencher modification, \3Rox\ indicates a 3' ROX fluorophore modification. (B)At the end of the experiment another strand of DNA is added into the solution in order to fully displace all unreacted quencher strands on the Reporter. As above, sequences are color-coded by domain.

Fitted Four-way Unimolecular Rates		
toehold length	average (sec^{-1})	standard deviation
k_2^{open}	4.4×10^{-4}	4.5×10^{-5}
k_2^{closed}	1.5×10^{-3}	4.7×10^{-4}

Table 5.1: Unimolecular rates k_2^{open} and k_2^{closed} determined by mean squared error fitting to experimental traces. We assume that the bimolecular rate constant k_1 is $k_f = 3.0 \times 10^6$ M^{-1} sec^{-1}.

migration process that results in the m-product and n-product.

We first attempted to isolate $k_2(m, n)$ by examining the kinetics of toehold combinations in which the unimolecular reaction was the rate-limiting step. We used the toehold combinations $(m = 16, n = 0)$, $(m = 16, n = 2)$, $(m = 16, n = 4)$, $(m = 16, n = 6)$ to this end, and assumed that the bimolecular step when a 16-base toehold is present is limited by the rate of hybridization $k_f = 3.0 \times 10^6$ with a negligible backward reaction. This value for k_f was measured for DNA strands in a study on toehold-mediated three-way branch migration [Zhang and Winfree, 2009]. Although three-way branch migration and four-way branch migration are different molecular mechanisms, we presume that in both cases the initial hybridization step is the same, and that the differences in mechanism only affect the probability that a given molecular interaction results in successful branch migration completion.

We discovered that for $m = 16$, all toehold pairs in which $n \geq 2$ achieved the same unimolecular rate constant, within experimental error. There are two cases for k_2: the closed case (wherein $m \geq 2$ and $n \geq 2$), and the open case (wherein $m = 0$ or $n = 0$). The open unimolecular rate, k_2^{open} (4.4×10^{-4} sec^{-1}), was almost an order of magnitude slower than the closed rate, k_2^{closed} (1.5×10^{-3} sec^{-1}). See Figure 5.2A for $(m = 16, n = 2)$ data, and Table C.2 for rates and standard deviations. For individual rates, concentrations used and experimental traces, see Table S2, Table S3 and Figure S1. After finding experimental values for k_2^{open} and k_2^{closed}, we used these values to determine k_1 from experimental traces for each combinatorial pair of (m, n). Each experiment was run at least three times, with at least two concentrations. Concentrations were selected to ensure that the rate of the overall reaction was limited by the bimolecular step rather than the unimolecular step (Table S4).

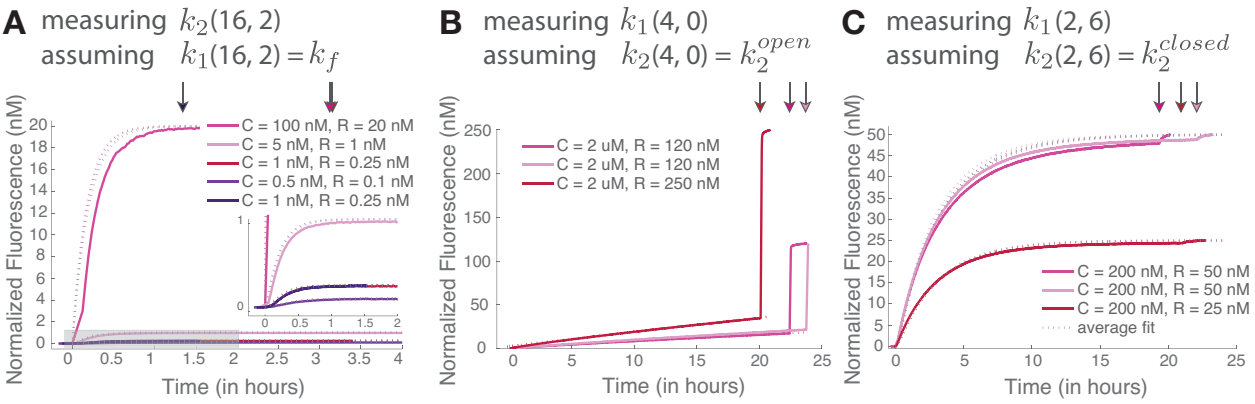


Figure 5.2: The kinetics of the reactions between Reporter and Complexes with kinetics in the (A) fast (B) slow and (C) medium-speed regimes. At the conclusion of each experiment n-displace is added in excess (as indicated by arrows). (A) Traces of $(m = 16, n = 2)$ with zoom-in of low concentration data (shaded region). (B) Traces of $(m = 4, n = 0)$. (C) Traces of $(m = 2, n = 6)$. Solid lines indicate experimental traces, dotted colored lines depict simulation of individual traces, and grey dotted lines show fits of the average fit rate at each concentration.

Combinatorial pairs of (m, n) fell into one of three regimes: slow (Figure 3B), medium (Figure 3C), and fast (Figure 3A). Figure 5.2 shows data and ODE simulations for one set of experiments in each of these regimes. Slow reactions were conducted at high concentrations and failed to achieve a 20% completion rate within 24 hours (Figure 5.2B). Medium-speed reactions were conducted at concentrations on the order of 25 nM and reached completion within 24 hours (Figure 5.2C). Fast reactions were conducted at low concentrations of 5 nM or less, and resulted in greater error, as there were fewer data points to fit (Figure 5.2A). Over these 16 combinatorial pairs of toeholds, we see $k_1^{fit}(m, n)$ rates ranging from less than 0.033 M^{-1} sec^{-1} to 6.9×10^5 M^{-1} sec^{-1} (Figure 3). The data suggest that four-way branch migration can be controlled over at least seven orders of magnitude by utilizing toeholds to mediate the reaction. Further, as was observed with toehold-mediated three-way branch migration, there is a rough correlation between (total) toehold length and the effective rate constant. (All rates and standard deviations can be found in Table 5.2, see Figures S2-S7 for all traces.) The speed of the reactions limits our confidence on the extreme ends: slow reactions were too slow to accurately measure using our methods (these rates may be slower than what we could measure), and fast reactions offered too few data points to accurately measure completion time (these rates may be faster than what we could measure).

Fitted Four-way Bimolecular Rates				
toehold length	$\Delta G^{\circ}_{k_1}$ (kcal mol^{-1})	$k_1^{fit}(m,n)$ (M^{-1}sec^{-1})	$k_1^{calc}(m,n)$ (M^{-1}sec^{-1})	$k_1^{sim}(m,n)$ (M^{-1}sec^{-1})
$m=0, n=0$	2.41	0.034 ± 0.0071	1.1×10^{-5}	≤ 0.099
$m=2, n=0$	0.57	0.047 ± 0.015	2.4×10^{-4}	≤ 0.35
$m=2, n=2$	-0.21	0.10 ± 0.0063	0.045	≤ 0.61
$m=0, n=2$	-1.34	0.033 ± 0.0023	5.9×10^{-3}	≤ 0.36
$m=2, n=4$	-1.82	0.93 ± 0.088	0.68	2.29 ± 1.3
$m=0, n=4$	-2.95	0.039 ± 0.0032	0.089	≤ 0.51
$m=4, n=0$	-4.02	0.97 ± 0.23	0.54	0.65 ± 0.645
$m=4, n=2$	-4.80	56 ± 1.4	100	92 ± 10.45
$m=2, n=6$	-5.64	490 ± 34	430	185 ± 117
$m=6, n=0$	-6.24	58 ± 6.8	23	7.30 ± 2.475
$m=4, n=4$	-6.41	770 ± 88	1.6×10^3	$5.20 \times 10^3 \pm 640$
$m=0, n=6$	-6.77	5.0 ± 0.028	56	24 ± 4
$m=6, n=2$	-7.02	$9.4 \times 10^3 \pm 3.5 \times 10^3$	4.4×10^3	$2.59 \times 10^3 \pm 640$
$m=6, n=4$	-8.63	$7.0 \times 10^4 \pm 3.0 \times 10^4$	6.4×10^4	$1.59 \times 10^5 \pm 1.925 \times 10^4$
$m=4, n=6$	-10.23	$2.8 \times 10^5 \pm 1.7 \times 10^5$	7.4×10^5	$1.66 \times 10^5 \pm 2.055 \times 10^4$
$m=6, n=6$	-12.45	$6.9 \times 10^5 \pm 3.5 \times 10^5$	2.8×10^6	$4.63 \times 10^5 \pm 2.09 \times 10^4$

Table 5.2: Bimolecular rates k_1^{fit} determined by mean squared error fitting to experimental traces, with comparison to numerical models. We assume the unimolecular rate is $k_2^{open} = 4.4 \times 10^{-4}$ sec^{-1} in the case where only one toehold is present on the intermediate complex, and $k_2^{closed} = 1.5 \times 10^{-3}$ sec^{-1} when two toeholds are present on the intermediate complex. The unimolecular rate was not utilized in cases where the experiments did not reach a 20% completion level within 24 hours, as we fit these rates linearly. Calculated reaction rates, k_1^{calc}, are predicted by the mechanistic model (see Section 5). Simulated reaction rates, k_1^{sim}, were calculated from between 200 and 10^6 Multistrand trajectories and scaled uniformly to best fit the data as described in Section 4 and the Materials and Methods section. Experimental reaction rates are shown with the standard deviation of independent fits to three or more experimental trajectories, taken with at least two different reactant concentrations. Simulated reaction rates are shown with an estimate of the standard error of the mean. Values that are upper-bounded indicate reactions that did not have a single successful trajectory over all simulations.

Each experiment was concluded with the addition of 20 μL of a displacement strand to the reaction solution for a final concentration of 6.3 μM of displacement strand (this amount is at least $50\times$ the concentration of reporter in each cuvette, and results in a dilution factor of 1.3% for the 1500 μL solution). We were able to use the maximum fluorescence of a fully consumed Reporter to normalize the fluorescence data such that the maximum signal is equivalent to a reaction completion level of 100%. We noted a lack of completion in the faster reactions with very low concentrations. In the $(m = 4, n = 6)$, $(m = 6, n = 2)$, $(m = 6, n = 4)$, and $(m = 6, n = 6)$ experiments, we set the 100% completion level to the maximum fluorescence signal that was recorded before the displacement strand was added. We traced this effect back to the age of each complex and observed that older complexes reached lower levels of completion. Across all batches, the variance of fitted rates is not correlated with completion level, independent of the Reporter's age. We are confident that our fits of these traces to the lowered completion level does not affect the results within the standard deviation reported (see Figures S4-S7).

We observed a disparity in experimental reaction rates between the m and n toeholds. For example, $(m = 0, n = 4)$ has a k_1^{fit} of 0.039 $\text{M}^{-1}\text{sec}^{-1}$, while $(m = 4, n = 0)$ has a k_1^{fit} of 0.97 $\text{M}^{-1}\text{sec}^{-1}$. This may result from sequence differences, toehold interference or small unanticipated secondary structures in the toeholds. Even though both the m and n toeholds were designed to have equal GC content, this changes as the sequences are truncated (Table S1).

The kinetics of the displacement reaction when using the n-displace strand were 2 orders of magnitude slower than when using the m-displace strand (see Table S5 for concentrations used, Figure S8 for experimental traces). These results are consistent with our observation that the same toehold lengths can differ in their rates by more than an order of magnitude when present on the m-toehold versus the n-toehold. Negligible rate differences of the displacement reaction in the presence or absence of the other toehold gives us confidence that the presence of both full-length toeholds on the Reporter did not interfere with the experiments (Figure S8).

We gauged the significance of a reverse reaction in which additional toeholds were present on the opposite side of the m-product and n-product. We tested the case where the m-product and n-product have one complementary toehold, y, of three bases in length (Figure S9 illustrates the experimental set-up and Tables S6 and S7 list the sequences and concentrations used). In this case

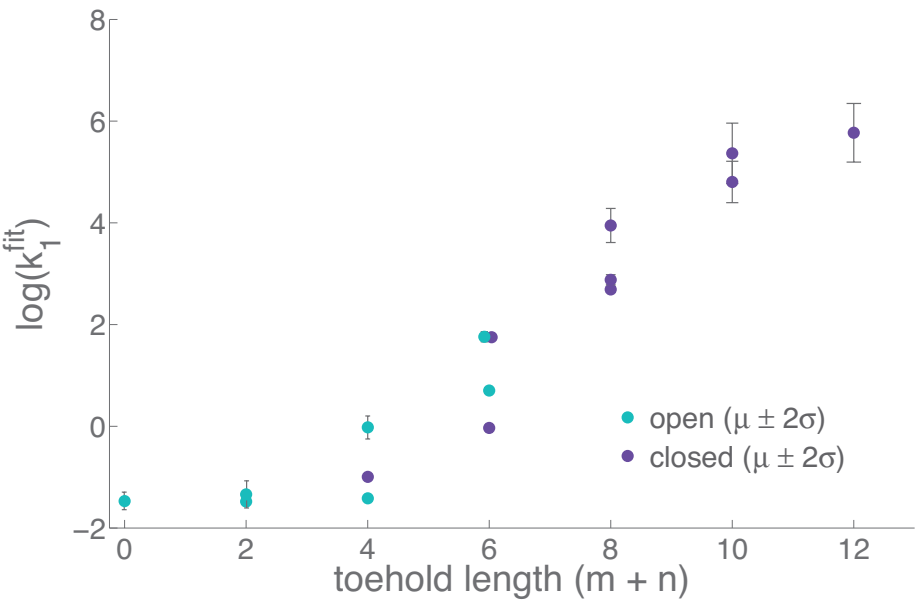


Figure 5.3: Plot of toehold length $(m + n)$ versus $\log(k_1^{fit})$, showing the correlation between the sum of toeholds m and n to the experimentally fit mean k_1 rate constants. Dots correspond to open (blue) and closed (purple) loop reactions. Error bars show two standard deviations from the mean.

the reactions most likely to go backward are $(m = 0, n = 0)$, $(m = 0, n = 2)$, $(m = 2, n = 0)$. The reverse reactions in the smallest toehold cases were negligible over the course of three days (Figure S10 and Table S8). Since these were the most energetically favorable of all of the reverse reactions and all experiments were conducted over a period of less than 24 hours, we are confident that reverse reactions should have no measurable effect on estimated rate constants.

The experimental data (Table 2) corroborates the hypothesis that toehold length can be used to control the rate of toehold-mediated four-way branch migration (Figure 5.3). However, the data also indicates that for a given sum of toeholds **m** and **n**, there can be a one to two order of magnitude spread of bimolecular reaction rates. Following the observation that toehold free energy is a better predictor of bimolecular reaction rates than toehold length [Yurke and Mills, 2003,Zhang and Winfree, 2009] in toehold-mediated three-way branch migration, we reasoned that similar arguments could better explain the behavior of toehold-mediated four-way branch migration. This required careful consideration of the relevant energy landscape and reaction mechanism steps.

5.5 Energy Landscapes and Elementary Step Simulations

The analysis of toehold-mediated four-way branch migration is complicated by the variety of ways that the two toeholds can initiate (or fail to initiate) the binding process, and by the multiple steps in which base pairs break or are formed during each branch migration step. These features can, in principle, be accounted for by secondary structure models that explicitly track each change in base pairing, while incorporating known thermodynamic and kinetic behaviors of DNA.

DNA secondary structure is the base pairing information within a set of DNA strands. We can assign an energy to a particular configuration of the molecules using the well-known nearest neighbor energy model [Mathews et al., 2004, SantaLucia Jr and Hicks, 2004]. This model breaks down the secondary structure configuration into local components known as loops, which are defined by single stranded regions and their neighboring base pairs. There are several different categories of loops, such as stacks (i.e., two neighboring base pairs with no intervening single stranded regions), hairpins, bulges, and multiloops. We typically consider states in the energy landscape to be adjacent if they differ by exactly one base pairing. Even though the total number of adjacent states to a given state is at most quadratic in the total length of the strands, the entire state space is typically at least exponential in that length. In order to modify the energy model to handle multi-stranded systems, an additional energy term is required to account for the entropic initiation cost of bringing two strands together [Dirks et al., 2007]. For each complex we have an additional energy contribution associated with the entropy of the volume [Schaeffer, 2012]. In this paper, we refer to the specific version of the energy model that we use as "the NUPACK model".

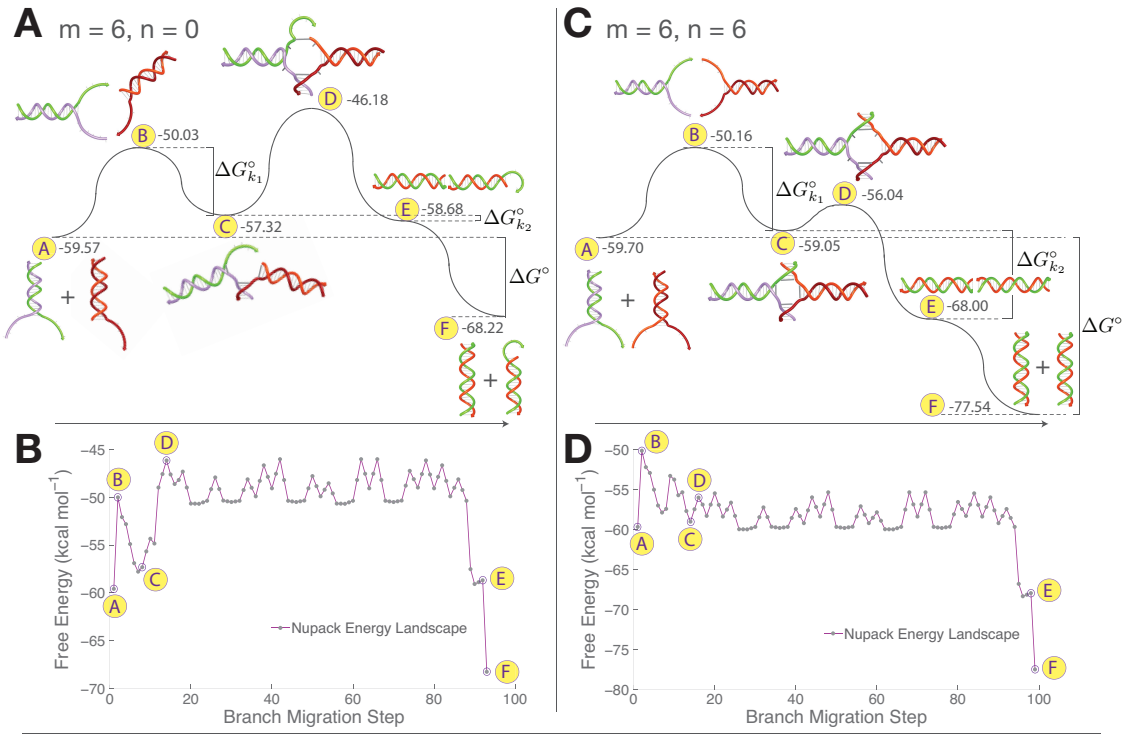
Secondary structure kinetics models can be defined as a continuous time Markov process over a secondary structure state space where states are considered adjacent if they differ by exactly one base pair [Flamm et al., 2000], and the elementary step is the formation or breaking of a single base pair. These kinetics models define the rate of transition between adjacent states using what is usually called a rate method. For example, the Metropolis rate method defines the rate of energetically favorable steps as being 1 (in arbitrary time units) and the rate of unfavorable steps as the negative exponential of the energy difference between the two states [Metropolis et al., 1953], thus ensuring that the equilibrium probabilities are consistent with the Boltzmann distribution using the

thermodynamic energies. However, thermodynamics does not determine kinetics: the rates for any step and its reverse can be arbitrarily scaled by the same amount without affecting the equilibrium distribution. Using this principle, the simulated kinetics can be calibrated to experimental data by uniformly scaling all steps.

To simulate toehold-mediated four-way branch migration, we used the Multistrand software, which simulates elementary-step secondary structure kinetics of a finite number of DNA molecules interacting within a finite volume, effectively performing random walks on the energy landscape of the system [Schaeffer, 2012]. Employing Multistrand's "first step" mode, every simulation was started with a bimolecular base-pair formation step that occurs between the Reporter and the Complex and was followed until one of two distinct end states was reached: the Reporter and Complex falling apart, or the molecules reacting into the m-product and n-product. Each trajectory ends in one of two states (non-reactive or reactive). A full simulation provides data on what percentage of initial interactions react to completion and the rate at which reactive or nonreactive collisions occur. That data is then used to calculate the simulated bimolecular reaction rate, k_1^{sim}, between the two complexes [Schaeffer, 2012].

All rates in the Multistrand kinetic model have scaling factors so that the simulated time approximates real time. These scaling factors were calibrated to match duplex formation and three-way branch migration experiments using two fitting parameters [Schaeffer, 2012]. Given the degree to which kinetics is underdetermined from thermodynamics, we expect the reported times from the simulator to require an additional scaling factor in order to approximate real time for this experimental system (see Materials and Methods). We found that the simulated rates for the complexes, k_1^{sim}, correspond well with experimental fits after being scaled uniformly by a factor of 20 (data shown in Table 2). The secondary structure energy landscape can be used as the basis for understanding toehold-mediated four-way branch migration, but because it is exponentially large, it does not provide a simple understanding suitable for analytic explanations.

We present an intuitive scheme for a reaction pathway in Figure 5.4A and C, where we have abstracted away the entire secondary structure state space into six key states: **(A)** The Reporter and Complex separate within a volume, **(B)** the Reporter and Complex co-localized with no base-pairs between them, **(C)** the Reporter and Complex bound by all available toeholds, **(D)** the initation



(A) $\Delta G_A = \Delta G^\circ_{reporter} + \Delta G^\circ_{complex} + 2 \times \left(RT \ln \frac{V}{V_0} \right)$

(B) $\Delta G_B = \Delta G^\circ_{reporter} + \Delta G^\circ_{complex} + 3 \times \left(RT \ln \frac{V}{V_0} \right)$

(C) $\Delta G_C = \Delta G^\circ_{reporter-complex} + 3 \times \left(RT \ln \frac{V}{V_0} \right)$

(D) $\Delta G_D = \Delta G^\circ_{reporter-complex^\ddagger} + 3 \times \left(RT \ln \frac{V}{V_0} \right)$

(E) $\Delta G_E = \Delta G^\circ_{m-product} + \Delta G^\circ_{n-product} + 3 \times \left(RT \ln \frac{V}{V_0} \right)$

(F) $\Delta G_F = \Delta G^\circ_{m-product} + \Delta G^\circ_{n-product} + 2 \times \left(RT \ln \frac{V}{V_0} \right)$

$$\begin{aligned} \Delta G^\circ_{k_1} &= \Delta G_C - \Delta G_B \\ &= \Delta G^\circ_{reporter-complex} - \Delta G^\circ_{complex} - \Delta G^\circ_{reporter} \end{aligned}$$

$$\begin{aligned} \Delta G^\circ_{k_2} &= \Delta G_E - \Delta G_C \\ &= \Delta G^\circ_{m-product} + \Delta G^\circ_{n-product} - \Delta G^\circ_{reporter-complex} \end{aligned}$$

$$\begin{aligned} \Delta G^\circ &= \Delta G_F - \Delta G_A \\ &= \Delta G_E - \Delta G_B \\ &= \Delta G^\circ_{k_1} + \Delta G^\circ_{k_2} \end{aligned}$$

Figure 5.4: Energy landscapes for toehold-initiated four-way branch migration. (A) and (C) show example diagrammatic energy landscapes for the open case where $(m = 6, n = 0)$ and for the closed case where $(m = 6, n = 6)$. The free energy differences $\Delta G^\circ_{k_1}$, $\Delta G^\circ_{k_2}$ and the overall ΔG° are indicated. (B) and (D) show the corresponding elementary step energy landscapes, in which each point indicates the making or breaking of a single base pair. In all energy landscapes, we define the 0 kcal energy reference point as the state in which all four strands of DNA that make up the Reporter and Complex are separated with no base pairs in the system. The energies of each system state are calculated using NUPACK energies adjusted by the entropic penalty $(RT \ln V/V_0)$ as described in the text. (The reader will notice stretches of "flat" steps in the elementary step landscapes, when one might expect an uninterrupted sawtooth diagram; the data plotted reflects the treatment of dangles in the NUPACK model.) Labels correspond to the indicated state along the reaction coordinate; energy equations are shown below.

of branch migration, **(E)** branch migration completed and the two products still co-localized, and **(F)** the two products separate within the volume. This reaction pathway allows us to examine two hypothesized energy barriers: co-localization and initiation of branch migration. We use the secondary structure nearest neighbor model energies of the aforementioned key states to estimate the barriers present in this primary reaction pathway.

Free energies of system states are calculated as follows. A state consists of the secondary structure description for a hypothetical box of volume V containing one copy of each molecule (the four strands of the Reporter-Complex, either together or separate) at an effective concentration of 100 nM. We define the 0 kcal mol^{-1} energy reference point for the system as the state in which all four strands are fully disassociated. We estimate the system energy by summing the nearest-neighbor model energies for the complexes as calculated by NUPACK [Zadeh et al., 2010] and then, as in Multistrand [Schaeffer, 2012], including the entropic penalty $RT \ln(V/V_0) = +9.54$ kcal mol^{-1} for each additional co-localized molecule, where V_0 is the volume occupied by one molecule in a 1 M solution. For example, we add two of these entropic penalty terms to the NUPACK predicted energies for the complexes in states **A** (Reporter and Complex) and **F** (m-product and n-product) in Figure 5.4 to account for the cost of bringing two strands together, twice. States **B** and **E** show two separate molecules that are co-localized, and thus pay an entropic penalty despite not being bound together in one molecule. We add one more energetic term (for a total of three) to the NUPACK predicted energies for the complexes in states **B**, **C**, **D** and **E** to account for the entropic cost of bringing the Reporter and the Complex together. These extra terms cancel in the overall reaction shown in Figure 5.1A (the energy difference between states **A** and **F** in Figure 5.4).

We now explore a path from **A** to **F** using the full secondary structure state space. While there are many possible trajectories that can take us from **A** to **F** in this state space, we will consider a particular "minimum energy barrier pathway" which can be used to define a reaction coordinate [Moulton et al., 2000]. The pathway from **A** to **F** is as follows: the **m** toehold binds first, followed by the **n** toehold (if present), and branch migration proceeds forward by the two bond-breaking steps followed by two bond-formation steps that comprise a single step of the branch migration (Figure 5.4B and D). An additional trajectory is provided in Figure C.11.

These elementary step energy landscapes provide a few key insights into the mechanism. In contrast to the diagrammatic energy landscapes, the actual trajectories are nuanced: the two main energy barriers in the open case are co-localization (state **B**) and closing the multi-loop (state **D**), but in the elementary step landscape we also observe the extended branch migration "plateau"; any effect this plateau may have on kinetics can only be examined in the more detailed model (Figure 5.4A and B). The major energy barrier in the closed case is co-localization (state **B**), while the barrier to the first step of branch migration (state **D**) is now comparable with those of other branch migration steps (Figure 5.4C and D). During toehold binding (the states between **B** and **C**), we can clearly distinguish the first toehold binding, the cost of closing the multiloop, and the second toehold binding. The analysis below shows that the dominant factor controlling overall reaction kinetics is the probability that, from the toehold bound state **C**, the molecules fall apart without reacting (first return to state **B**) or initiate and complete branch migration (first reach state **E**, going through **D**).

5.6 Mechanistic Model

The phenomenological model that guided our experiments in Section 3 (the two-step reaction mechanism shown in Figure 5.1A) was only concerned with the two reactants, one intermediate, and the two products of the toehold-mediated four-way branch migration reaction (Figure 5.5A). While this model guided our experiments, we cannot use it to explain how our measured rate constants depend upon m and n. In contrast, the general-purpose elementary-step model using a nearest-neighbor secondary structure energy landscape provides a unified means to predict rate constants for all (m, n) pairs – but does not provide simple analytic understanding. Our goal in this section is to capture the accuracy of the Multistrand simulations with (nearly) the simplicity of the coarse-grained approach that guided our experiments.

We derive a mechanistic model that explains all of the data with only four new parameters (k_2^{open}, k_2^{closed}, $\Delta G^{fit}_{k_2^{open}}$ and $\Delta G^{fit}_{k_2^{closed}}$), a hybridization rate constant (k_f) and the established NUPACK parameters. One difference between the mechanistic and phenomenological models is the former's consideration of the barrier to forming a stable Holliday junction from which the uni-

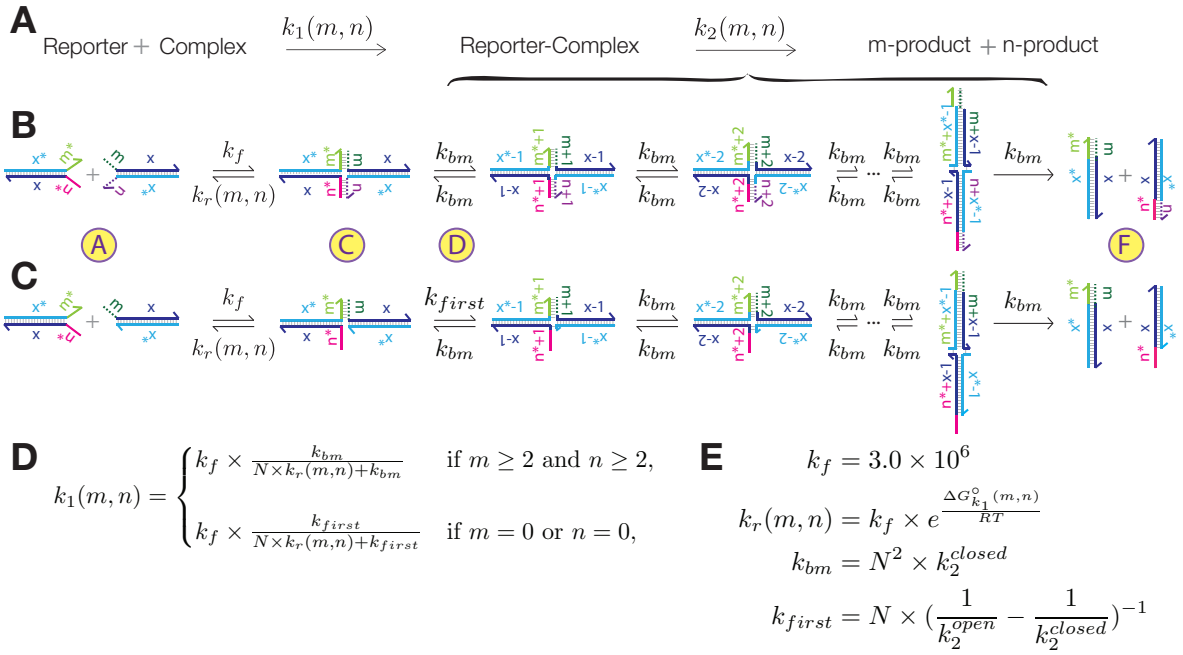


Figure 5.5: A summary of our phenomenological and mechanistic models. (A) The phenomenological reaction with rates $k_1(m,n)$ and $k_2(m,n)$. (B) A mechanism for the closed reaction ($m \geq 2$ and $n \geq 2$) shows branch migration steps, each corresponding to four individual base pair breaking and formation steps in Figure 4. (C) A mechanism for the open reaction ($m = 0$ or $n = 0$) also shows branch migration steps, but now distinguishes the first step, where initiating branch migration requires closing the multiloop. Yellow circles label states that correspond to those in Figure 5.4, State D corresponds to an intermediate transition state. (D) Formulas for calculating $k_1(m,n)$ as derived in Section 5. (E) Formulas for k_f, $k_r(m,n)$, k_{bm}, and k_{first} as derived in Section 5.

molecular branch migration can proceed. As a result, the mechanistic model uses a small parameter set to fit all of the data, whereas the phenomenological model requires 32 independent parameters to separately fit 16 toehold pairs.

The mechanistic model explicitly considers individual branch migration steps, but with less detail than the NUPACK one-dimensional landscape. The model is simple: we ignore sequences and only consider the key free energies associated with the unimolecular and bimolecular processes of the reaction. We did not test how differing sequences might affect the kinetics of the reaction, rather we assume that different sequences will behave similarly (however, sequence effects are taken into account by NUPACK in the calculation of complex free energies). In addition, we chose to model the unimolecular branch migration process as an unbiased random walk (ignoring branch

sequence dependence as well).

Figure 5.5 summarizes the mechanistic model. In Figure 5.5B and 5.5C we distinguish between the two cases under which four-way migration may proceed: the "closed" case in which both toeholds are present on both reactants and the "Reporter-Complex" intermediate forms a Holliday junction, and the "open" case in which only one toehold is able to form in the Reporter-Complex intermediate resulting in a branched DNA complex with no initial Holliday junction. What in the phenomenological model was considered simply the bimolecular rate is in the mechanistic model broken down into a forward and a reverse rate. The forward rate, k_f, is a fixed constant independent of toehold length in both the closed and open cases of the reaction. The reverse rate, $k_r(m,n)$, of intermediate formation is the rate at which the intermediate disassociates into the original complexes before branch migration completes, and is dependent on the length and sequence (or "strength") of toeholds **m** and **n**. This allows us to introduce the assumption that non-successful interactions are due to the release of toeholds before branch migration is successful.

Given the detailed balance condition, $k_r(m,n)$ can be expressed in terms of k_f and $\Delta G^\circ_{k_1}$, the free energy of intermediate formation (i.e. toehold binding), as follows:

$$k_r(m,n) = k_f \times e^{\Delta G^\circ_{k_1}/RT}, \tag{5.1}$$

where $\Delta G^\circ_{k_1} = \Delta G^\circ_{k_1^{open}}$ or $\Delta G^\circ_{k_1^{closed}}$ as appropriate. The overall unimolecular rate, either k_2^{closed} or k_2^{open}, describes the transition from the intermediate structure to the formation of the two products. This rate includes the time to initiate and complete branch migration (we assume that initiation time is negligible in the closed case), and thus will depend on the rates of the individual unimolecular steps, k_{bm} and k_{first}.

Below, we show how both the closed and open cases of the mechanistic model can be derived from an elementary step consideration of the branch migration mechanism. A branch migration can be thought of as a random walk process in one dimension: at any given position in the branch migration process, the next step can be a bond breaking and bond formation step that moves the Holliday junction either one step to the left or one step to the right (Figure 5.5B and Figure 5.5C). Since branch migration is a random walk along an N-step path, the probability,

$P(m,n)$, of successfully completing branch migration is dependent on the number of steps N (in this paper $N = 21$). We ignore the sequence dependent differences observed in the NUPACK energy landscape for simplicity, and we assume that once the Reporter-Complex intermediate is formed, the molecule has already begun the process of branch migration, with an equal chance of moving the Holliday junction to the left (toward disassociation of the reactants) or to the right (toward completion of the branch migration in the direction of the products).

The value of $k_1(m,n)$ can be defined as $k_f \times P(m,n)$, where $P(m,n)$ is the probability that the Reporter and Complex will complete branch migration once the two complexes are joined in the intermediate state. This probability is derived differently in the closed and open cases.

$$k_1(m,n) = k_f \times P(m,n). \tag{5.2}$$

5.6.1 Closed Model

In order to model $k_1(m,n)$ for $m \geq 2$ and $n \geq 2$ we define it in terms of k_2^{closed}, the unimolecular rate that we extracted from the long-toehold experiments, k_f, the hybridization rate of single stranded DNA, and $k_r(m,n)$, the reverse rate of intermediate formation. In the closed case, we assume that all of the elementary steps along the branch migration pathway are equivalent, including the first step, hence the rate of each individual step is k_{bm}.

The probability of initiating four-way branch migration (or moving one step to the right or toward the products) from the Reporter-Complex intermediate state is the probability of going to the right divided by the sum of the probabilities of moving left or right. Since the rate of branch migration is k_{bm} and the rate of disassociation is $k_r(m,n)$, the probability of making the first step toward the products in this random walk is $\frac{k_{bm}}{k_r(m,n)+k_{bm}}$. If the branch migration begins, it will successfully complete with probability $\frac{1}{N}$, before the complexes disassociate [Feller, 1968].

Since we assume that the probability of branch migrating to the left or to the right is 0.5, it follows that the probability of successfully completing branch migration from the intermediate starting point before the two complexes disassociate is $\frac{1}{N}$, and the probability of returning to the initial starting point is $\frac{N-1}{N}$. If the branch migration is started, and returns to the intermediate

state before it completes, then it will once again have probability $P(m,n)$ of successfully completing branch migration. Thus, $P(m,n)$ is the probability of beginning branch migration from the intermediate step multiplied by the probability of reaching the end before the two complexes disassociate:

$$P(m,n) = \frac{k_{bm}}{k_r(m,n)+k_{bm}} \times \left(\frac{1}{N} + \frac{N-1}{N} \times P(m,n)\right) \tag{5.3}$$

This reduces to:

$$P(m,n) = \frac{k_{bm}}{N \times k_r(m,n)+k_{bm}}. \tag{5.4}$$

By substituting equation (5.4) for the value of $P(m,n)$ in equation (5.2) we get:

$$k_1^{calc} = k_f \times \frac{k_{bm}}{N \times k_r(m,n)+k_{bm}}. \tag{5.5}$$

All that remains to be shown is how we arrive at a value for k_{bm}. k_2^{closed} is the overall rate of the unimolecular branch migration reaction, or the time it takes to complete branch migration once it is initiated. From the probability theory of one dimensional random walks, we know that the expected time to reach a position x from the origin is x^2 [Feller, 1968]. Thus, we can derive the step rate, k_{bm}, from k_2^{closed} as follows: the expected time for a single branch migration step is $\frac{1}{k_{bm}}$, the expected time to complete the whole branch migration is $\frac{1}{k_2^{closed}}$, and therefore

$$\frac{1}{k_2^{closed}} = \frac{N^2}{k_{bm}}, \tag{5.6}$$

so that k_{bm} is equal to $k_2^{closed} \times N^2$. When we substitute this value for k_{bm} into equation (5.5) we arrive at:

$$k_1^{calc} = k_f \times \frac{k_2^{closed}}{\frac{k_r(m,n)}{N} + k_2^{closed}}. \tag{5.7}$$

5.6.2 Open Model

The open model has an additional consideration: the barrier to forming a Holliday junction from the Reporter-Complex intermediate. In this case we now have a large uphill first step to initiate branch migration because the complexes must overcome the entropic cost of forming a loop (as seen in Figure 5.4).

If branch migration is fast with respect to the rate of the first step, k_{first} (which consists of closing the loop and initiating branch migration), then k_2^{open} would be dominated by the rate of this first step, and

$$k_2^{open} \approx \frac{1}{N} \times k_{first}. \tag{5.8}$$

As above, $\frac{1}{N}$, is the probability that once branch migration is initiated it will complete before the complex returns to the initial starting point (the Reporter-Complex intermediate). We can use the value of k_2^{closed} to calculate the expected time of the first step:

$$\mathbb{E}[Time_{C\to E}] = N \times \mathbb{E}[Time_{C\to D}] + \mathbb{E}[Time_{D\to E}] \tag{5.9}$$

where $\mathbb{E}[Time_{C\to E}]$ is the expected time to complete the unimolecular reaction (moving from state C to state E in Figure 5.4), $\mathbb{E}[Time_{C\to D}]$ is the expected time to complete the first step (moving from state C to state D in Figure 5.4) and $\mathbb{E}[Time_{D\to E}]$ is the expected time to complete branch migration after the first step (moving from state D to state E in Figure 5.4). N is the number of attempts required to initiate branch migration before successful completion. $\mathbb{E}[Time_{D\to E}]$ is equivalent to the expected time to complete the unimolecular reaction in the closed case, which is $\frac{1}{k_2^{closed}} = \frac{N^2}{k_{bm}}$. Thus,

$$\frac{1}{k_2^{open}} = \frac{N}{k_{first}} + \frac{N^2}{k_{bm}}, \tag{5.10}$$

which yields:

$$k_2^{open} = \frac{1}{\frac{N}{k_{first}} + \frac{1}{k_2^{closed}}}. \quad (5.11)$$

Now we can solve for k_{first} from measurable quantities.:

$$k_{first} = \frac{N}{\frac{1}{k_2^{open}} - \frac{1}{k_2^{closed}}}. \quad (5.12)$$

The probability of successfully closing the loop and initiating branch migration is

$$P(m,n) = \frac{k_{first}}{N \times k_r(m,n) + k_{first}}. \quad (5.13)$$

Thus,

$$k_1^{calc} = k_f \times \frac{k_{first}}{N \times k_r(m,n) + k_{first}}. \quad (5.14)$$

5.6.3 Results

The mechanistic model presented here is built on parameters from the literature (k_f and ΔG°), parameters that were directly measured (k_2^{open} and k_2^{closed}), and two parameters that were fit to our data to adjust predicted ΔG° values (described below). A summary of all parameters and equations used in the model can be found in Figure 5.5. We now return to the energy landscape discussion that informed the mechanistic model in order to explain additional fit parameters and to compare the model directly to both the experiments and simulations.

The diagrammatic energy landscapes (Figure 5.4A and C) show the two main energy barriers to the completion of a toehold-mediated four-way branch migration reaction: $\Delta G^\circ_{k_1}$ corresponds to the standard free energy of the bimolecular reaction in which the Reporter and Complex bind by their toeholds to form the Reporter-Complex intermediate, and $\Delta G^\circ_{k_2}$ corresponds to the standard free energy of the unimolecular reaction in which the Reporter-Complex intermediate completes the branch migration to yield the m-product and n-product. Both are negative by convention, and

overall,

$$\Delta G^\circ = \Delta G^\circ_{k_1} + \Delta G^\circ_{k_2}. \tag{5.15}$$

While we have confidence in the NUPACK-predicted free energy ΔG°, we found that the NUPACK-predicted values for $\Delta G^\circ_{k_1}$ did not result in good fits of the mechanistic model equations (Figure 5.5D and E) to the experimental data. (NUPACK's values predict on average a 408-fold slow-down of closed reactions relative to experimental results, and a speed up of open reactions by as much as a factor of 65.8.) This could be due to inaccuracies in the predicted energy of the intermediate state **C** (e.g. because of unmodeled coaxial stacking effects), due to oversimplifications of the mechanistic model relative to the NUPACK energy landscape, due to inaccuracies in the experimentally measured k_2 values, or mostly likely a combination of the above. Therefore, we chose to incorporate an empirical adjustment to $\Delta G^\circ_{k_1}$ for use within the mechanistic model. To do so we can use either $\Delta G^\circ_{k_1}$ or $\Delta G^\circ_{k_2}$ to adjust NUPACK's free energy predictions, since estimating one of these values gives us the other and we treat NUPACK's prediction of ΔG° as reliable. While $\Delta G^\circ_{k_1}$ depends on each combination of m and n, NUPACK predicts $\Delta G^\circ_{k_2}$ to be independent of toeholds m and n beyond whether they belong to the open or closed case. Thus we chose to calculate $\Delta G^\circ_{k_1}$ using NUPACK's prediction for ΔG° and two fit values for $\Delta G^\circ_{k_2}$, one for the open case and one for the closed case. $\Delta G^{fit}_{k_2^{open}}$ and $\Delta G^{fit}_{k_2^{closed}}$ are each a least squares fit of experimental data points in Figure 5.6B to values of k_1 calculated as in Figure 5.5, where all other parameters are fixed. We arrived at the following values: $\Delta G^{fit}_{k_2^{open}}$ is -2.41 kcal mol^{-1} and $\Delta G^{fit}_{k_2^{closed}}$ is -5.39 kcal mol^{-1}. For reference, if we had calculated these values from NUPACK, we would have arrived at $\Delta G^{NUPACK}_{k_2^{open}} = -1.36$ kcal mol^{-1} and $\Delta G^{NUPACK}_{k_2^{closed}} = -8.95$ kcal mol^{-1}.

In total the mechanistic model has four fit parameters (k_2^{open}, k_2^{closed}, $\Delta G^{fit}_{k_2^{open}}$ and $\Delta G^{fit}_{k_2^{closed}}$) in addition to k_f and ΔG° values from the literature. Figure 5.6 compares the mechanistic models for open and closed-loop toehold-mediated four-way branch migration to the experimentally fit mean k_1 rates and to rates predicted from the elementary step simulations. We now see excellent agreement. An exception is the three slowest open-loop cases, where the mechanistic model predicts lower rates than are observed. It is unclear whether this discrepancy reflects an inaccuracy of

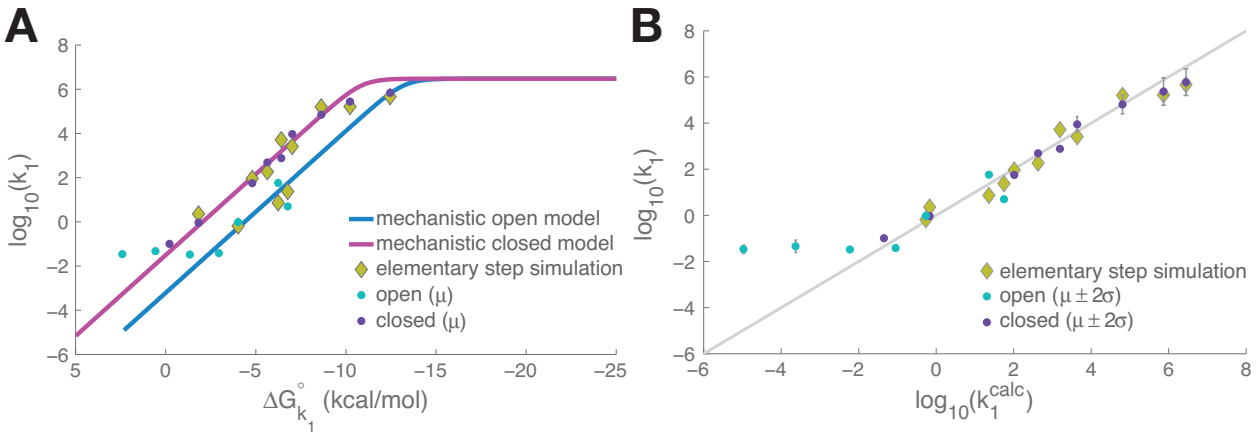


Figure 5.6: (A) Plot of adjusted $\Delta G^{\circ}_{k_1}$ versus $\log_{10}(k_1)$ comparing the mechanistic models for open (blue) and closed (magenta) loop toehold-mediated four-way branch migration to the experimentally fit mean k_1 rates. Dots correspond to open (blue) and closed (purple) loop reactions, and an elementary step simulation using predicted NUPACK free energies (yellow diamonds). Note that neither the $\Delta G^{fit}_{k_2}$ parameter nor the k_2^{fit} parameter was used in the elementary step model, but the x-axis $\Delta G^{\circ}_{k_1}$ is plotted with the adjusted value. (B) Plot comparing the $\log_{10}(k_1)$ rate calculated by the mechanistic model to the experimentally fit mean rates. Error bars show two standard deviations of error in experimental measurements. The gray line indicates the points at which $x = y$.

the mechanistic model (e.g. for small or positive values of $\Delta G^{\circ}_{k_1}$ where the nature of the energy landscape changes, c.f. Figure S11), or whether it simply reflects a limitation in our experimental technique's ability to accurately measure very slow reactions (e.g. below the rate of 0.01 M^{-1} sec^{-1}). Our ability to accurately measure the kinetics of the fastest reactions is also limited, as reflected by greater variance in the data. The measured values are not unreasonable; even the fastest reaction rates are lower than the k_f value from prior work that we use in the mechanistic model.

The success of both the mechanistic model (using NUPACK-predicted energies and empirical adjustments) and the elementary step simulations (using the full NUPACK energy landscape without adjustments) suggests, first, that the NUPACK energy landscape is not too far off (i.e. inaccuracies due to coaxial stacking and divalent salt interactions at the junction are not fatal), and therefore, second, that the empirical adjustments k_2^{fit} and $\Delta G^{fit}_{k_2}$ serve rather to accommodate coarse-graining effects found in the mechanistic model.

5.7 Discussion and Conclusions

In our model we experimentally derived unimolecular reaction rates using complexes with long-toeholds. We assume that k_2^{closed} is the rate for an overall N-step branch migration to complete with no initiation barrier. We found that the time to half completion for the closed case is 7.7 minutes. We derive the step time by dividing by N^2 where N is the length of the double stranded region that is exchanged during four-way branch migration. This yields a step time of 1.05 seconds ($\approx \frac{1}{k_{bm}}$). This is comparable with other experimentally measured rates for four-way branch migration stepping times. McKinney et al. found that the time between steps in DNA branch migration is sequence dependent and ranges between 1 and 2 seconds per step (experiments were conducted at 25 °C with comparable salt concentrations to those that we used) [McKinney et al., 2005]. Panyutin and Hsieh measured the step time of branch migration with and without 10 mM Mg^{++}, but at a higher temperatures than that used here. We estimate a step time of approximately 2.7 seconds at 25 °C by assuming that the exponential slowing of branch migration kinetics that they observed from 50 °C to 37 °C continues to hold at lower temperatures [Panyutin and Hsieh, 1993]. Our experimental data is reasonably consistent with prior studies of four-way branch migration stepping time.

We also examined how our experimentally-fit unimolecular rates match those that would be predicted by the energy landscape in Figure 5.4. Consider the energy barrier between State C and the branch migration plateau in our energy landscape, which is State D. The height of this barrier, ΔG_{D-C}, should determine k_{bm} and k_{first}, much as the energy barrier to toehold dissociation, $-\Delta G^{\circ}_{k_1}$ determines $k_r = k_f \times e^{\Delta G^{\circ}_{k_1}/RT}$ in Equation 5.1. We should be able to predict k_{bm} and k_{first} in a fashion similar to the way we define k_r (i.e. $k_r = k_f \times e^{\Delta G^{\circ}_{k_1}/RT}$ from Equation 5.1). This suggests that k_{bm} should be proportional to $e^{-\Delta G^{closed}_{D-C}/RT}$ and that k_{first} should be proportional to $e^{-\Delta G^{open}_{D-C}/RT}$ with the same kinetic pre-factor that reflects the base rate for unimolecular transitions in our model. Thus we should be able to extract the ratio

$$\frac{k_{bm}}{k_{first}} = \frac{e^{-\Delta G^{closed}_{D-C}/RT}}{e^{-\Delta G^{open}_{D-C}/RT}} \tag{5.16}$$

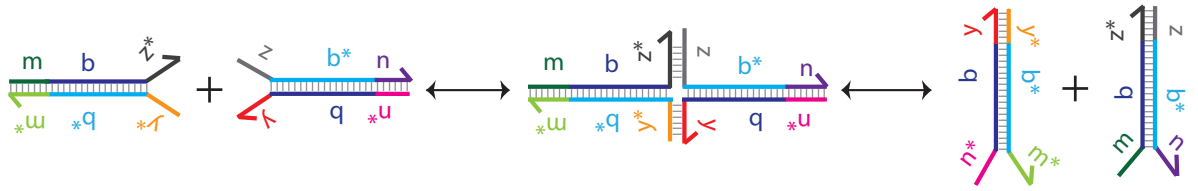


Figure 5.7: A toehold-mediated four-way strand exchange mechanism. Note that the mechanism is identical to our toehold-mediated four-way branch migration scheme except for the addition of two more domains on each of the complexes. The reaction is reversible.

from the energy landscape. In the NUPACK energy landscape, $\Delta G^{closed}_{D-C} = 3.01$ and $\Delta G^{open}_{D-C} = 11.14$. This gives us a ratio of $\approx 760,000$. When we compare this value to the ratio of ≈ 73 that we get from our experimental data where $k_{bm} = 0.95$ sec^{-1} and $k_{first} = 0.013$ sec^{-1}, it is off by four orders of magnitude. However if we correct the energies of State C using our $\Delta G^{fit}_{k_2}$ values, we get $\Delta G^{closed}_{D-C} = 6.57$ and $\Delta G^{open}_{D-C} = 10.09$ and a predicted k_{bm}/k_{first} ratio of 353, which is within one order of magnitude of the ratio derived from our experimental results. These results fit reasonably well and justify our intuition.

We investigated a limited set of toehold lengths due to cost and time constraints; however the Multistrand simulator can be used to investigate the entire range of toehold lengths at the cost of only additional simulation time. There are many remaining unknowns in controlling the kinetics of four-way branch migration reactions. For example, we have not explored how our model will generalize to other sequences in either the toehold or branch regions. We classified the "open" and "closed" cases of four-way branch migration, but we don't know how a one base toehold would affect the unimolecular step (e.g. when $m = 16$ and $n = 1$), nor how mismatches in the toehold sequence would behave. In principle, Multistrand can be used to simulate all of these conditions and make predictions about kinetics that can guide experimental design. Multistrand can be used to identify sequence designs that have unintended features such as alternate reaction pathways or kinetic traps, which may be difficult to diagnose in an experimental setting. Another feature of using a simulator in this context lies in its ability to make predictions about the kinetics of reactions that are beyond our experimental resolution, such as the very slow short toehold reactions that were discussed in this paper (see Figure 5.6).

While this characterization of toehold-mediated four-way branch migration explores irre-

versible processes, we recognize that more interesting dynamic behavior may be generated by reversible toehold-mediated four-way branch migration reactions. An analog to this system is the reversible toehold-mediated three-way branch migration system that was studied by [Zhang and Winfree, 2009]. Reversible three-way systems have been used to construct complex chemical logic circuits [Zhang et al., 2007, Qian and Winfree, 2011, Qian et al., 2011] and appear essential to more complex theoretical constructions [Soloveichik et al., 2010,Cardelli, 2011]. To our knowledge, four-way strand exchange systems have not been constructed yet, but may prove useful. This manuscript presents a first-step toward understanding reversible four-way branch migration reactions as a reversible system demonstrates the same kinetics as an irreversible system when one toehold is stronger than the other. Just as we can derive toehold-mediated three-way strand exchange kinetics from toehold-mediated three-way strand displacement reactions, we expect that we can do the same using toehold-mediated four-way branch migration reactions.

Our work presents a model for controlling the rate of a four-way branch migration using toeholds, based on the length and strength of these toeholds. We have shown that we can control the rate of a four-way branch migration reaction over at least seven orders of magnitude, and that these rates are exponential in the free energy of the toehold binding step. We experimentally characterized how the length and strength of toeholds affects the kinetics of the bimolecular rate. We have used both thermodynamic (NUPACK) and kinetic (Multistrand) modeling to elucidate this mechanism and have derived a mechanistic model with four fit parameters and a rate constant that is consistent with an elementary step model simulating the full nearest neighbors secondary structure energy landscape. We hope the model presented here will be used by DNA nanotechnologists to design more complicated and robust dynamical systems.

5.8 Materials and Methods

Experimental System A typical fluorescence kinetics experiment contains a Complex with toeholds (m, n), and a Reporter (labeled with a fluorophore and quencher pair on opposite strands) mixed together in solution (Figure 5.1A), where the Complex is added last to trigger the reaction. As the Complex and and Reporter exchange strands, the fluorophore and quencher pair on the Re-

porter are separated from each other, yielding an increased fluorescence signal in the solution. At the end of the experiment another strand of DNA, typically n-displace, was added into the solution in order to fully displace all unreacted quencher strands on the Reporter (Figure 5.1B). This "displacement" strand was added in at least $50\times$ excess to the concentration of Reporter in the solution to ensure that this reaction quickly goes to completion. We use the final completion level to normalize our fluorescence signals.

DNA Sequences and Design The sequences presented here (in Figure 5.1 and in Tables C.1 and C.6) are based on those used in a previous insertional polymerization motor [Venkataraman et al., 2007]. These sequences were designed using the NUPACK web application [Zadeh et al., 2010,Zadeh et al., 2011] and our in-house DNA Design software package to minimize the presence of any unanticipated secondary structures that might interfere with the kinetics under investigation.

Toehold Binding Energy Calculations The free energy ΔG° of all complexes was found using the NUPACK web application [Dirks et al., 2007, Zadeh et al., 2010]. The structure and sequence information was entered into the utility function on Nupack.org with ion concentrations set to 0.05 M Na+ and 0.0125 M Mg++, and dangles were set to "some". The free energy of a reaction was determined by subtracting the partition function-based free energies of the reactants from the free energies of the products. In calculating intermediate structures a mole fraction correction was used [Dirks et al., 2007]. The $\Delta G^\circ_{k_1}$ values presented in Table 5.2 are the NUPACK values corrected by a $\Delta G^{fit}_{k_2}$ parameter to correct for possible coaxial stacking effects unaccounted for in current thermodynamic models of DNA.

Buffer Conditions DNA oligonucleotides were stored in TAE 12.5 mM Mg++ buffer (purchased as $50\times$ stock TAE from Invitrogen, and solid Magnesium Acetate from Sigma) at 4°C directly preceding experiments. All experiments and purifications were performed at 25°C, with temperature controlled using an external temperature bath.

Annealing All annealing processes were performed with an Eppendorf Mastercycler Gradient thermocycler. The samples were brought down from 95°C to 16°C at a constant rate over the course of 90 min.

Substrate Purification DNA oligonucleotides used in this study were purchased from Integrated DNA Technologies (IDT), with standard desalting purification, except for strands with a

quencher, fluorophore or a $5'$ toehold involved in the four-way branch migration. These strands were ordered with HPLC purification. Concentrations of individual strand stocks were determined from the measured absorbance at 260 nM using an Eppendorf Biophotometer and the calculated extinction coefficients provided by IDT.

Reaction complexes were further purified by nondenaturing (ND) polyacrylamide gel electrophoresis (PAGE) as follows: Strands for each sample were prepared with nominally correct stoichiometry at 10 nM and annealed. The acrylamide (19:1 acrylamide:bis) was diluted from 40% acrylamide stock (Ambion). ND loading dye (containing Bromphenol Blue in 50% glycerol) was added to all samples, achieving a final gycerol concentration of 10% by volume. The samples were then run on 12% ND PAGE at 120 V for 6 h.

Gels were run at room temperature ($\approx 25°$C) using a Hoefer Vertical Slab Gel unit. The proper bands were cut out and eluted in 1 mL of 12.5 mM TAE Mg++ buffer for two days. Purified complexes were quantitated by measurement of absorbance at 260 nm using an Eppendorf Biophotometer and extinction coefficients calculated by summing the IDT provided extinction coefficients for individual strands.

Spectrofluorimetry Studies Spectrofluorimetry studies were done using a SPEX Fluorolog-3 (Horiba) with external water bath and 1.6 mL synthetic quartz cells (Hellma 119-004F). The excitation was at 584 nm, while emission was at 604 nm. In all spectrofluorimetry experiments, the total reaction volume was 1.5 mL and the temperature was $25°$C. For net reaction studies in which the concentration of the reporter was in excess of 1 nM, 2 nm band-pass slits were used for both excitation and emission monochrometers; for experiments in which the reporter concentration was less than 1 nM, 4 nm slits were used. Experiments were conducted with an integration time of 10 s for every 60 s time-point. The fifth data set in the long toehold experiments was conducted with an integration time of 10 s for every 15 s time-point. Prior to each experiment, all cuvettes were cleaned as follows: each cuvette was washed 15 times in Milli-Q water, 5 times in 70% ethanol, another 15 times in Milli-Q water, and finally once more in 70% ethanol and then Milli-Q water. For the slit size, concentrations, and times chosen, no measurable photobleaching was observed.

Carrier Strands It is well-known that DNA sticks nonspecifically to pipet tips. Since this loss

is inconsistent, we introduced 20-nucleotide-long poly-T "carrier" strands into our experiments to coat our pipet tips. We used the carrier strands only in experiments in which one or more of our complexes occurred at a final concentration of less than 10 nM. In these experiments, the pipette tip used to add a complex kept at a stock of less than 1 μM was first dipped into a stock of carrier strand at 100 μM concentration; this stock was pipetted up and down into the tip 15 times, before being released. After this the same pipette tip was used to add the low concentration complex to the cuvette. Poly-T strands have minimal influence on the reactions of other DNA molecules in this system [Zhang et al., 2007, Zhang and Winfree, 2009].

Fluorescence Normalization Fluorescence is normalized so that one normalized unit of fluorescence corresponds to 1 nM of unquenched fluorophore-labeled strand Reporter-2. This normalization is based on the fluorescence levels of annealed samples with a minimal fluorescence measurement taken of the diluted Reporter complex before the experiment was initiated, and a maximal fluorescence value taken at the end of the experiment, after the m-displace (or n-displace) strand is added to displace all unreacted fluorophore-quencher pairs.

Parameter Fitting At least three traces for each toehold combination were analyzed and each toehold combination was investigated with at least two different concentration sets (see Appendix C for concentrations used, and all traces). The best-fit rate constants to experimental data were fitted using the "fminunc" or the "polyfit" function in Matlab to minimize the mean squared error between experimental data and our model (see sample code in Supplementary Information). All traces that did not reach a completion level of 20% within 24 hours were fit linearly using the polyfit function in Matlab. Each experimental trace was fit separately.

In order to determine k_2 we designed reaction complexes with a 16-basepair toehold. We assumed $k_1 = k_f = 3 \times 10^6$, a value taken from prior work studying toehold-mediated strand displacement kinetics [Zhang and Winfree, 2009]. After fitting the k_2^{open} and k_2^{closed} values using these "long-toehold" experiments, we used these values in fitting the k_1 rates on the smaller toehold length experiments.

We noted a lack of completion in our faster reactions with very low concentrations. In the $(m = 4, n = 6)$, $(m = 6, n = 2)$, $(m = 6, n = 4)$, and $(m = 6, n = 6)$ experiments, we adjusted the simulation to set the maximal concentration to the maximal fluorescence signal before

the displacement strand was added.

Finally, after fitting the reaction rates, we fit the data to our model by adding a $\Delta G_{k_2}^{fit}$ parameter to the ΔG° values collected from NUPACK. We found this parameter by using a minimum least squares error fit between the data and our model (see explanation in Section 5.6.3 and sample code in Appendix C).

As each trace was fit individually, the $k_1^{fit}(m,n)$ values and standard deviations reported in Table 2 is the mean value and standard deviation across all experiments for each (m,n) pair. Error bars used in Figure 5.6 show two standard deviations above and below the rate constant fitted using all three (or more) data traces.

Trajectory Simulations Simulated reaction rates (k_1^{sim}) were calculated using Multistrand [Schaeffer, 2012], an analysis tool that simulates the kinetics of multistranded DNA systems with single-basepair resolution utilizing the NUPACK energetics model. Sample size indicates the number of trajectories simulated. Final values were normalized by computing a scaling factor by minimizing the mean multiplicative factor that best fit the raw multistrand results to the experimental results, which in this case was a factor of 20. Values that are upper-bounded indicate reactions that did not have a single forward result.

Chapter 6

Conclusion and Future Directions

This thesis has demonstrated the implementation of molecular machines while formulating an overarching method to program very simple molecules.

We have identified a new class of programmable behaviors that require an energy source external to the system and cannot be implemented on "energetically-incomplete" systems even if the systems are Turing-complete. As we have demonstrated, a sufficiently expressive implementation of an "active" molecular self-assembly approach can achieve these behaviors. Using an external source of fuel solves part of the the problem, so the system is not "energetically incomplete." But the programmable system needs to have sufficient expressive power to achieve the specified behaviors. Perhaps surprisingly, some of these systems do not even require Turing completeness to be sufficiently expressive, as we proved at the end of Chapter 3.

We have constructed a new computational model for active self-assembly (Chapter 3), the first of its kind to be implemented in DNA molecules (Chapter 4). We have designed and experimentally verified the construction of the first synthetic linear polymer capable of growing exponentially fast and also capable of dividing. Finally, we have characterized the kinetics of the insertional mechanism used to implement our model (Chapter 5).

When visualizing the capabilities of our system, the reader might wonder whether the transformation of information into an active entity is inherently bottlenecked by some linear phase. In biology, all information is stored in the genome whether it be chromosomal or mitochondrial. This information must be transcribed into RNA which is then translated into strings of proteins that fold into their desired shape. We have shown that exponential growth of a polymer is possible, but we

have yet to construct a two or three dimensional model or implementation of such a behavior. This thesis also presented a proposed implementation for treadmilling in a linear system by combining an insertion primitive with a division/deletion primitive, but this behavior runs in linear time.

Another open question for future work would be the analysis of the complexity of a system capable of metamorphosis: what is the trade-off between complexity of information inherent in the system and the flexibility of its component molecules?

In the context of nanotechnology and material science this work presents an advance in our ability to manipulate matter. This work is part of a growing push in these fields toward fabricating smart materials that can be programmed and that interact via molecular reactions, thus rendering them capable of being interfaced with biological compounds. Such materials capable of complex behaviors and programs will not run the kinds of programs that computers run (nor should they). Rather these molecules will most likely be used to do what molecules do best: communicate information via shape, structure and interaction. They will be used by future doctors to deploy an army of molecular surgeons like those that Richard Feynman envisioned [1] [Feynman, 1960]. They may be used to reprogram the nano wires in reusable hardware.

These future molecules and the materials they comprise will likely be used to design materials capable of modifying themselves, reusing materials, and adapting or metamorphosing as needs change. In short, they will be used to do what nature and biology and biochemistry do best, but perhaps at the control of a human re-imagining his or her reality. This ultimate goal is one that will be reached at the intersection of several fields that converge on materials, information, biochemistry and physics.

While we have motivated this work by asking how it is that information becomes an active entity, in the process this work touches or creates a small piece of what Jean Baudrillard calls "the hyperreal" [2] [Baudrillard, 1994]. In the larger context of philosophical discourse this is significant

[1] In his essay '*There's Plenty of Room at the Bottom*, Feynman writes: "(Albert R. Hibbs) suggests a very interesting possibility for relatively small machines. He says that, although it is a very wild idea, it would be interesting in surgery if you could swallow the surgeon. You put the mechanical surgeon inside the blood vessel and it goes into the heart and 'looks' around...It finds out which valve is the faulty one and takes a little knife and slices it out. Other small machines might be permanently incorporated in the body to assist some inadequately-functioning organ."

[2] Baudrillard explains this concept as follows: "Today abstraction is no longer that of the map, the double, the mirror, or the concept. Simulation is no longer that of a territory, a referential, or a substance. It is the generation by models of a real without origin or reality: a hyperreal. The territory no longer precedes the map ..."

as human society is embarking on an era in which we are no longer building out of the raw materials of the earth, but decomposing those materials into molecules and remaking nature as we see it. While this future technology may enable myriad possibilities, society at large must continue to confront how quickly our bodies, our psychologies, and our cultures will adapt to materials that are evolving thousands of times faster than we are [Toffler, 1984].

Appendix A

Partial Supplementary Material for Molecular Robots Guided by Prescriptive Landscapes[0]

A.1 Supporting Discussion

Robots are often defined by their ability to sense their environment, perform computations, and take actions; as such, they have revolutionized our ability to automate factories, send autonomous vehicles to remote or dangerous locations, and improve our daily lives. The potential for autonomous sensing and acting at the molecular scale is illustrated by the sophisticated machinery within biological cells, where molecular motors and biochemical circuitry coordinate the cell's active responses to its environment. From a chemist's perspective, the potential for molecular robotics goes far beyond what is observed in biology, but the challenges of realizing that potential are daunting, due to the need to synthesize behavior.

[0]This work was coauthored by Kyle Lund, Anthony J. Manzo, Nadine Dabby, Nicole Michelotti, Alexander Johnson-Buck, Jeanette Nangreave, Steven Taylor, Renjun Pei, Milan N. Stojanovic*, Nils G. Walter*, Erik Winfree*, & Hao Yan* and published in 2010 [Lund et al., 2010] with the following contributions: AFM experiments were performed by K.L. (majority), J.N., and N. D.; analysis was performed by N. D., K.L., J.N., S.T., and supervised by E.W., and H.Y.; fluorescence microscopy and particle tracking analysis were performed by A.J.M., N.M., A.J.B, and supervised by N. G. W.; spiders were synthesized, purified, and their integrity confirmed and monitored by S.T.; SPR experiments were performed by R. P.; research coordination by M.N.S., material transfer coordination by S.T., J.N., and K.L. Experimental design and manuscript was done with input from all authors.

As with protein motors, an isolated molecular robot by itself serves no purpose; to be useful, it must interact with its environment of other molecules and molecular machines; it must behave. Despite vast differences in size, classical robotics [Braitenberg, 1984, Brooks, 1991, Simon, 1996, Siegwart, 2004] can provide a framework for designing interacting molecular machines with complex behaviors within their environments.

A simple example of a molecular robot would be a "walking" DNA molecule that can recognize and follow an arbitrary trail ("bread crumbs"). If such a simple molecular robot could be demonstrated, its capabilities then could be expanded by incorporating additional layers of control mechanisms from DNA nanotechnology and concepts from computer science. For example, integration of logic and memory into the robot's body would enhance the robot's ability to respond to its environment intelligently [Stojanovic et al., 2002]; interactions between multiple molecular robots could lead to collective behavior [Kube and Zhang, 1993, Rus et al., 2002, Dorigo and Stützle, 2004]; and the ability to read and transform the landscape (e.g., pick up and deposit loads) would in theory provide the essential mechanism for Turing-universal algorithmic behavior [Turing, 1936, Von Neumann and Burks, 1966, Bennett, 1982, Gajardo et al., 2002].

Research in programmable DNA walkers [Bath and Turberfield, 2007] started with non-autonomous remote-controlled systems [Sherman and Seeman, 2004, Shin and Pierce, 2004], progressed to autonomous walkers that modify visited sites to achieve directed (but brief) motion on linear tracks [Bath et al., 2005, Tian et al., 2005, Yin et al., 2008, Omabegho et al., 2009], or to achieve continuous processive (but undirected) motion in two or three dimensions [Pei et al., 2006], and shows promise for processive and directed walking on undisturbed tracks [Green et al., 2008]. While synthesizing suitably well-defined tracks has been an important technical challenge (no previous walker has been demonstrated to take more than three steps on a linear track), our interest here is in how robotic behavior can be obtained from the interaction between a simple random walker and its environment.

In this work, we present an implementation of molecular robots that integrates aspects of DNA-based computing devices [Adleman, 1994, Stojanovic et al., 2003, Seelig et al., 2006, Yin et al., 2008], complex structures [Yan et al., 2003, Rothemund et al., 2004, Seeman, 2005, Rothemund, 2006, Aldaye et al., 2008, He et al., 2008, Jungmann et al., 2008] and actuators [Pei et al., 2006, Ding

and Seeman, 2006]. The DNA walkers chosen for this work, called "molecular spiders", comprise an inert body and multiple catalytic "legs". Specifically, here we use three-legged spiders with a streptavidin body. Spider legs are adapted from DNA enzyme 8-17 that binds and cleaves single-stranded oligodeoxynucleotide substrates with a single ribose moiety into two shorter products that have a lower affinity for the enzyme [Santoro and Joyce, 1997]. In the context of substrates that are immobilized at sites on a surface, spider behavior can be modeled using local rules [Antal, 2007]: a leg bound to substrate will cleave it at a low rate; a leg bound to product will detach at an intermediate rate; and a free leg will quickly bind (with little or no bias) to a nearby substrate or product. For a multipedal spider positioned at the interface between regions of product and substrate, these rules predict that after a given leg cleaves and then lifts, it will by trial-and-error search out a nearby substrate to bind, thus moving the spider's body toward the substrate region while enlarging the product region behind it. A Monte Carlo simulation using these rules is presented further below. On 2D surfaces or in a 3D matrix, such spider movement results in a random walk with memory of visited sites, while on a 1D linear track it results in directed motion as the substrate is consumed. Crucially, unlike related "burnt bridge" Brownian ratchet mechanisms used in DNA walkers [Sherman and Seeman, 2004, Shin and Pierce, 2004, Bath et al., 2005, Tian et al., 2005, Omabegho et al., 2009] and observed in nature [Saffarian et al., 2004], these local rules predict that multipedal spiders will not readily dissociate even from tracks consisting exclusively of product strands, and indeed will perform a rapid unbiased random walk there until they again encounter substrate.

Considering spider legs to be simultaneously sensors that detect nearby oligonucleotides and actuators that modify their environment to inhibit reverse motion, we exploit this sensor-actuator feedback to design prescriptive landscapes that direct the spiders' motion along a predefined path (Figs 1c and d). A spider traversing this landscape of oligonucleotide substrates can sense the set of available cues within its reach and take action accordingly. Here, we show that in the context of a precisely-defined track laid out on two-dimensional (2D) DNA origami [Rothemund, 2006], the previously introduced processive but random walker [Pei et al., 2006] becomes a processive and directed walker capable of path-following behavior. The importance of these results lies not in the walkers reaching stable thermodynamic endpoints, but in reaching those points through

autonomously guided dissipative processes that can be programmed. Such processes could, in the future, be used to couple the behavior of multiple walkers through their interactions with a common landscape.

A.2 Materials and Methods

A.2.1 Abbreviations

iSp18 is a hexa-ethyleneglycol internal spacer; Bio is biotin; and BioTEG is biotin-tetra-ethyleneglycol.

A.2.2 Preparation of Spiders

Materials and Instrumentation for the Preparation and Characterization of NICK$_{3.4A+1}$ and NICK$_{3.4A+1}$· (Cy3)$_3$. Synthesis and purification of the modified DNA strands used to construct NICK$_{3.4A+1}$ and NICK$_{3.4A+1}$· (Cy3)$_3$ were carried out by Integrated DNA Technologies (Coralville, IA) and used as received. Streptavidin was obtained from Pierce, product number 21125 (Rockford, IL). IE-HPLC purification was performed using a Shimadzu LC-6AD pump equipped with a Shimadzu SPD-M10A PDA detector, with separation carried out on an anion exchange TSKgel DEAE-NPR column, 4.6×50 mm (IDxL) (Tosoh Biosciences). Concentrations of oligonucleotides were determined on an Amersham Biosciences Ultrospec 3300 pro UV/visible spectrophotometer. **Assembly of NICK$_{3.4A+1}$.** Part A; capture leg [5′ - GCC GAG AAC CTG ACG CAA GT/iSp18// iSp18//3Bio/ - 3′] (C) (47 nmoles in 10 mL of 10 mM HEPES, 150 mM NaCl, pH 7.4) was added drop-wise to a stirred solution of streptavidin (STV) (5 mg, 94 nmoles in 1 mL of 10 mM K_3PO_4, pH 6.5). The desired one-to-one conjugate product (STV-(C)1) was purified by ion exchange (IE) HPLC. Part B; deoxyribozyme leg [5′ - /5BioTEG//iSp18//iSp18/TCT CTT CTC CGA GCC GGT CGA AAT AGT GAA AA - 3′] (L) (100μM, in water) was titrated into the isolated 1:1 conjugate HPLC fraction from Part A above, until all three remaining biotin binding sites of the 1:1 conjugate STV-(C)$_1$ were occupied by L to give the final desired product STV-(C)$_1$(L)$_3$ i.e. NICK$_{3.4A+1}$. The titration was monitored by IE-HPLC, and was deemed complete when a slight excess of L was observed with no intermediate species, i.e. no STV-(C)$_1$(L)$_1$ or STV-(C)$_1$(L)$_2$, present. The

assembly was purified by IE-HPLC and the volume of the elutent reduced (by centrifugation) to give a final concentration of 2.3μM, as determined by absorbance at 260 nm. Characterization of the assembly was carried out by IE-HPLC and PAGE. The assembly was stable at $-20°$C for at least six months.

Assembly of NICK$_{3.4A+1}$·(Cy3)$_{3^{**}}$. Part A and part B were carried out in identical fashion to the assembly of NICK$_{3.4A+1}$ above, except (C) was [$5'$ /5Cy3/GCC GAG AAC CTG ACG CAA GT/iSp18//iSp18//3Bio/ - $3'$] and triethanolamine (20 mM) was used in place of HEPES and TRIS for the assembly and HPLC purification respectively. Part C; the volume of NICK$_{3.4A+1}$ (Cy3)$_1$, fraction isolated by HPLC, was concentrated to 1 mL (0.834 nmoles) and Cy3 Mono NHS ester (20 nmoles) (PA13101, Lot number 359269, GE Healthcare) dissolved in DMSO added to the solution containing the assembly (giving a total DMSO concentration of 10%). The resulting mixture was incubated at room temperature overnight, protected from light. Excess dye was separated from the NICK$_{3.4A+1}$· (Cy3)$_3$ product by gel filtration (PD-10 column, 17-0851-01, lot 367770, GE Healthcare). Ratio of dye to streptavidin-DNA assembly was obtained by determining concentrations at 550 nm ($\epsilon_{max}150,000$ M^{-1}cm^{-1}) and 260 nm (Extinction coefficient max $1,220,000$ M^{-1}cm^{-1}) respectively.

One should note that the number of Cy3 dyes per spider is an average. This particular protocol sometimes produced an average of four Cy3 dyes per spider molecule, hence such spiders will be notated in the text as NICK$_{3.4A+1}$·(Cy3)$_4$.

A.2.3 Surface Plasmon Resonance (SPR)

Materials and Instrumentation for SPR Experiments. Immunopure avidin was purchased from Pierce (Rockford, USA). We used a Biacore X system, commercially available Biacore SA sensor chips, and Biacore C1 sensor chips, from GE Healthcare (Piscataway, USA). $1\times$ HBS buffer (10 mM HEPES, pH 7.4 with 150 mM NaCl) was employed as running buffer.

Preparation of Substrates on pseudo-2D Hydrogel Matrix Surfaces for SPR. A 20μM solution of cleavable substrates ($5'$-BioTEG-TTTTTTTTCACTATrAGGAAGAG, "r" precedes a ribonucleotide) was applied to both channels of the SA sensor chip (carboxymethylated dextran

matrix pre-immobilized with streptavidin) for 16 minutes at 5μL/min, followed by a 60 second wash with 4 M urea and 15 mM EDTA in both channels to remove any nonspecifically adsorbed materials. The quantity of substrates adsorbed was calculated by the change in measured mass as described [Pei et al., 2006].

Preparation of Substrates on 2D Monolayer Surfaces for SPR. Avidin was covalently bound to the C1 sensor chip surface (a carboxymethylated monolayer) via amino groups using the following protocol. The carboxymethylated surface was first activated at a flow rate of 5μL/min by using a 7 minute injection pulse of an aqueous solution containing N-hydroxysuccinimide (NHS, 0.05 M) and N-ethyl-N′-(dimethylaminopropyl) carbodiimide (EDC, 0.2 M). Next, an 80μL injection of 1 mg/mL avidin (in $1\times$ HBS) was flowed over the activated surfaces of both channels for 40 minutes at 2μL/min. The remaining activated sites on the chip surfaces were blocked with a 35μL injection of an ethanolamine hydrochloride solution (1 M, pH 8.5). Then, a 20μM solution of cleavable substrate was applied to both channels of C1 sensor chip for 20 minutes at 4μL/min, followed by a 60 second wash with 4 M urea and 15 mM EDTA. Based on the average SPR responses for avidin ($2,010$ RU, 0.03 pmole/mm^2) and substrate (450 RU, 0.056 pmole/mm^2), there are two substrates bound for each avidin molecule. The average intersubstrate distance is 5.5 nm.

SPR Monitoring of Dissociation of NICK$_{3.4A+1}$ Spider on Non-cleavable Substrate and Product Surfaces. The non-cleavable substrate analog (substrate in which rA was substituted with A) or product surfaces were prepared in a similar manner to the preparation of substrate on 2D monolayer surfaces. The spider was loaded to channel 2, with channel 1 serving as a negative control. We calculated the ratio of spider to non-cleavable substrate or product by measuring the change in SPR response units (RU) after the spider was flowed onto the chip, then used the equation: ratio $(spider/S$ or $P) = Mw(S$ or $P) \times RU(spider)/[Mw(spider) \times RU(S$ or $P)]$. Monitoring the dissociation of the spider was performed in $1\times$ TA-Mg buffer (40 mM Tris, 20 mM acetic acid, 12.5 mM Magnesium acetate) with 1 mM $ZnCl_2$.

We could not directly measure the dissociation rate of spiders from cleavable substrate because 1) dissociation of the cleavage product from the surface accounts for the vast majority of the SPR response, and 2) the ratio of substrate to cleavage product changes with time, so the dissociation rate of spiders is not constant. Therefore, we instead monitored the SPR response to obtain the

dissociation rate of spider on non-cleavable substrate, and on product. We observed that over the course of 30 minutes more than 92% of spiders remained on a product covered surface and over the course of 60 minutes 86% remained bound. These percentages represent an upper-bound on spider dissociation from our tracks (which will be a mixture of substrates and products as the spider walks over it). So we estimate an upper-bound for the dissociation rate as less than $8-14\%$ over the time scale of our experiments on AFM and fluorescence microscopy.

SPR Monitoring of Cleavage of Substrates by NICK$_{3.4A+1}$ Spider. Spiders (0.8 6.3 nM in $1\times$ HBS buffer) were loaded only on channel 2 at 5μL/min, with channel 1 used as a negative control. The amount of spider applied was controlled by adjusting concentrations and the reaction times of spiders in the loading solution. Monitoring the cleavage of the substrate was initiated by switching to $1\times$ TA-Mg buffer with 1 mM $ZnCl_2$ or $1\times$ HBS buffer with 1 mM $ZnCl_2$ with the Biacore X system "Working Tools Wash". Product formation in real time was measured through the decrease in mass, using the formula $1,000$ RU = 1 ng·mm^2. Rates of cleavage were determined from the approximately linear region of the product release curves during the initial 10% of substrates cleaved. On the 2D monolayer surface, real-time processivity of spiders was measured to be 79% (percentage of total substrate cleaved over the course of the experiment) at a 1:291 ratio of spider (17.8 RU) to substrate (448.4 RU) with a cleavage rate of 1.42 min^{-1} per spider. On the pseudo-2D matrix surface, spiders showed a real-time processivity: 86% of total substrate cleaved at a 1:990 ratio of spider (26 RU) to substrate ($2,222$ RU) with a cleavage rate of 2.81 min^{-1} per spider.

A.2.4 Preparation of Spider-Origami Arrays

Assembly of Spider-Origami Arrays for Atomic Force Microscopy (AFM). The spider arrays consist of M13mp18 viral DNA (New England Biolabs) and 202 ssDNA staples (Integrated DNA Technologies, see below for DNA sequences). The arrays were annealed in $1\times$ TA-Mg Buffer (40 mM Tris, 20 mM acetic acid, 12.5 mM Mg^{++}, pH 7.6) using a 1:3 ratio of M13 to staple strands and a final concentration of 10 nM (M13). The arrays were annealed in two hours from 94°C to 25°C using an Eppendorf PCR machine (Eppendorf). The NICK$_{3.4A+1}$ or NICK$_{3.4A+1}$·(Cy3)$_3$ were then added to the arrays at a 1:1 ratio of START strand to spider and left at room temperature

overnight. Because origami folding is sensitive to stoichiometry, we expect that some fraction of origami are missing the START strand and are thus unable to position a spider before the TRACK is deposited. The substrate strand and CONTROL strand were then added at a 1:1 (for initial ABD, EABC and Before EABD samples) or 1:3 (for 15, 30 and 60 minute EABD samples) ratio of staple probes to substrate or CONTROL and allowed to bind overnight at room temperature (20°C to 24°C). We observed (by AFM) a larger percentage of apparently unbroken TRACKS when excess substrate was added. In the presence of excess substrate there is a low probability that a spider leg may bind to a free floating substrate or STOP strand that would deter or inhibit interactions with the TRACK. Note that the 8-17 deoxyribozyme has reduced but non-negligible activity in TA-Mg buffer (relative to maximal activity with Zn^{++}), suggesting that spiders bound at START may cleave immediately neighboring substrates during the overnight incubation. Since spiders undergo (unbiased) walks on product tracks with little dissociation, this possibility is not a concern. To minimize stacking interactions that can cause aggregation of origami, the staples on the left and right edges of the origami were removed.

Modification of Spider-Origami Arrays for Fluorescence Microscopy. To make the origami arrays compatible with fluorescence microscopy, we returned four of the removed staples to the corners of the origami. In order to affix the origami to slides for analysis, we divided the corresponding staples into two strands so that we could affix biotin labels onto the 5× end that is antiparallel to staple probes. We modified the CONTROL strand by adding a Cy5 fluorophore to its 3′ end, which resulted in 6 Cy5 fluorophores labeling the STOP position. On all landscapes, CONTROL staples were replaced with staples lacking the non-cleavable substrate probes. The EAC landscape used in both fluorescence microscopy and AFM experiments lacked a CONTROL site. In addition, the EAC arrays for fluorescence microscopy were annealed in 5× SSC buffer (75 mM sodium citrate, pH 7.0, 750 mM NaCl), and the EABC and EABD arrays in 1× TA-Mg buffer. Fluorescence microscopy was also performed for origami arrays containing a truncated substrate TRACK, or product TRACK. The product strand is 8 nucleotides shorter than the full length substrate and includes only the sequence 5′ of the RNA base. The resulting 31 oligonucleotides have the same sequence as the corresponding portion of the full length cleavable substrate. All other assembly details for origami arrays for fluorescence microscopy including DNA concentrations,

relative strand ratios, and binding conditions were unchanged.

A.2.5 Atomic Force Microscopy

AFM Imaging. "Before" samples were deposited on mica without the addition of TRIGGER or $ZnCl_2$. "After" samples were prepared by releasing the spider from the START strand through the addition of a 27-base TRIGGER strand, immediately followed by the addition of 10 mM $ZnCl_2$ to a final concentration of 1 mM. Spiders were allowed to traverse the product or substrate TRACK array in solution for 15, 30, or 60 minutes (depending on the experiment) at room temperature before the origami were deposited on mica. Samples (2μL) were deposited onto a freshly cleaved mica surface (Ted Pella, Inc.) and left to adsorb for 3 minutes. Buffer ($1\times$ TA-Mg, 400μL) was added to the liquid cell and the sample was scanned in tapping mode on a Pico-Plus AFM (Molecular Imaging, Agilent Technologies) with NP-S tips (Veeco, Inc.). Each sample was scanned for 2-3 hrs before being discarded (therefore "30 minutes after" means that the sample spent 30 minutes in solution followed by up to 3 hours on mica). Note that the reduced but non-negligible deoxyribozyme cleavage rate in TA-Mg raises the possibility that spiders could move during the this imaging period; however, given the apparent difficulty of spider movement on mica-bound origami even in the presence of Zn^{++} (see AFM Imaging for Movie) and the consistent trends in the time-lapse experiments (Fig. 2.2, main text), we conclude that very little movement takes place during the imaging period. All imaging by AFM was carried out at room temperature.

AFM Imaging for Movie. The sample (2μL) was deposited onto a freshly cleaved mica surface and left to bind for 2 minutes. Then 1μL of TRIGGER strand was added to the sample on the surface and after 2 minutes 270μL of buffer and 30μL of 10 mM $ZnCl_2$ was added to the sample cell. The four images were taken over a 26-minute time frame with about 10 minutes between the saving of each scan. (It should be noted that many prior and subsequent attempts were made to capture another AFM movie using various optimizations of our buffer, and protocol, without success.) Although we were only able to capture one movie, reported in Fig. 2.3, we are convinced that it is not an artifact. The origami with the moving spider is substrate face-up while the three origami in the same image are substrate side down (see below for a discussion of how the face of the

origami affects spider analysis). As a result spiders on the three adjacent origami are stationary over the time course of the movie. In addition the spider's motion follows the TRACK in each frame (therefore it is not randomly diffusing, because it neither moves backwards nor off the TRACK). If the AFM tip were merely pushing the spider forward we would not expect the spider to turn in the transition from frame 3 to frame 4.

AFM Time Lapse Experiments. There is one seeming contradiction in our report that we would like to address here. If we were to suggest (as we do in Fig. 2.3) that the spider can walk on origami deposited on mica, then how could we expect to obtain viable statistics from time lapse experiments imaged for up to 3 hours? We assume that under these conditions, most spiders get stuck on the origami, while some small percentage of spiders are able to continue moving. We find that we can differentiate between samples deposited at 15 minutes from those deposited at 30 and 60 minutes. These results help to explain why obtaining the AFM movie was so difficult.

Statistical Analysis of AFM Images. We divided our flattened AFM images into $1 \times 1\mu$m images and numbered them. Within each of these images, we assigned a roman character to each origami (thus each origami we analyzed could be uniquely identified by a number and letter). The origami arrays were classified by the following criteria: orientation (is the origami "face-up" or "face-down"?), number of spiders (0,1, multiple), location of spiders (START, TRACK, STOP, CONTROL), image quality (do imaging errors or sample impurities make the classification difficult?). This process was conducted independently by three people, for each data set excluding the EABD 15 minute and EABD 60 minute data sets, which were conducted by two people. The classifications were then compared: if two or more people agreed on the origami classification it was held, otherwise the origami was discarded from further analysis. By this method, we sought to ensure that our results are neither subjective nor irreproducible. While it is possible that some putative spiders were actually image artifacts or molecular contaminants, it is unlikely that this inaccuracy in our measurements could affect the main trends in our data or the qualitative conclusions we drew from them.

An origami that is "face up" is one that displays its substrates and spiders on the face opposite the mica; an origami that is "face down" displays its substrates on the face that rests on the mica. Orientation was determined by landscape asymmetries in the positions of the TRACK and marker.

By analyzing the statistics of origami classification, we concluded that the probability of an origami landing on one face or the other was approximately equal. However, we discovered that "face down" origami appeared to have a larger number of spiders at the STOP. We conducted a double-blind study in which six researchers were given an AFM image of origami and asked to classify these according to our criteria. We discovered that in the absence of spiders, all "face-up" origami were classified as vacant while a significant portion of "face-down" origami were classified as displaying a spider at the STOP site, when in fact there was none. Due to this "false positive" effect, we did not count "face-down" origami in our statistics. Approximately 50% of "face-up" origami were unoccupied by any spiders, and between 0 and 7% displayed more than one spider on the TRACK. Because the quantity of multiply occupied origami was small compared to the quantity of unoccupied and singly-occupied origami, we only considered singly-occupied origami to simplify our analysis (Fig. 2.2).

Experimental results for all four landscapes with substrate TRACKS showed that the fraction of spiders at the START diminishes with a concomitant increase in spiders observed on the STOP positions (Fig. 2.2c,g, main text). Our shortest track (ABD, spanning 48 nm) efficiently delivers spiders to the STOP, with less than 20% of spiders on the TRACK after 30 minutes (Fig. 2.2c, main text). If the TRACK was omitted on the ABD landscape, spiders were equally distributed between the STOP and CONTROL sites after 30 minutes, implying that the track is needed for efficient delivery to the STOP site. On longer TRACKS (such as EABD, spanning 90 nm) 15% of spiders are delivered to the STOP within 15 minutes after release. Longer incubation times (30 and 60 minutes) increase the efficacy of delivering spiders to the STOP to up to 70%, (Fig. 2.2c,g, main text). Even at 60 minutes, however, we observed between 10-15% of spiders still on the TRACK. This outcome could be attributed to the distribution of spider velocities resulting from the stochastic nature of individual walks and possibly from backward steps onto product, initiating an unbiased random walk on product. We observed no significant difference in the efficacy of "turn right" and "turn left" actions (paths EABD and EABC, respectively) 30 minutes after release (Fig. 2.2c,g, main text).

A.2.6 DNA Sequences

Name	Sequence
1	TTTTCGATGGCCCACTACGTAAACCGTC
2	TATCAGGGTTTTCGGTTTGCGTATTGGGAACGCGCG
3	GGGAGAGGTTTTTGTAAAACGACGGCCATTCCCAGT
3A	GGGAGAGGTTTTTGTAAAAC
3B	Biotin GACGGCCATTCCCAGT
4	CACGACGTTTTTGTAATGGGATAGGTCAAAACGGCG
5	GATTGACCTTTTGATGAACGGTAATCGTAGCAAACA
6	AGAGAATCTTTTGGTTGTACCAAAAACAAGCATAAA
7	GCTAAATCTTTTCTGTAGCTCAACATGTATTGCTGA
8	ATATAATGTTTTCATTGAATCCCCCTCAAATCGTCA
9	TAAATATTTTTTGGAAGAAAAATCTACGACCAGTCA
10	GGACGTTGTTTTTCATAAGGGAACCGAAAGGCGCAG
11	ACGGTCAATTTTGACAGCATCGGAACGAACCCTCAG
11A	ACGGTCAATTTTGACAGCAT
11B	Biotin CGGAACGAACCCTCAG
12	CAGCGAAAATTTTACTTTCAACAGTTTCTGGGATTTTGCTAAACTTTT
13	TGGTTTTTAACGTCAAAGGGCGAAGAACCATC
14	CTTGCATGCATTAATGAATCGGCCCGCCAGGG
15	TAGATGGGGGGTAACGCCAGGGTTGTGCCAAG
16	CATGTCAAGATTCTCCGTGGGAACCGTTGGTG
17	CTGTAATATTGCCTGAGAGTCTGGAAAACTAG
18	TGCAACTAAGCAATAAAGCCTCAGTTATGACC
19	AAACAGTTGATGGCTTAGAGCTTATTTAAATA
20	ACGAACTAGCGTCCAATACTGCGGAATGCTTT
21	CTTTGAAAAGAACTGGTCCTCTTTTGAGGAACAAGTTTTCTTGT CTCATTATTTAATAAA

Name	Sequence
22	ACGGCTACTTACTTAGTCCTCTTTTGAGGAACAAGTTTTCTTGT CCGGAACGCTGACCAA
23	GAGAATAGCTTTTGCGGGATCGTCGGGTAGCA
24	ACGTTAGTAAATGAATTTTCTGTAAGCGGAGT
25	ACCCAAATCAAGTTTTTTGGGGTCAAAGAACG
26	TGGACTCCCTTTTCACCAGTGAGACCTGTCGT
27	GCCAGCTGCCTGCAGGTCGACTCTGCAAGGCG
28	ATTAAGTTCGCATCGTAACCGTGCGAGTAACA
29	ACCCGTCGTCATATGTACCCCGGTAAAGGCTA
30	TCAGGTCACTTTTGCGGGAGAAGCAGAATTAG
31	CAAAATTAAAGTACGGTGTCTGGAAGAGGTCA
32	TTTTTGCGCAGAAAACGAGAATGAATGTTTAG
33	ACTGGATAACGGAACAACATTATTACCTTATG
34	CGATTTTAGAGGACAGTCCTCTTTTGAGGAACAAGTTTTCTTGT ATGAACGGCGCGACCT
35	GCTCCATGAGAGGCTTTCCTCTTTTGAGGAACAAGTTTTCTTGT TGAGGACTAGGGAGTT
36	AAAGGCCGAAAGGAACAACTAAAGCTTTCCAG
37	AGCTGATTACAAGAGTCCACTATTGAGGTGCC
38	CCCGGGTACTTTCCAGTCGGGAAACGGGCAAC
39	GTTTGAGGGAAAGGGGGATGTGCTAGAGGATC
40	AGAAAAGCAACATTAAATGTGAGCATCTGCCA
41	CAACGCAATTTTTGAGAGATCTACTGATAATC
42	TCCATATACATACAGGCAAGGCAACTTTATTT
43	CAAAAATCATTGCTCCTTTTGATAAGTTTCAT
44	AAAGATTCAGGGGGTAATAGTAAACCATAAAT
45	CCAGGCGCTTAATCATTCCTCTTTTGAGGAACAAGTTTTCTTGT TGTGAATTACAGGTAG
46	TTTCATGAAAATTGTGTCCTCTTTTGAGGAACAAGTTTTCTTGT TCGAAATCTGTACAGA
47	AATAATAAGGTCGCTGAGGCTTGCAAAGACTT

Name	Sequence
48	CGTAACGATCTAAAGTTTTGTCGTGAATTGCG
49	GTAAAGCACTAAATCGGAACCCTAGTTGTTCC
50	AGTTTGGAGCCCTTCACCGCCTGGTTGCGCTC
51	ACTGCCCGCCGAGCTCGAATTCGTTATTACGC
52	CAGCTGGCGGACGACGACAGTATCGTAGCCAG
53	CTTTCATCCCCAAAAACAGGAAGACCGGAGAG
53A	CTTTCATCCCCAAAAA
53B	Biotin CAGGAAGACCGGAGAG
54	GGTAGCTAGGATAAAAATTTTTAGTTAACATC
55	CAATAAATACAGTTGATTCCCAATTTAGAGAG
56	TACCTTTAAGGTCTTTACCCTGACAAAGAAGT
57	TTTGCCAGATCAGTTGAGATTTAGTGGTTTAA
57A	TTTGCCAGATCAGTTG
57B	Biotin AGATTTAGTGGTTTAA
58	TTTCAACTATAGGCTGGCTGACCTTGTATCAT
59	CGCCTGATGGAAGTTTCCATTAAACATAACCG
60	ATATATTCTTTTTTCACGTTGAAAATAGTTAG
61	GAGTTGCACGAGATAGGGTTGAGTAAGGGAGC
62	TCATAGCTACTCACATTAATTGCGCCCTGAGA
63	GAAGATCGGTGCGGGCCTCTTCGCAATCATGG
64	GCAAATATCGCGTCTGGCCTTCCTGGCCTCAG
65	TATATTTTAGCTGATAAATTAATGTTGTATAA
66	CGAGTAGAACTAATAGTAGTAGCAAACCCTCA
67	TCAGAAGCCTCCAACAGGTCAGGATCTGCGAA
68	CATTCAACGCGAGAGGCTTTTGCATATTATAG
69	AGTAATCTTAAATTGGGCTTGAGAGAATACCA

Name	Sequence
70	ATACGTAAAAGTACAACGGAGATTTCATCAAG
71	AAAAAAGGACAACCATCGCCCACGCGGGTAAA
72	TGTAGCATTCCACAGACAGCCCTCATCTCCAA
73	CCCCGATTTAGAGCTTGACGGGGAAATCAAAA
74	GAATAGCCGCAAGCGGTCCACGCTCCTAATGA
75	GTGAGCTAGTTTCCTGTGTGAAATTTGGGAAG
76	GGCGATCGCACTCCAGCCAGCTTTGCCATCAA
77	AAATAATTTTAAATTGTAAACGTTGATATTCA
78	ACCGTTCTAAATGCAATGCCTGAGAGGTGGCA
79	TCAATTCTTTTAGTTTGACCATTACCAGACCG
80	GAAGCAAAAAAGCGGATTGCATCAGATAAAAA
81	CCAAAATATAATGCAGATACATAAACACCAGA
82	ACGAGTAGTGACAAGAACCGGATATACCAAGC
83	GCGAAACATGCCACTACGAAGGCATGCGCCGA
84	CAATGACACTCCAAAAGGAGCCTTACAACGCC
85	CCAGCAGGGGCAAAATCCCTTATAAAGCCGGC
86	GCTCACAATGTAAAGCCTGGGGTGGGTTTGCC
87	GCTTCTGGTCAGGCTGCGCAACTGTGTTATCC
88	GTTAAAATTTTAACCAATAGGAACCCGGCACC
89	AGGTAAAGAAATCACCATCAATATAATATTTT
90	TCGCAAATGGGGCGCGAGCTGAAATAATGTGT
91	AAGAGGAACGAGCTTCAAAGCGAAGATACATT

Name	Sequence
92	GGAATTACTCGTTTACCAGACGACAAAAGATT
93	CCAAATCACTTGCCCTGACGAGAACGCCAAAA
94	AAACGAAATGACCCCCAGCGATTATTCATTAC
95	TCGGTTTAGCTTGATACCGATAGTCCAACCTA
96	TGAGTTTCGTCACCAGTACAAACTTAATTGTA
97	GAACGTGGCGAGAAAGGAAGGGAACAAACTAT
98	CCGAAATCCGAAAATCCTGTTTGAAGCCGGAA
99	GCATAAAGTTCCACACAACATACGAAGCGCCA
100	TTCGCCATTGCCGGAAACCAGGCATTAAATCA
101	GCTCATTTTCGCATTAAATTTTTGAGCTTAGA
102	AGACAGTCATTCAAAAGGGTGAGAAGCTATAT
103	TTTCATTTGGTCAATAACCTGTTTATATCGCG
103A	TTTCATTTGGTCAATA
103B	Biotin ACCTGTTTATATCGCG
104	TTTTAATTGCCCGAAAGACTTCAAAACACTAT
105	CATAACCCGAGGCATAGTAAGAGCTTTTTAAG
106	GAATAAGGACGTAACAAAGCTGCTCTAAAACA
107	CTCATCTTGAGGCAAAAGAATACAGTGAATTT
108	CTTAAACATCAGCTTGCTTTCGAGCGTAACAC
109	ACGAACCAAAACATCGCCATTAAATGGTGGTT
110	CGACAACTAAGTATTAGACTTTACAATACCGA
111	CTTTTACACAGATGAATATACAGTAAACAATT
112	TTAAGACGTTGAAAACATAGCGATAACAGTAC
113	GCGTTATAGAAAAAGCCTGTTTAGAAGGCCGG

Name	Sequence
114	ATCGGCTGCGAGCATGTAGAAACCTATCATAT
115	CCTAATTTACGCTAACGAGCGTCTAATCAATA
116	AAAAGTAATATCTTACCGAAGCCCTTCCAGAG
117	TTATTCATAGGGAAGGTAAATATTCATTCAGT
118	GAGCCGCCCCACCACCGGAACCGCGACGGAAA
119	AATGCCCCGTAACAGTGCCCGTATCTCCCTCA
120	CAAGCCCAATAGGAACCCATGTACAAACAGTT
121	CGGCCTTGCTGGTAATATCCAGAACGAACTGA
122	TAGCCCTACCAGCAGAAGATAAAAACATTTGA
123	GGATTTAGCGTATTAAATCCTTTGTTTTCAGG
124	TTTAACGTTCGGGAGAAACAATAATTTTCCCT
125	TAGAATCCCTGAGAAGAGTCAATAGGAATCAT
126	AATTACTACAAATTCTTACCAGTAATCCCATC
127	CTAATTTATCTTTCCTTATCATTCATCCTGAA
128	TCTTACCAGCCAGTTACAAAATAAATGAAATA
129	GCAATAGCGCAGATAGCCGAACAATTCAACCG
130	ATTGAGGGTAAAGGTGAATTATCAATCACCGG
128	AACCAGAGACCCTCAGAACCGCCAGGGGTCAG
132	TGCCTTGACTGCCTATTTCGGAACAGGGATAG
133	AGGCGGTCATTAGTCTTTAATGCGCAATATTA
134	TTATTAATGCCGTCAATAGATAATCAGAGGTG

Name	Sequence
135	CCTGATTGAAAGAAATTGCGTAGACCCGAACG
136	ATCAAAATCGTCGCTATTAATTAACGGATTCG
137	ACGCTCAAAATAAGAATAAACACCGTGAATTT
138	GGTATTAAGAACAAGAAAAATAATTAAAGCCA
139	ATTATTTAACCCAGCTACAATTTTCAAGAACG
140	GAAGGAAAATAAGAGCAAGAAACAACAGCCAT
141	GACTTGAGAGACAAAAGGGCGACAAGTTACCA
142	GCCACCACTCTTTTCATAATCAAACCGTCACC
143	CTGAAACAGGTAATAAGTTTTAACCCCTCAGA
144	CTCAGAGCCACCACCCTCATTTTCCTATTATT
145	CCGCCAGCCATTGCAACAGGAAAAATATTTTT
146	GAATGGCTAGTATTAACACCGCCTCAACTAAT
147	AGATTAGATTTAAAAGTTTGAGTACACGTAAA
148	ACAGAAATCTTTGAATACCAAGTTCCTTGCTT
149	CTGTAAATCATAGGTCTGAGAGACGATAAATA
150	AGGCGTTACAGTAGGGCTTAATTGACAATAGA
151	TAAGTCCTACCAAGTACCGCACTCTTAGTTGC
152	TATTTTGCTCCCAATCCAAATAAGTGAGTTAA
153	GCCCAATACCGAGGAAACGCAATAGGTTTACC
154	AGCGCCAACCATTTGGGAATTAGATTATTAGC
155	GTTTGCCACCTCAGAGCCGCCACCGATACAGG
156	AGTGTACTTGAAAGTATTAAGAGGCCGCCACC

Name	Sequence
157	GCCACGCTATACGTGGCACAGACAACGCTCAT
158	ATTTTGCGTCTTTAGGAGCACTAAGCAACAGT
159	GCGCAGAGATATCAAAATTATTTGACATTATC
160	TAACCTCCATATGTGAGTGAATAAACAAAATC
160A	TAACCTCCATATGTGA
160B	Biotin GTGAATAAACAAAATC
161	CATATTTAGAAATACCGACCGTGTTACCTTTT
162	CAAGCAAGACGCGCCTGTTTATCAAGAATCGC
163	TTTTGTTTAAGCCTTAAATCAAGAATCGAGAA
164	ATACCCAAGATAACCCACAAGAATAAACGATT
164A	ATACCCAAGATAACCC
164B	Biotin ACAAGAATAAACGATT
165	AATCACCAAATAGAAAATTCATATATAACGGA
166	CACCAGAGTTCGGTCATAGCCCCCGCCAGCAA
167	CCTCAAGAATACATGGCTTTTGATAGAACCAC
168	CCCTCAGAACCGCCACCCTCAGAACTGAGACT
169	GGAAATACCTACATTTTGACGCTCACCTGAAA
170	GCGTAAGAGAGAGCCAGCAGCAAAAAGGTTAT
171	CTAAAATAGAACAAAGAAACCACCAGGGTTAG
172	AACCTACCGCGAATTATTCATTTCCAGTACAT
173	AAATCAATGGCTTAGGTTGGGTTACTAAATTT
174	AATGGTTTACAACGCCAACATGTAGTTCAGCT
175	AATGCAGACCGTTTTTATTTTCATCTTGCGGG
176	AGGTTTTGAACGTCAAAAATGAAAGCGCTAAT
177	ATCAGAGAAAGAACTGGCATGATTTTATTTTG

Name	Sequence
178	TCACAATCGTAGCACCATTACCATCGTTTTCA
179	TCGGCATTCCGCCGCCAGCATTGACGTTCCAG
180	TAAGCGTCGAAGGATTAGGATTAGTACCGCCA
181	CTAAAGCAAGATAGAACCCTTCTGAATCGTCT
182	CGGAATTATTGAAAGGAATTGAGGTGAAAAAT
183	GAGCAAAAACTTCTGAATAATGGAAGAAGGAG
184	TATGTAAACCTTTTTTAATGGAAAAATTACCT
185	AGAGGCATAATTTCATCTTCTGACTATAACTA
186	TCATTACCCGACAATAAACAACATATTTAGGC
187	CTTTACAGTTAGCGAACCTCCCGACGTAGGAA
188	TTATTACGGTCAGAGGGTAATTGAATAGCAGC
189	CCGGAAACACACCACGGAATAAGTAAGACTCC
190	TGAGGCAGGCGTCAGACTGTAGCGTAGCAAGG
191	TGCTCAGTCAGTCTCTGAATTTACCAGGAGGT
192	TATCACCGTACTCAGGAGGTTTAGCGGGGTTT
193	GAAATGGATTATTTACATTGGCAGACATTCTG
194	GCCAACAGTCACCTTGCTGAACCTGTTGGCAA
195	ATCAACAGTCATCATATTCCTGATTGATTGTT
196	TGGATTATGAAGATGATGAAACAAAATTTCAT
197	TTGAATTATGCTGATGCAAATCCACAAATATA
198	TTTTAGTTTTTCGAGCCAGTAATAAATTCTGT

Name	Sequence
199	CCAGACGAGCGCCCAATAGCAAGCAAGAACGC
200	GAGGCGTTAGAGAATAACATAAAAGAACACCC
201	TGAACAAACAGTATGTTAGCAAACTAAAAGAA
202	ACGCAAAGGTCACCAATGAAACCAATCAAGTT
203	TGCCTTTAGTCAGACGATTGGCCTGCCAGAAT
204	GGAAAGCGACCAGGCGGATAAGTGAATAGGTG
205	AAACCCTCTTTTACCAGTAATAAAAGGGATTCACCAGTCACACGTTTT
206	GATGGCAATTTTAATCAATATCTGGTCACAAATATC
206A	GATGGCAATTTTAATCAATA
206B	Biotin TCTGGTCACAAATATC
207	AAAACAAATTTTTTCATCAATATAATCCTATCAGAT
208	ACAAAGAATTTTATTAATTACATTTAACACATCAAG
209	TAAAGTACTTTTCGCGAGAAAACTTTTTATCGCAAG
210	TATAGAAGTTTTCGACAAAAGGTAAAGTAGAGAATA
211	GCGCATTATTTTGCTTATCCGGTATTCTAAATCAGA
212	TACATACATTTTGACGGGAGAATTAACTACAGGGAA
213	AGCACCGTTTTTTAAAGGTGGCAACATAGTAGAAAA
214	ACAAACAATTTTAATCAGTAGCGACAGATCGATAGC
214A	ACAAACAATTTTAATCAGTA
214B	Biotin GCGACAGATCGATAGC
215	AGGGTTGATTTTATAAATCCTCATTAAATGATATTC
216	TTTTTATAAGTATAGCCCGGCCGTCGAG
217	AACATCACTTGCCTGAGTAGAAGAACT
218	TGTAGCAATACTTCTTTGATTAGTAAT
219	AGTCTGTCCATCACGCAAATTAACCGT

Name	Sequence
220	ATAATCAGTGAGGCCACCGAGTAAAAG
221	ACGCCAGAATCCTGAGAAGTGTTTTT
222	TTAAAGGGATTTTAGACAGGAACGGT
223	AGAGCGGGAGCTAAACAGGAGGCCGA
224	TATAACGTGCTTTCCTCGTTAGAATC
225	GTACTATGGTTGCTTTGACGAGCACG
226	GCGCTTAATGCGCCGCTACAGGGCGC

The following three sequences are attached to the $5'$ end of the staple sequences, as a probe, for the START position, binding of the cleavable substrate, and binding of the non-cleavable substrate. For fluorescence microscopy, strands 3A, 3B, 11A, 11B, 206A, 206B, 214A, 214B were incorporated into the origami and CONTROL staples were replaced with staples lacking the non-cleavable substrate probes.

Spider START (green)
$5'$- GATGTCTACTTGCGTCAGGTTCTCGGC[staple]

Spider Cleavable Substrate Probes (brown)
$5'$- CCTCTCACCCACCATTCATC[staple]

Spider Non-Cleavable Substrate Probes (for STOP and CONTROL; red)
$5'$- GGTTCAGTTCGTTGAGCCAG[staple]

Spider Cleavable Substrate
$5'$- GATGAATGGTGGGTGAGAGGTTTTTCACTATrAGGAAGAG

Spider Non-Cleavable Substrate (STOP and CONTROL)

5′- CTGGCTCAACGAACTGAACC TTTTTCACTATAGGAAGAG

Spider Non-Cleavable Substrate (STOP) for fluorescence microscopy

5′- CTGGCTCAACGAACTGAACC TTTTTCACTATAGGAAGAG-Cy5

Spider TRIGGER Strand

5′- GCCGAGAACCTGACGCAAGTAGACATC

Appendix B

Supplementary Material for A Synthetic Polymer that Grows Exponentially Fast[0]

[0]This work was coauthored by Nadine Dabby & Ho-Lin Chen*, and is in preparation [Dabby and Chen, 2013a] with the following contributions: experiments and analysis were performed by N.D. with supervision from H-L.C.; manuscript was written with input from both authors.

B.1 Exponential Growth System Experiments

B.1.1 DNA Sequences Final Version 6-3v1

Table B.1: 6-3 Toehold Design Version1

m6n3 Insertion Init1.v1	ACCGCACGTCCACGGTGTCGCACCCAC
m6n3 Insertion Init2.v1	AACGCGACACCGTGGACGTGCGGT
m6n3 Insertion Init2.ROX	AACGCGACACCGTGGACGTGCGGT /3Rox_N/
m6n3 Insertion_H1.v1	GTGGGTGCGACACCGTGGACGTGCCTCAGACCAAGAGCACGTCCACGGTGTCGCGTT
m6n3 Insertion_H2.v1	TCTGAGGCACGTCCACGGTGTCGCACCCACAACGCGACACCGTGGACGTGCGGT
m6n3 Insertion_H3.v1	ACCGCACGTCCACGGTGTCGCACCCACAACGCGACACCGTGGACGTGCTCTTGG
m6n3 Divide_H1-3'	CCAAGAGCACGTCCACGGTGTCGCGTT
m6n3 Divide_H1-5'	GTGGGTGCGACACCGTGGACGTGCCTCAGA
m6n3v1.H2-Rox-Quench	/5IAbRQ/TCTGAGGCACGTCCACGGTGTCGCACCCACAACGCGACACCGTGGACGTGCGGT/3Rox_N/
m6n3 Linear_H3.v1	ACCGCACGTCCACGGTGTCGCTTTTTTTTTGCGACACCG TGG ACG TGC TCT TGG
m6n3v1.H2RQ_DISPLACE	ACCGCACGTCCACGG TGT CGC GTT GTG GGT GCG ACA CCG TGG ACG TGC CTC AGA

B.1.2 Leakage in Various Designs

Leakage and rate trials of 3 other Hairpin Designs: With Sequences.

Table B.2: 4-4 Toehold Design Version1

m4n4 Insertion_init1	GAGAGCACGTCCACGGTGTCGCACCC
m4n4 Insertion_init2	ACAGGCGACACCGTGGACGTGCTCTC
m4n4 H1	GGGTGCGACACCGTGGACGTGCCAGCCTCGGCACGTCCACGGTGTCGCCTGT
m4n4 H2	GCTGGCACGTCCACGGTGTCGCACCCACAGGCGACACCGTGGACGTGCTCTC
m4n4 H3	GAGAGCACGTCCACGGTGTCGCACCCACAGGCGACACCGTGGACGTGCCGAG

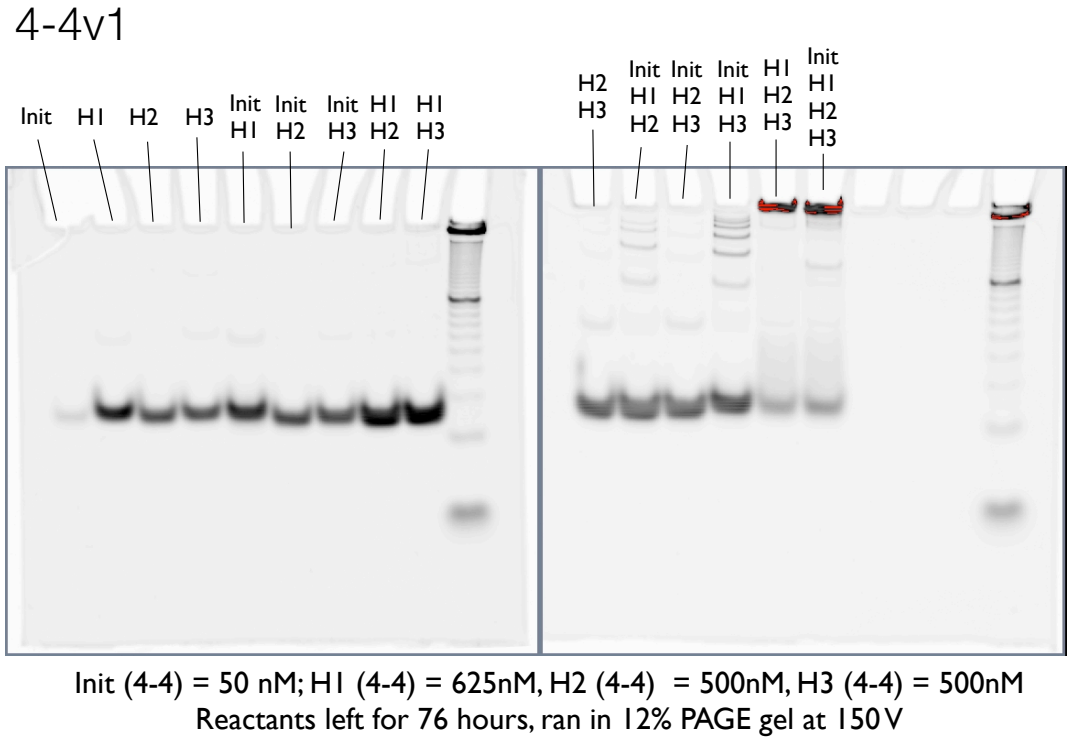


Figure B.1: Combinatorial gel for 4-4v1 design. The polyacrylamide gel above shows that a small leak occurs between reactants Hairpin 1 (H1) and Hairpin 2 (H2) and Hairpin 3 (H3) in the absence of Initiator. This set of strands reacts significantly more slowly than the other designs.

Table B.3: 5-4 Toehold Design Version2

5-4 v1.init1	GTTA-GCCCTGTATTGGGCTCGC-TCTCG
5-4 v1.init2	GCCT-GCGAGCCCAATACAGGGC-TAAC
5-4 v1.H1	CGAGA-GCGAGCCCAATACAGGGC-ACTCA-ATCAC-GCCCTGTATTGGGCTCGC-AGGC
5-4 v1.H2	TGAGT-GCCCTGTATTGGGCTCGC-TCTCG-GCCT-GCGAGCCCAATACAGGGC-TAAC
5-4 v1.H3	GTTA-GCCCTGTATTGGGCTCGC-TCTCG-GCCT-GCGAGCCCAATACAGGGC-GTGAT

5-4v2

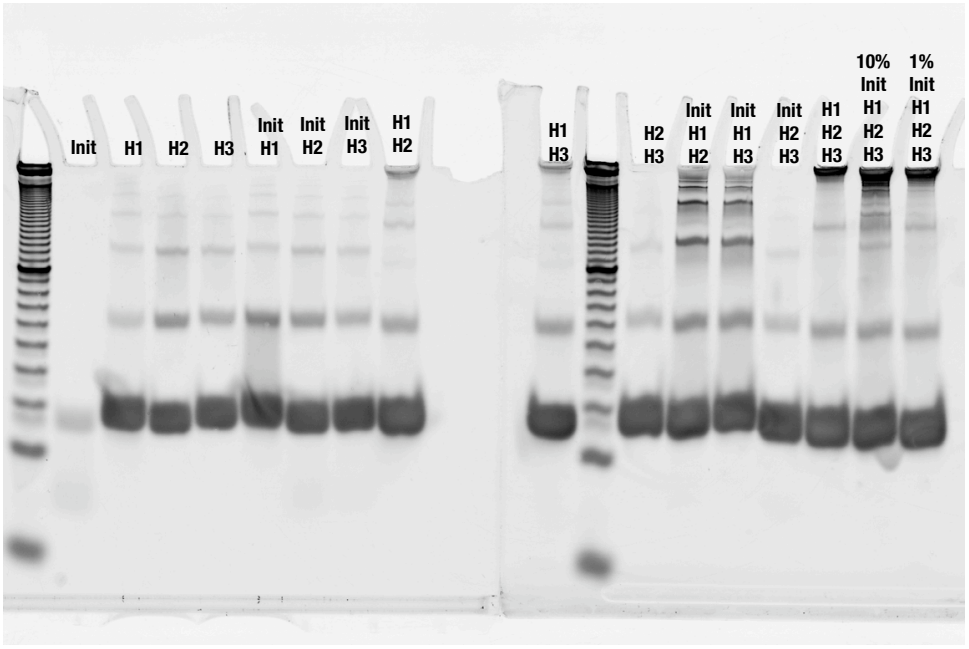


Figure B.2: Combinatorial gel for 5-4v2 design. The polyacrylamide gel above shows that a small leak occurs between reactants Hairpin 1 (H1) and Hairpin 2 (H2) and between Hairpin 1 (H1) and Hairpin 3 (H3) in the absence of Initiator.

Table B.4: 6-3 Toehold Design Version 2

6-3v2.init1	GTC-CGGGACGGACCCGTGCGC-CTTACG
6-3v2.init2	CTT-GCGCACGGGTCCGTCCCG-GAC
6-3v2.H1	CGTAAG-GCGCACGGGTCCGTCCCG-TGTCCA-AGCTAG-CGGGACGGACCCGTGCGC-AAG
6-3v2.H2	TGGACA-CGGGACGGACCCGTGCGC-CTTACG-CTT-GCGCACGGGTCCGTCCCG-GAC
6-3v2.H3	GTC-CGGGACGGACCCGTGCGC-CTTACG-CTT-GCGCACGGGTCCGTCCCG-CTAGCT

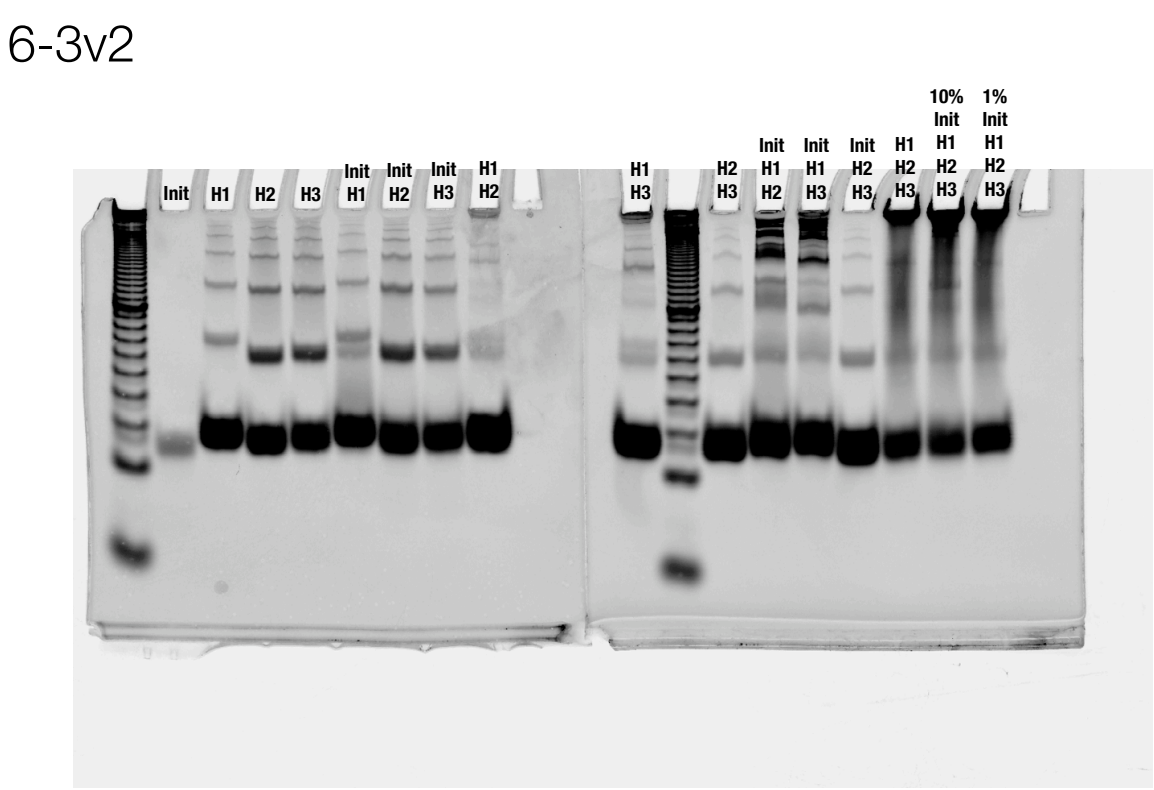


Figure B.3: Combinatorial gel for 6-3v2 design. The polyacrylamide gel above shows that a small leak occurs between reactants Hairpin 1 (H1) and Hairpin 2 (H2) and between Hairpin 1 (H1) and Hairpin 3 (H3) in the absence of Initiator.

B.1.3 Joining

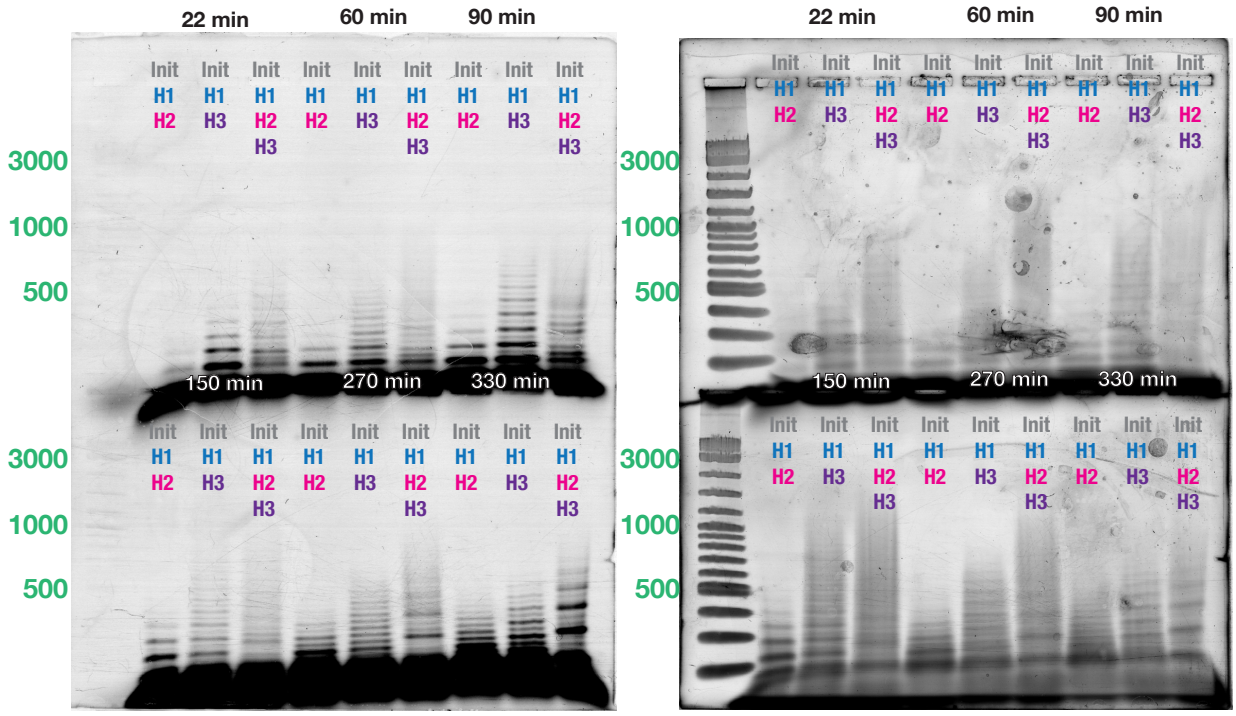


Figure B.4: The time lapse reaction of the experimental system with [Initiator] equivalent to hairpin concentrations. The gel on the left and right are the same but imaged under different conditions (left no stain, imaged at fluorophore emission wavelength; right same gel stained with SYBR Gold and imaged at SYBR Gold emission wavelength). If the polymers are randomly joining we would see an upward shift in the gel bands over time. This data shows that there is minimal joining.

B.1.4 Kinetics

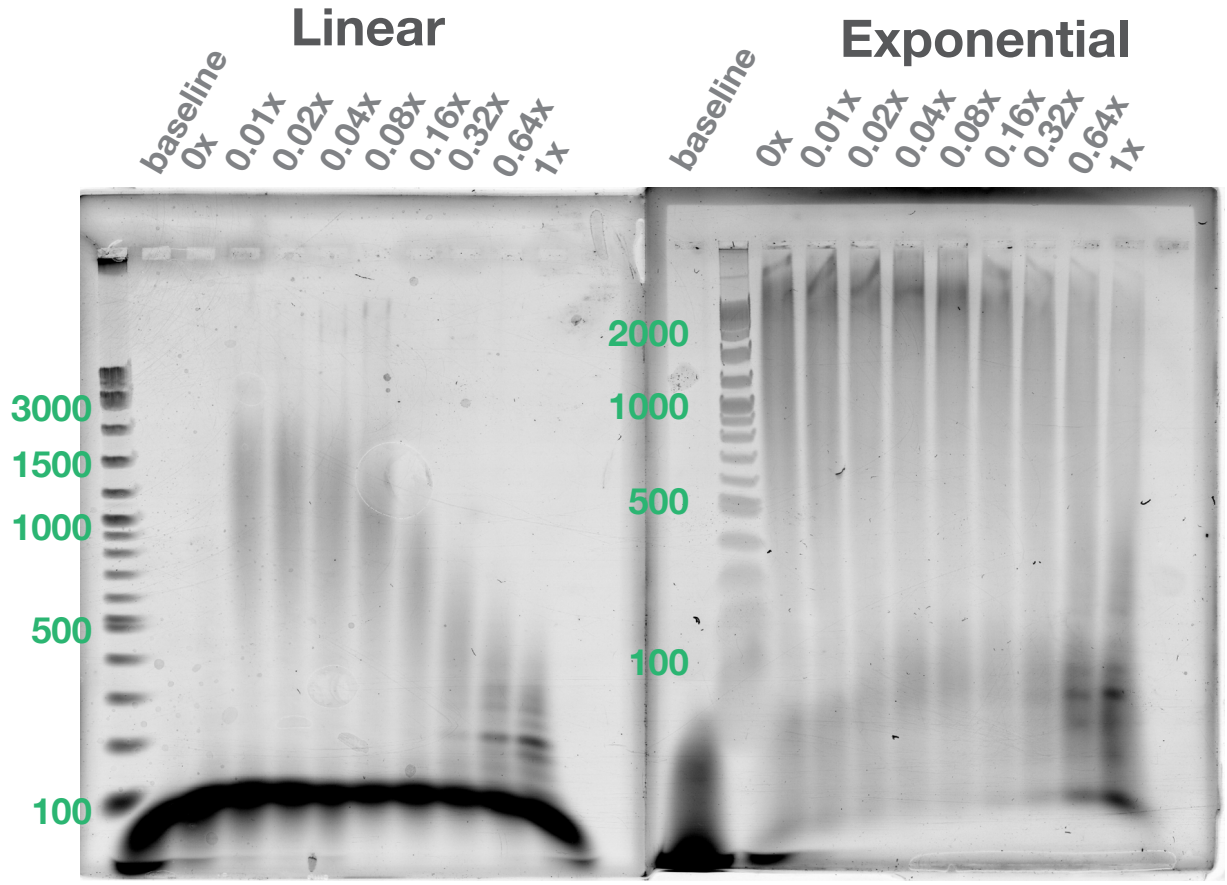


Figure B.5: The Spectrofluorimetry linear and exponential final values gels from Figure 4.9 are shown here after being stained with SYBR Gold. Post-staining makes the DNA ladder visible, allowing for the proper size classification of the polymers.

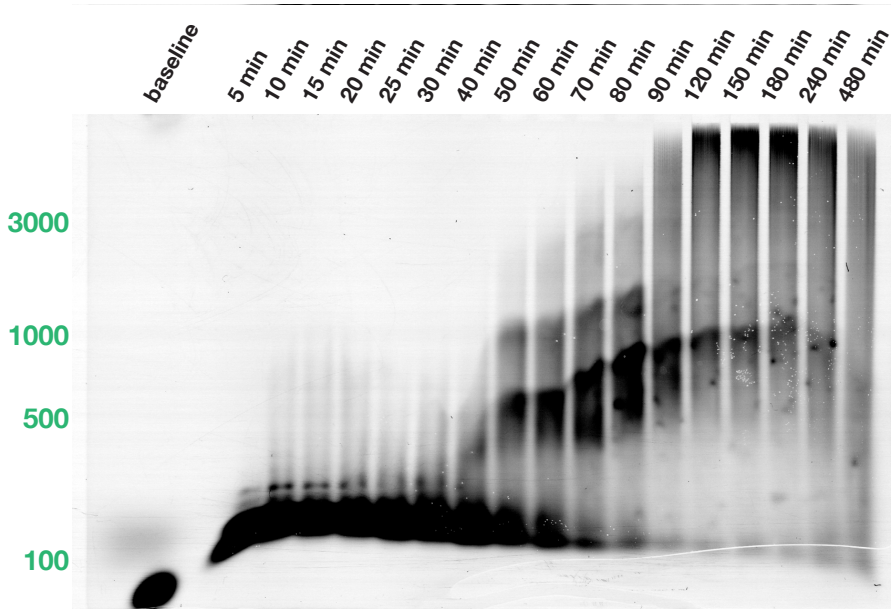


Figure B.6: Gel time-lapse studies of exponential polymer growth. Super Fine Resolution Agarose non-denaturing gels of the product of a polymerization reaction with 80 nM ROX-labeled Initiator, 1.5 μM Hairpin 1, and 1 μM of Hairpin 2 and Hairpin 3. ROX fluorescence was imaged prior to staining with SYBR Gold. Two additional experimental runs of this experiment can be found in Figures 4.13 and B.8.

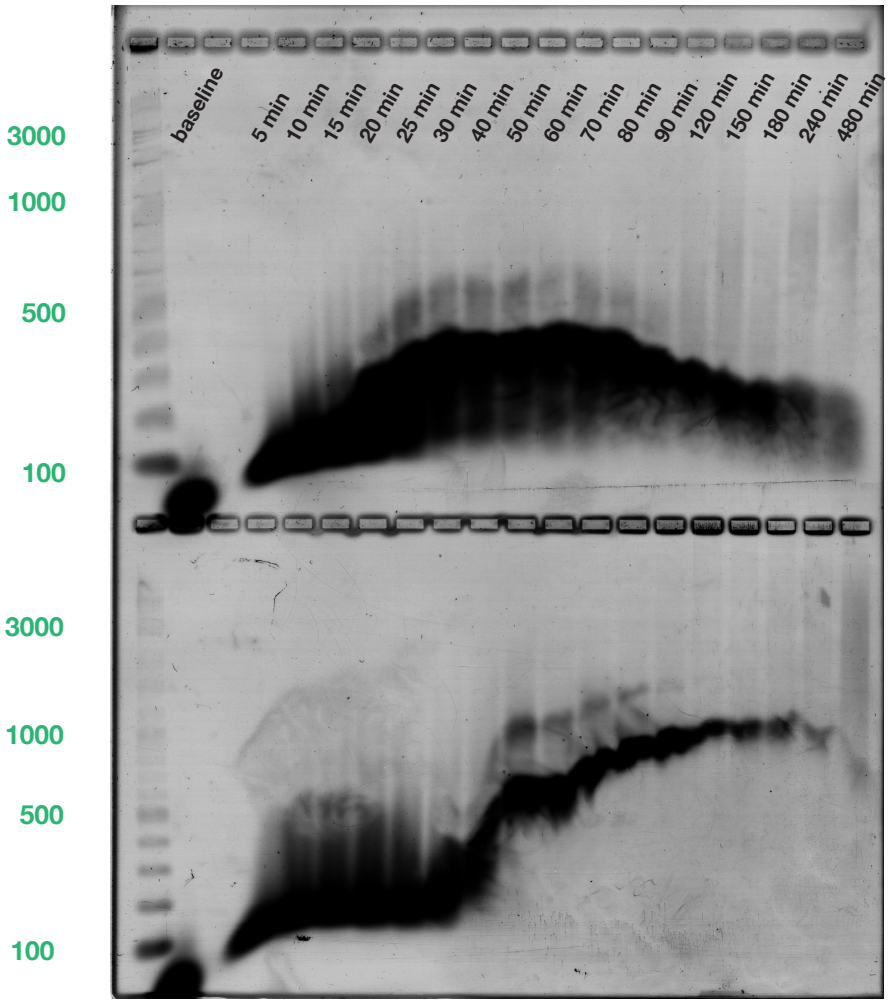


Figure B.7: The time lapse gels from Figure 4.12 (top) and Figure B.6 (bottom) are shown here after being stained with SYBR Gold. Post-staining makes the DNA ladder visible, allowing for the proper size classification of the polymers.

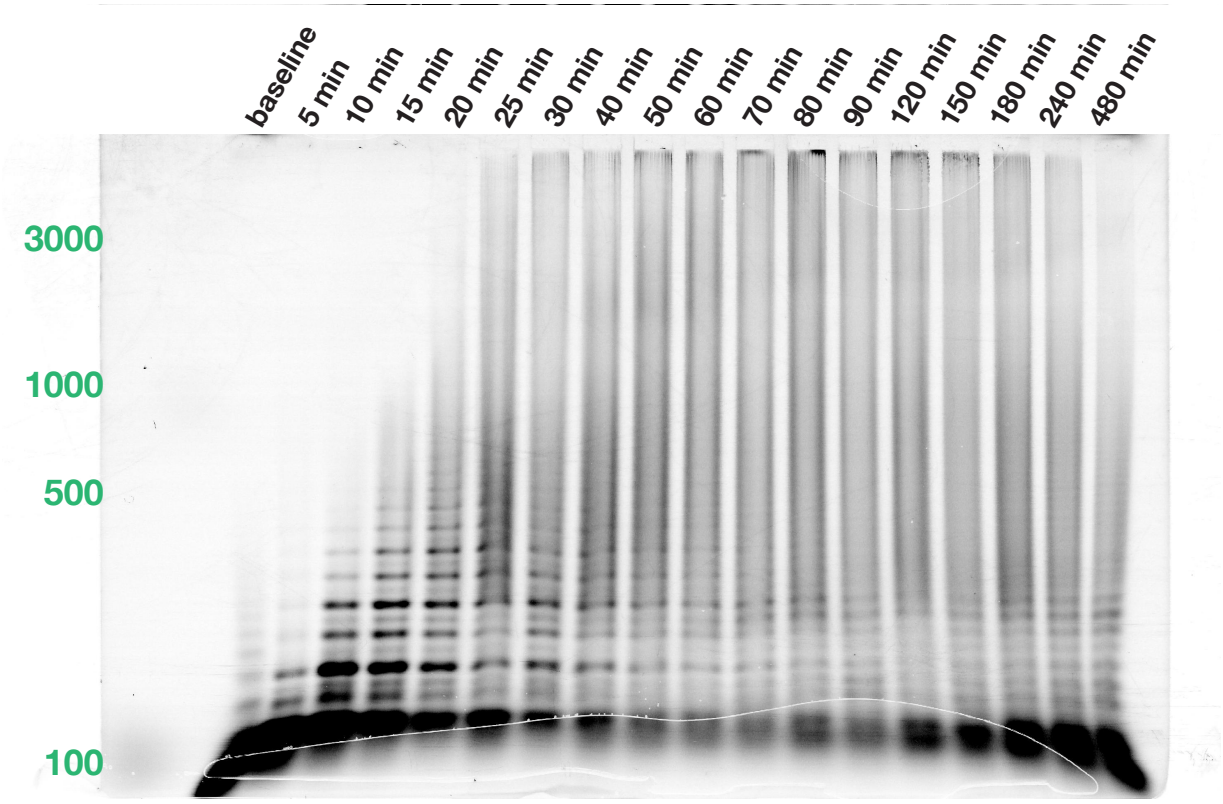


Figure B.8: Gel time-lapse studies of exponential polymer growth. Super Fine Resolution Agarose non-denaturing gels of the product of a polymerization reaction with 80 nM ROX-labeled Initiator, 1.5 μM Hairpin 1, and 1 μM of Hairpin 2 and Hairpin 3. ROX fluorescence was imaged prior to staining with SYBR Gold. Two additional experimental runs of this experiment can be found in Figures 4.13 and B.6.

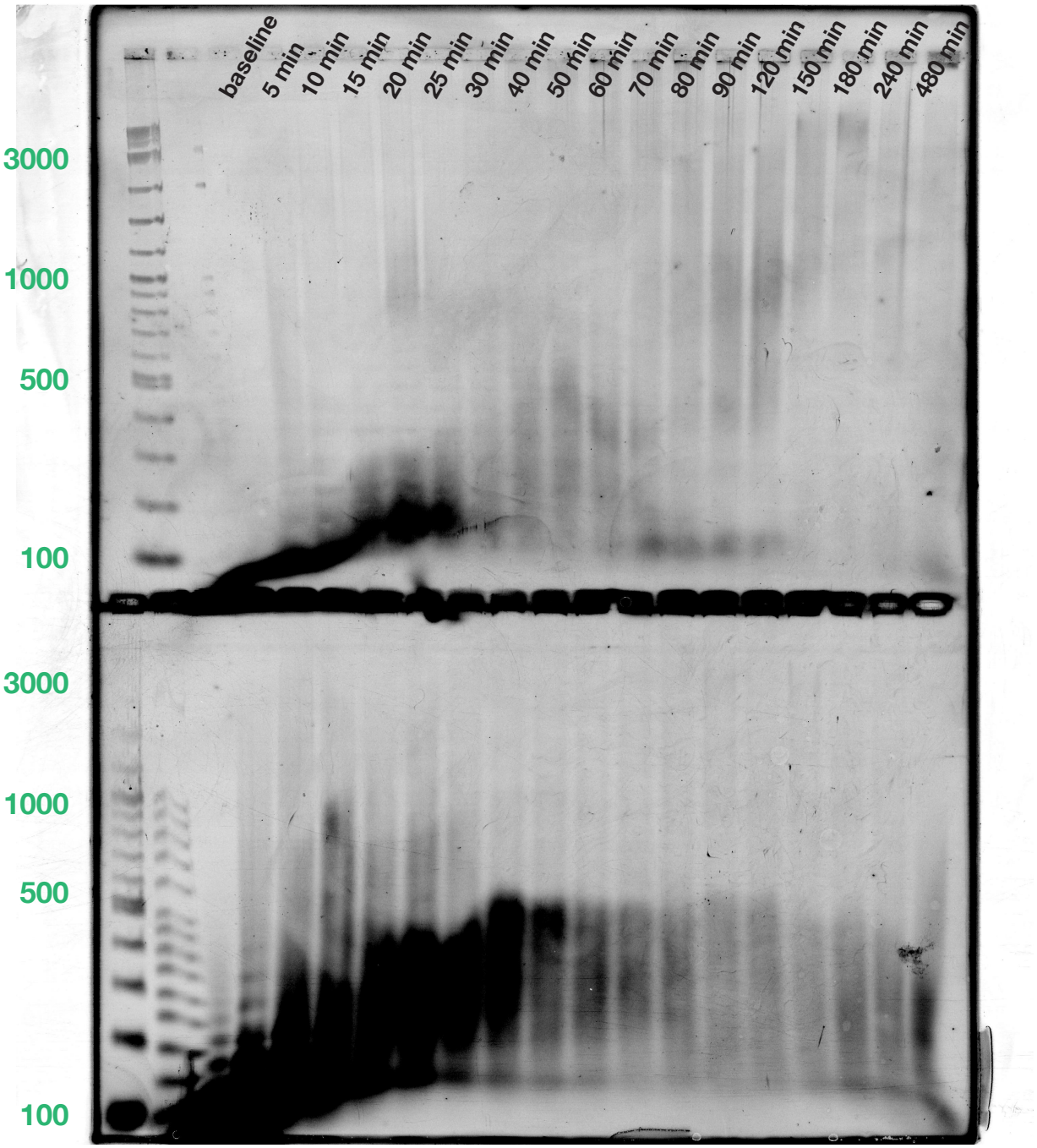


Figure B.9: The time lapse gels from Figure 4.13 (top) and Figure B.8 (bottom) are shown here after being stained with SYBR Gold. Post-staining makes the DNA ladder visible, allowing for the proper size classification of the polymers.

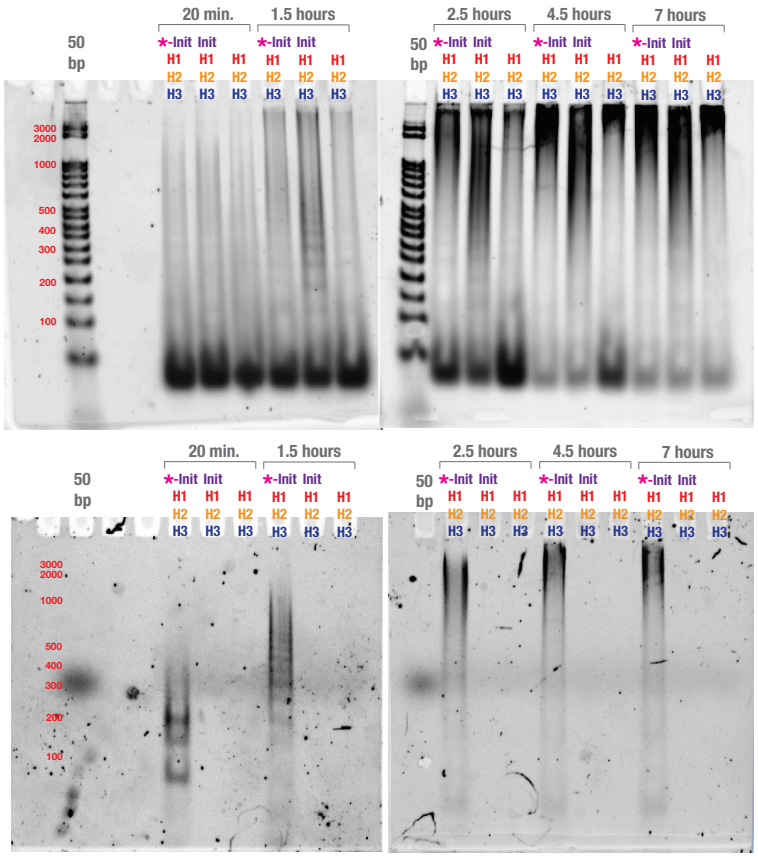


Figure B.10: Gel time-lapse studies of polymer growth. a) Polyacrylamide gels of the product of a reaction with 50 nM Initiator, 625 nM Hairpin 1, and 500nM of Hairpin2 and Hairpin 3. b) Agarose gels of the same samples.

B.2 Division Experiments

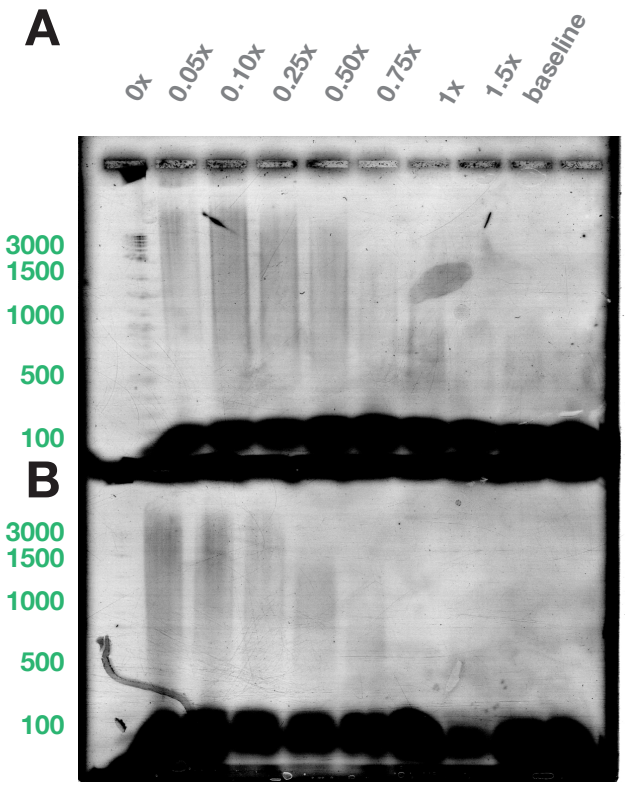


Figure B.11: The polymer division gels from Figure 4.16 are shown here after being stained with SYBR Gold. Post-staining makes the DNA ladder visible, allowing for the proper size classification of the polymers.

B.2.1 Division Kinetics

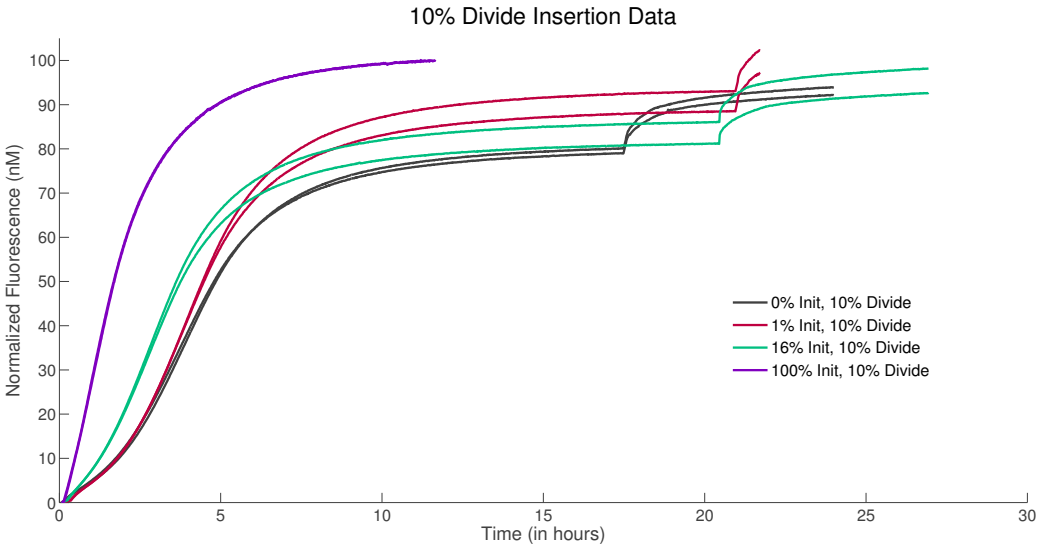


Figure B.12: Polymer division kinetics examined via fluorescence. As Hairpin 2 is incorporated into the growing and dividing polymer, the system's fluorescence increases. Plotted above are the kinetic traces of Hairpin 2 (all hairpins are present at 100 nM, the Divide complex is present at 10 nM) with varying amounts of Initiator.

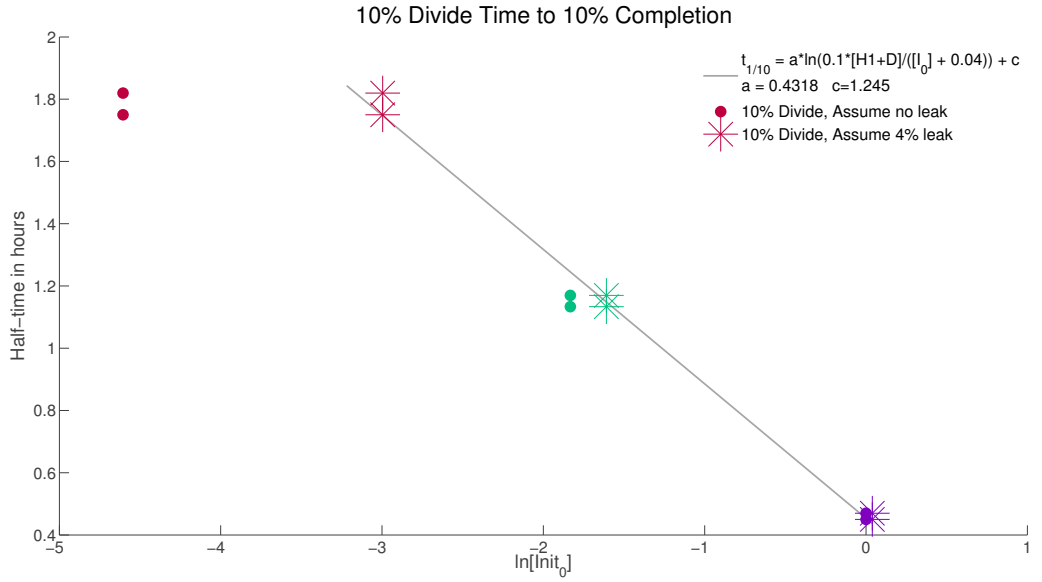


Figure B.13: Linear fit of the 10% completion time as a function of the relative concentration of Initiator to Hairpins. Filled circles correspond to a system where we assume no leak. Asterisks indicated the same points but assuming a leak equivalent to 4% of the Initiator concentration.

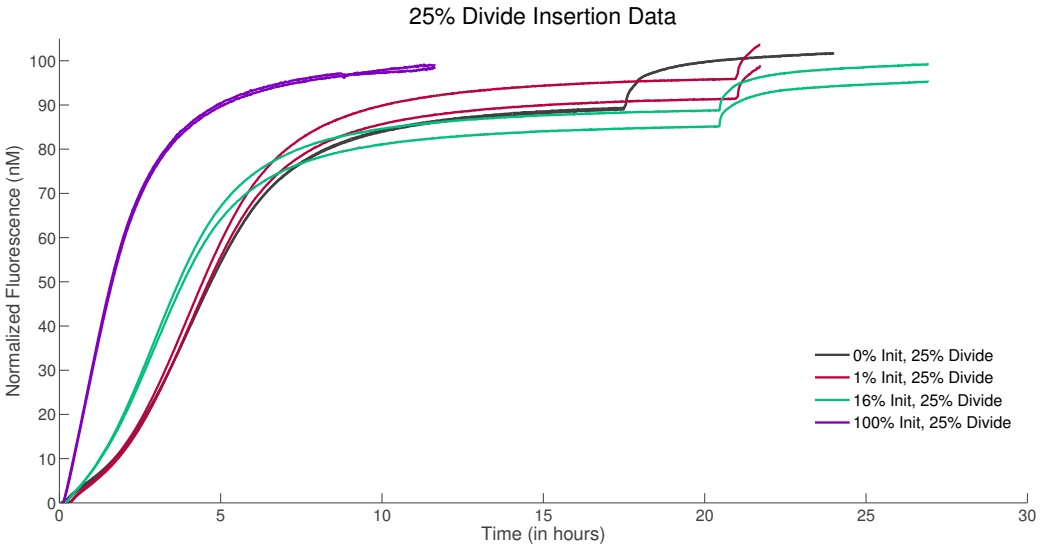


Figure B.14: Polymer division kinetics examined via fluorescence. As Hairpin 2 is incorporated into the growing and dividing polymer, the system's fluorescence increases. Plotted above are the kinetic traces of Hairpin 2 (all hairpins are present at 100 nM, the Divide complex is present at 25 nM) with varying amounts of Initiator.

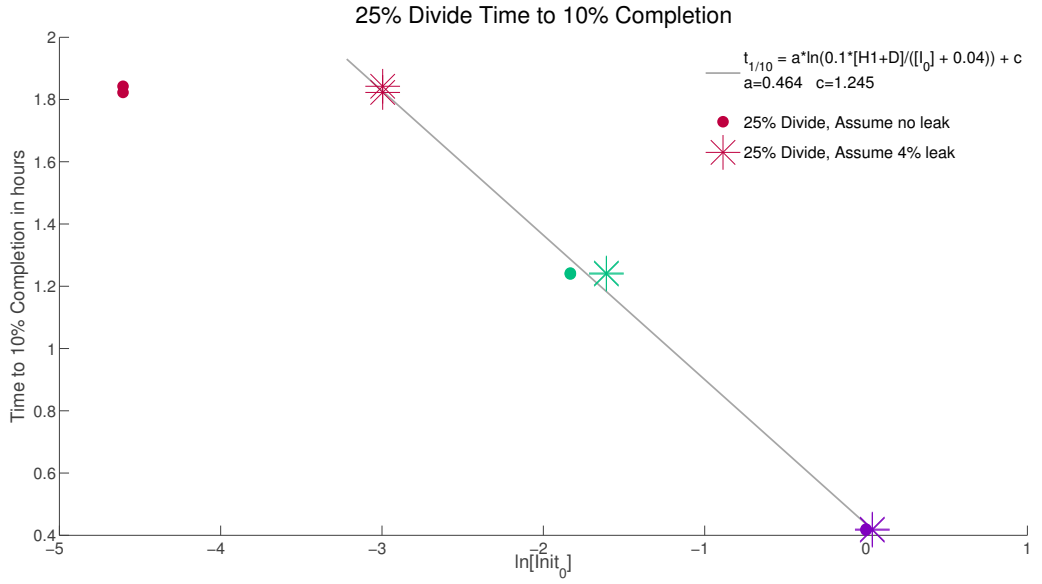


Figure B.15: Linear fit of the 10% completion time as a function of the relative concentration of Initiator to Hairpins. Filled circles correspond to a system where we assume no leak. Asterisks indicated the same points but assuming a leak equivalent to 4% of the Initiator concentration.

Appendix C

Supplemental: The Kinetics of Toehold-Mediated Four-way Branch Migration[0]

C.1 Experimental Details

Sequences for experiments discussed in the main text can be found in Table C.1. Colors correspond to the domains found in Figure 5.1. At least three traces for each toehold combination were analyzed (Figures C.1, C.2, C.3, C.4, C.5, C.6, C.7). Each toehold combination was investigated with at least two different concentration sets. The solid lines show fluorescence data, dotted lines show simulated fits for k_1 or k_2, and the gray dotted line shows the average fit simulation at each concentration (fitting method is explained below). Concentrations investigated can be found in Tables C.4 and C.3.

[0]This work was coauthored by Nadine Dabby, Ho-Lin Chen, Joseph Schaeffer, & Erik Winfree* and is currently in submission [Dabby et al., 2013] with the following contributions: all experiments were performed by N. D.; trajectory simulations were performed by J.S.; analysis was performed by N. D., H-L.C., J.S with supervision by E.W. Experimental design and manuscript was done with input from all authors.

Table C.1: Experiment sequences

reporter1-m16	\5IAbRQ\-ACCGCACGTCCACGGTGTCGC-ACCCACTCCTTCTCAA
reporter1-m6	\5IAbRQ\-ACCGCACGTCCACGGTGTCGC-ACCCAC
reporter2	GCTAAC-GCGACACCGTGGACGTGCGGT-\3Rox\
complex-m16	TTGAGAAGGAGTGGGT-GCGACACCGTGGACGTGCGGT
complex-m6	GTGGGT-GCGACACCGTGGACGTGCGGT
complex-m4	GGGT-GCGACACCGTGGACGTGCGGT
complex-m2	GT-GCGACACCGTGGACGTGCGGT
complex-m0	GCGACACCGTGGACGTGCGGT
complex-n6	ACCGCACGTCCACGGTGTCGC-GTTAGC
complex-n4	ACCGCACGTCCACGGTGTCGC-GTTA
complex-n2	ACCGCACGTCCACGGTGTCGC-GT
complex-n0	ACCGCACGTCCACGGTGTCGC
m-displace	GTGGGT-GCGACACCGTGGACGTGCGGT
n-displace	ACCGCACGTCCACGGTGTCGC-GTTAGC

Fitted Four-way Unimolecular Rates		
toehold length	average (sec^{-1})	standard deviation
$m = 16, n = 0$	4.4×10^{-4}	4.5×10^{-5}
$m = 16, n = 2$	1.4×10^{-3}	2.2×10^{-4}
$m = 16, n = 4$	1.9×10^{-3}	4.5×10^{-4}
$m = 16, n = 6$	1.2×10^{-3}	4.9×10^{-4}

Table C.2: Unimolecular rates (k_2) determined by mean squared error fitting to experimental traces. We assume that the bimolecular rate is 3×10^6 mol^{-1} sec^{-1} .

Table C.3: Four-way Branch Migration Long-Toehold Experimental Concentrations

toehold length	Reporter concentration	Complex concentration
m = 16, n = 0	0.1 nM	0.5 nM
	0.25 nM	1 nM
	1 nM	5 nM
	20 nM	100 nM
m = 16, n = 2	0.1 nM	0.5 nM
	0.25 nM	1 nM
	1 nM	5 nM
	20 nM	100 nM
m = 16, n = 4	0.1 nM	0.5 nM
	0.25 nM	1 nM
	1 nM	5 nM
	20 nM	100 nM
m = 16, n = 6	0.1 nM	0.5 nM
	0.25 nM	1 nM
	1 nM	5 nM
	20 nM	100 nM

Table C.4: Four-way Branch Migration Experimental Concentrations

toehold length	Reporter concentration	Complex concentration
m = 0, n = 0	120 nM	3 μM
	250 nM	3 μM
m = 0, n = 2	120 nM	3 μM
	250 nM	3 μM
m = 0, n = 4	120 nM	3 μM
	250 nM	3 μM
m = 0, n = 6	120 nM	1 μM
	250 nM	1 μM
m = 2, n = 0	120 nM	3 μM
	250 nM	3 μM
m = 2, n = 2	120 nM	3 μM
	250 nM	3 μM
m = 2, n = 4	120 nM	2 μM
	250 nM	2 μM
m = 2, n = 6	25 nM	200 nM
	50 nM	200 nM
m = 4, n = 0	120 nM	2 μM
	250 nM	2 μM
m = 4, n = 2	120 nM	1 μM
	250 nM	1 μM
m = 4, n = 4	25 nM	200 nM
	50 nM	200 nM
m = 4, n = 6	0.5 nM	1 nM
	0.5 nM	2.5 nM
	2.5 nM	2.5 nM
m = 6, n = 0	50 nM	1.5 μM
	100 nM	1.5 μM
m = 6, n = 2	0.5 nM	5 nM
	2.5 nM	5 nM
m = 6, n = 4	0.5 nM	1 nM
	0.5 nM	2 nM
	0.5 nM	2.5 nM
	0.5 nM	5 nM
	0.5 nM	10 nM
	2.5 nM	2.5 nM
m = 6, n = 6	0.5 nM	1 nM
	0.5 nM	2.5 nM
	2.5 nM	2.5 nM

C.2 Modeling and Sample Code

In order to determine k_2 we designed reaction complexes with a 16-basepair toehold. We assumed $k_1 = k_f = 3 \times 10^6$, a value taken from [Zhang and Winfree, 2009], due to the length of the toeholds. We then fit ak_2 value to each trace in MATLAB by minimizing the mean squared error of our fit function to the data. Sample code of the main function call for trace one of the m=16, n = 2 data set follows:

```
%   k0 is an initial estimate of k1;
k0 = log(0.0004);
[estimated_k] = fminunc(@err_func_m16n2v1, k0);
```

The error function *err_func_m16n2v1* follows:

```
function err_func = err_func_m16n2v1(input)

%   x-axis = over-all time
%   m_n = spectrofluorimeter read-out
%   start_conc = limited reactant concentration
%   offset = start time
%   index_offset = index into offset time  in the x-axis matrix
%   baseline = low fluorescence level
%   max = high fluorescence level
%   end_index = end of the fit region

index_offset1 = 7;
offset1 = 300;
m_n1 = cleanm16n0246v1(:,3);
xaxis1 = cleanm16n0246v1(:,1);
baseline1 = min(m_n1);
max1 = max(m_n1);
max1b = max1;
start_conc1 = 20 * 10^(-9);
data = [xaxis1, m_n1];
end_index = size(data, 1);
k = exp(input(1));
```

```
err_func = 0;
options = odeset('MaxStep',100,'refine',1e-10,'InitialStep',100,'RelTol',1e
    -10,'AbsTol',1e-10);
datasize = size(data, 1);
t = data(index_offset1:datasize,1)-offset1;
y0 = [start_conc1*(max1b-baseline1)/(max1-baseline1), 0, 0, k];
[t, y2] = ode45(@fmin_toehold_norm_1, t, y0, options);
ye = y2(:,3);

for i = index_offset1:(end_index - index_offset1) //cutting off the displace
    strand reaction at the end
  err_func = err_func + ((ye(i-(index_offset1 -1)) - (data(i, 2) - baseline1)
      /(max1-baseline1)*start_conc1)^2)*1e17;  %min square difference of sim -
       data
end
```

After finding the mean k_2 values for all long-toehold experiments (Figure C.1), we found a disparity between the open (one toehold connecting the complexes) and closed (both toeholds connecting the complexes) cases of toehold-mediated four-way branch migration. All other traces were fit exactly as above, except that we assumed the k_2^{open} and k_2^{closed} values to be those calculated above, and we instead fit k_1.

All traces that did not reach a completion level of 20% within 24 hours were fit linearly using the polyfit function in Matlab. Sample code from one of the $(m = 0, n = 2)$ data set follows:

```
function [slope] = polyfit_m0n2v1()

%   x-axis = over-all time
%   m_n = spectrofluorimeter read-out
%   start_conc = limited reactant concentration
%   offset = start time
%   index_offset = index into offset time  in the x-axis matrix
%   baseline = low fluorescence level
%   max = high fluorescence level
```

```
%   end_index = end of the fit region

end_index = 18*60;
index_offset1 = 5*60;
start_conc1 = 120 * 10^(-9);
m_n1 = cleanm0n0246(:, 3);
xaxis1 = cleanm0n0246(:, 1);
baseline1 = min(m_n1);
max1 = max(m_n1);
max1b = max1;

[slope] = polyfit(xaxis1(index_offset1: end_index), m_n1(index_offset1:
    end_index), 1)
```

Here, k_1 equals the estimated slope returned by the function.

Finally, after fitting the reaction rates, we fit the data to our model by adding a $\Delta G_{k_2}^{fit}$ parameter for both the open and the closed toehold-mediated four-way branch migration cases. We found this parameter by using a minimum least squares error fit between the experimentally-fit bimolecular rates k_1^{fit} and those calculated by our model, k_1^{calc}. The main function call for the closed case is as follows:

```
ddG = 5; %estimated parameter value
[estimated_ddG] = fminunc(@forwardclosedNupackSome, ddG);
```

The error function *forwardclosedNupackSome* follows:

```
function fvalue = forwardclosedNupackSome(ddG_closed)

%   k2_closed =  average fit reaction rate
%   N = number of steps in the branch
%   k_f = forward reaction rate
%   Nupack_closed = Nupack Calculated Delta G values
%   exp_k = reaction rates extracted from experimental data
%   k1 =  reaction rate calculated by Nupack energy values adjusted by adding
    our paramter (ddG)

k2_closed= 1.5*10^-3
N = 21
k_f = 3*10^6

Nupack_closed = [-5.6000, -7.2100, -11.0300,  -10.1900, -11.8000, -15.6200,
    -12.4100, -14.0200, -17.8400]
exp_k = [0.1015, 0.9329, 491.5167, 56.1973, 765.72, 277490, 9403.4,  70098,
    689780]
exp_k = log10(exp_k)

for i = 1: 9
   k1(i) = (k_f)*(k2_closed/( ((k_f)*(exp((Nupack_closed(i) + ddG)/0.593)))/N
       + k2_closed))
end

k1 = log10(k1);
fvalue = 0

for i = 1:9
    fvalue = fvalue + (k1(i) - exp_k(i))^2     %min sum of square difference
        between simulation and data
    ddG          % outputs fitted parameter at each round
end
```

C.3 Experimental Data

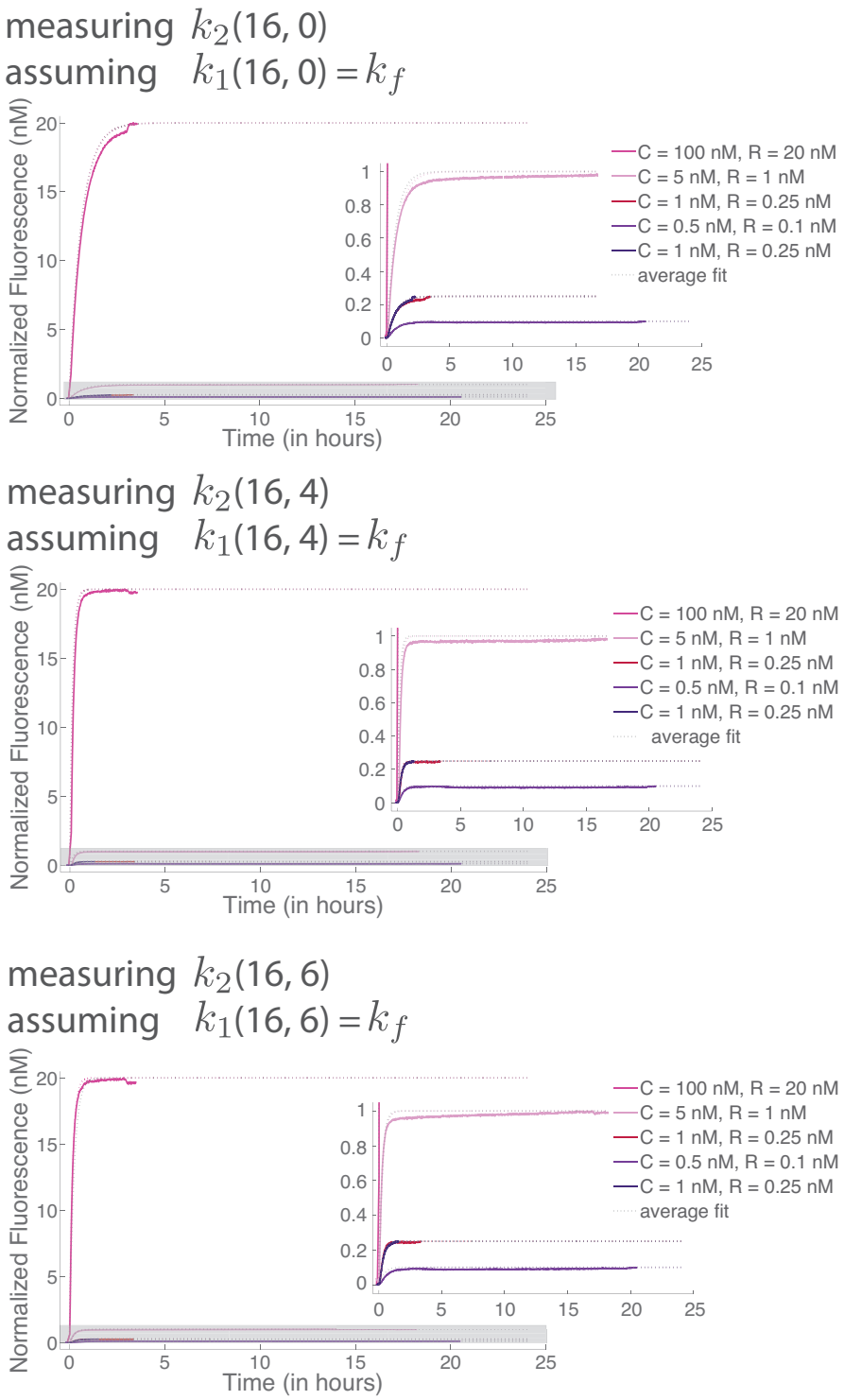


Figure C.1: Long toehold traces for fitting k_2.

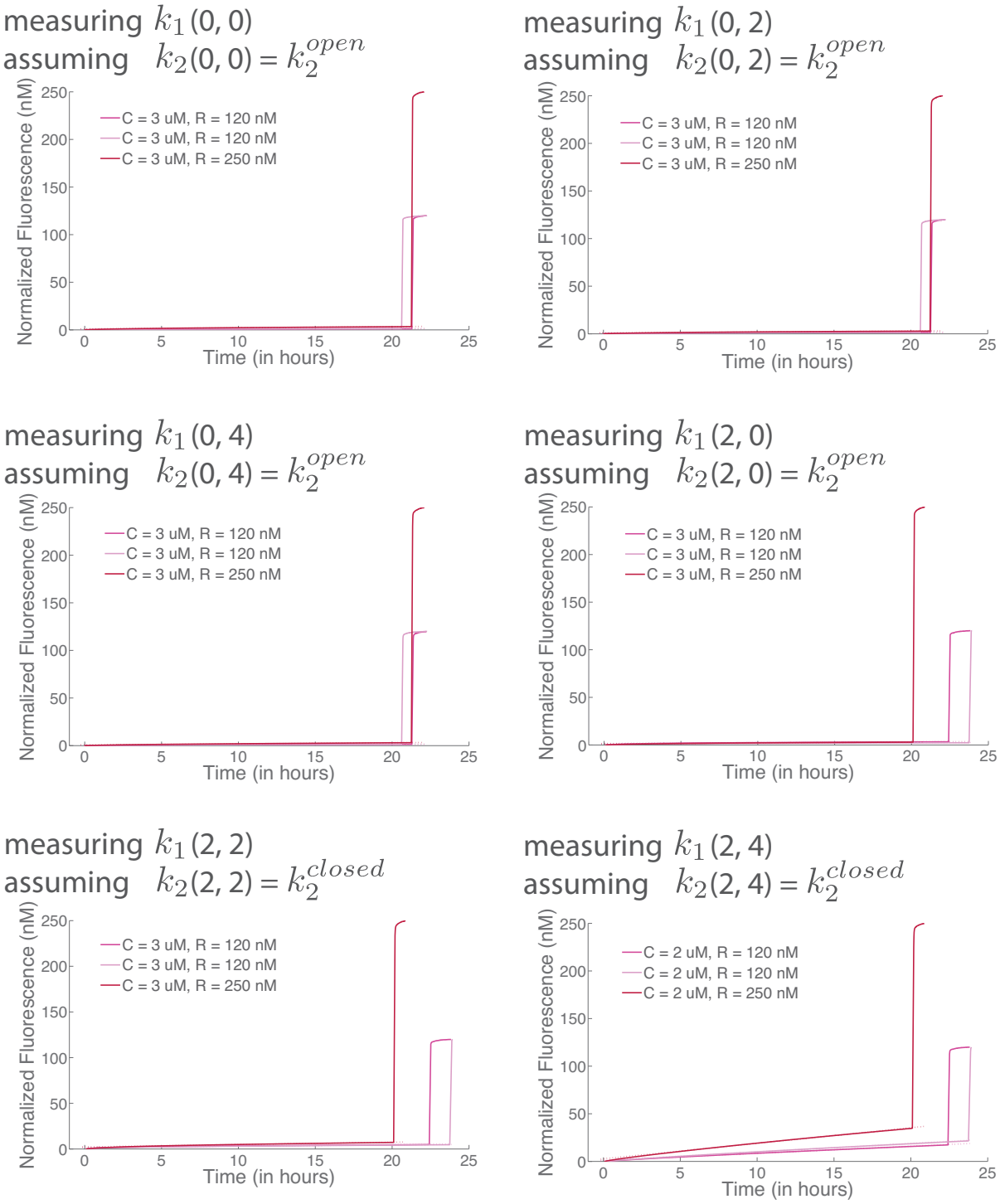


Figure C.2: Slow traces utilized a mean squared error linear fit to find k_1.

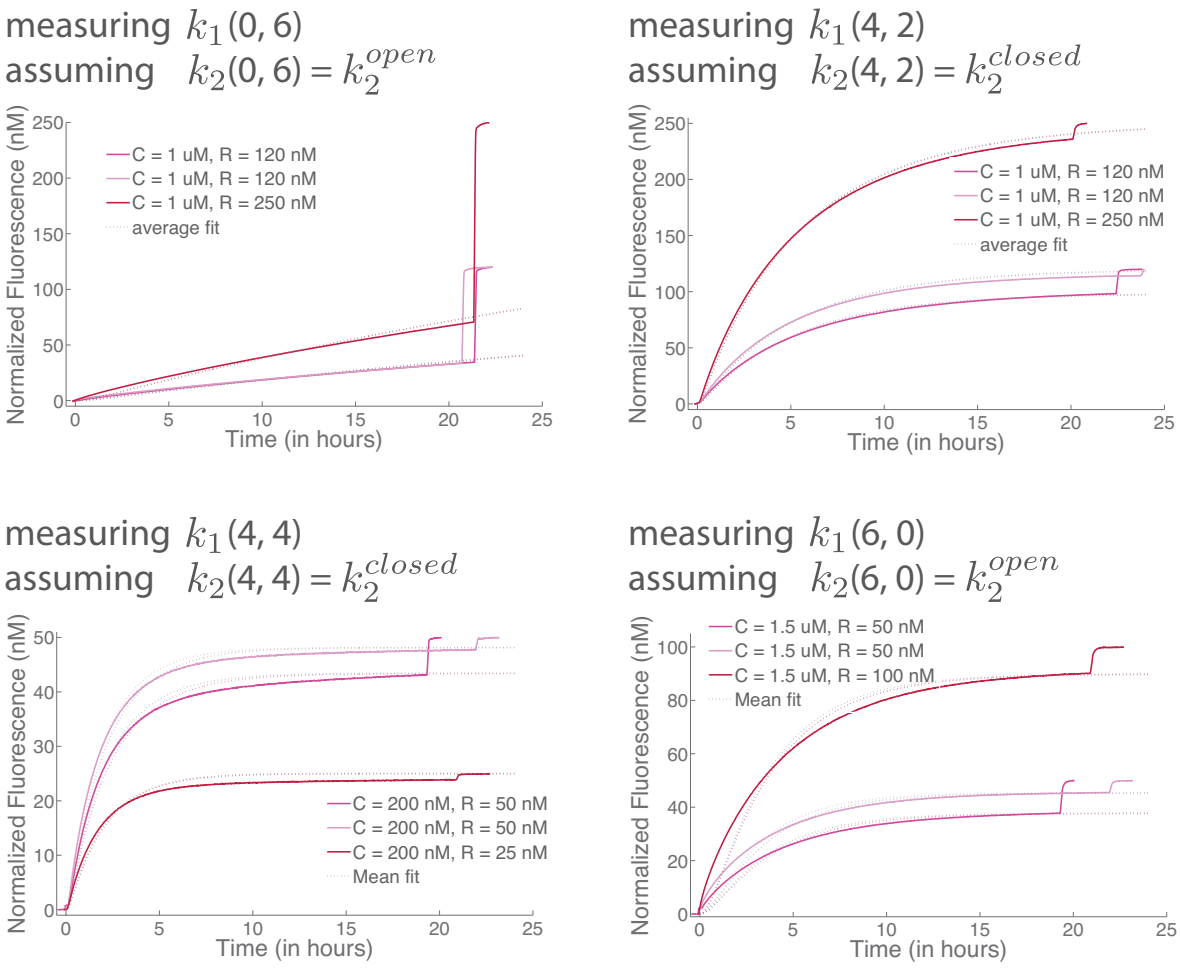


Figure C.3: Medium-speed traces utilized a mean squared error (fminunc) function to fit k_1.

C.3.1 Completion Levels

We noted a lack of completion in our faster reactions with very low concentrations. After tracing each batch to purification date, we noticed a general trend that the older the reaction complexes, the lower the completion level (see Figures C.4, C.5, C.6, C.7). Across all batches, the variance of fitted rates is not correlated with completion level – whether the reporter was six months old or a month old, we measure the same spread of rates. Thus we are confident that our decision to fit the traces to the lowered completion level for these traces should affect the results within the standard deviation reported.

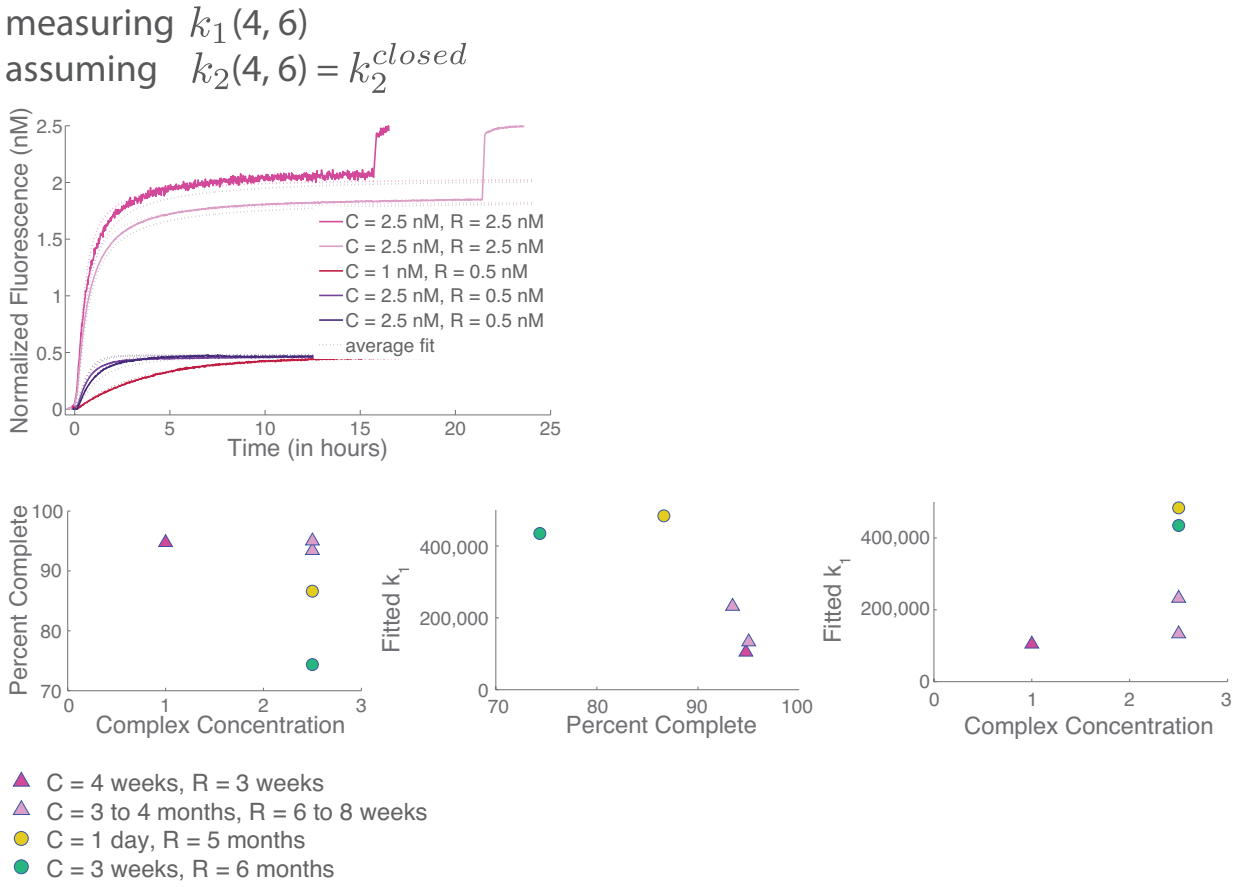


Figure C.4: We utilized a mean squared error (fminunc) function to fit k_1 for the $(m = 4, n = 6)$ reaction. Completion levels were assessed by batch below.

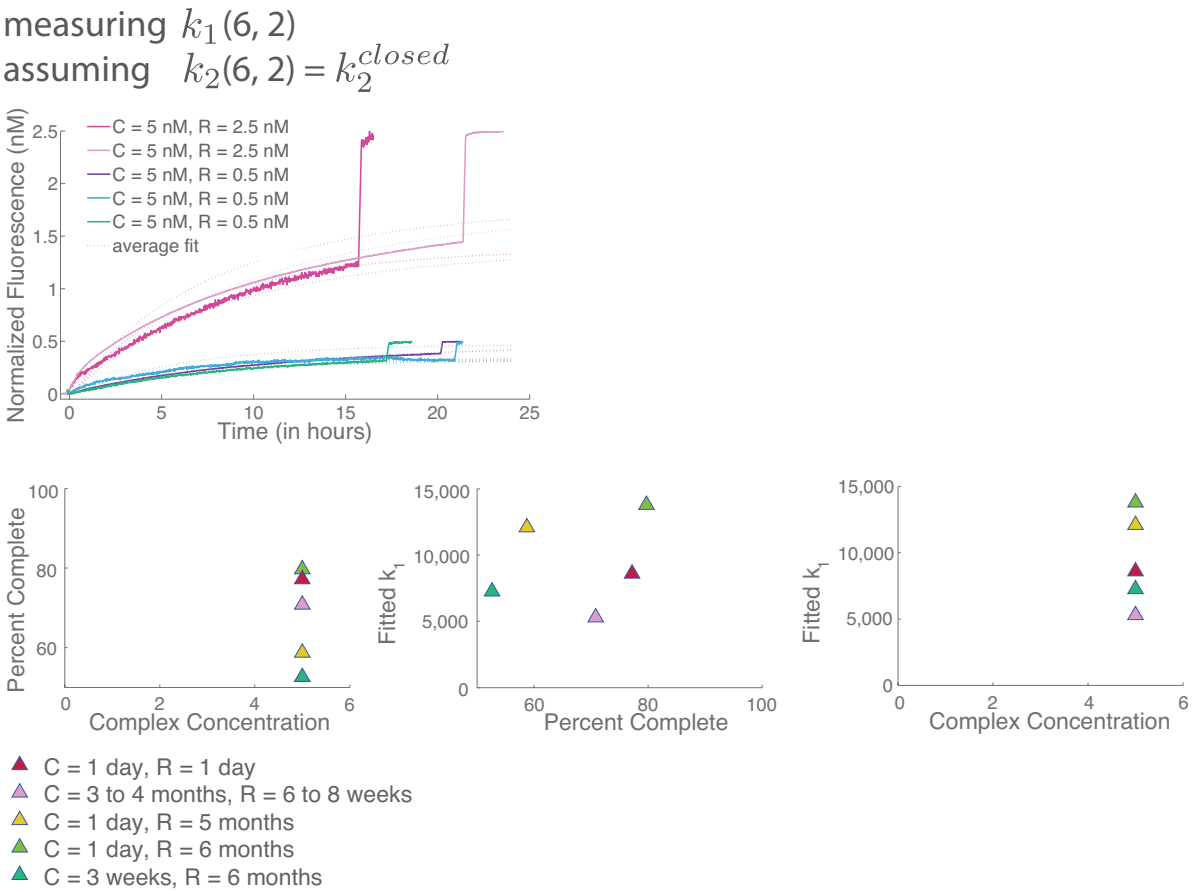


Figure C.5: We utilized a mean squared error (fminunc) function to fit k_1 for the $(m = 6, n = 2)$ reaction. Completion levels were assessed by batch below.

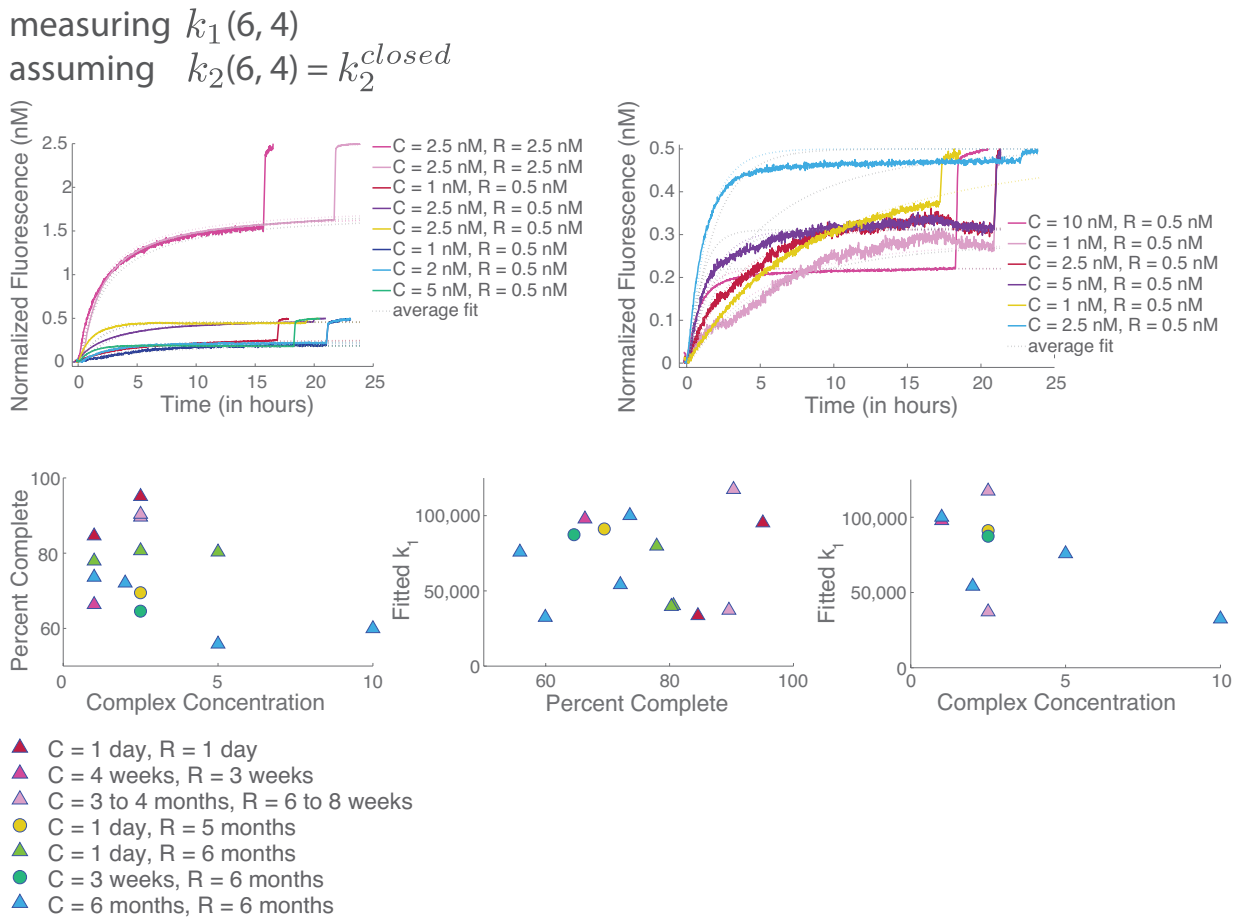


Figure C.6: We utilized a mean squared error (fminunc) function to fit k_1 for the $(m = 6, n = 4)$ reaction. Completion levels were assessed by batch below.

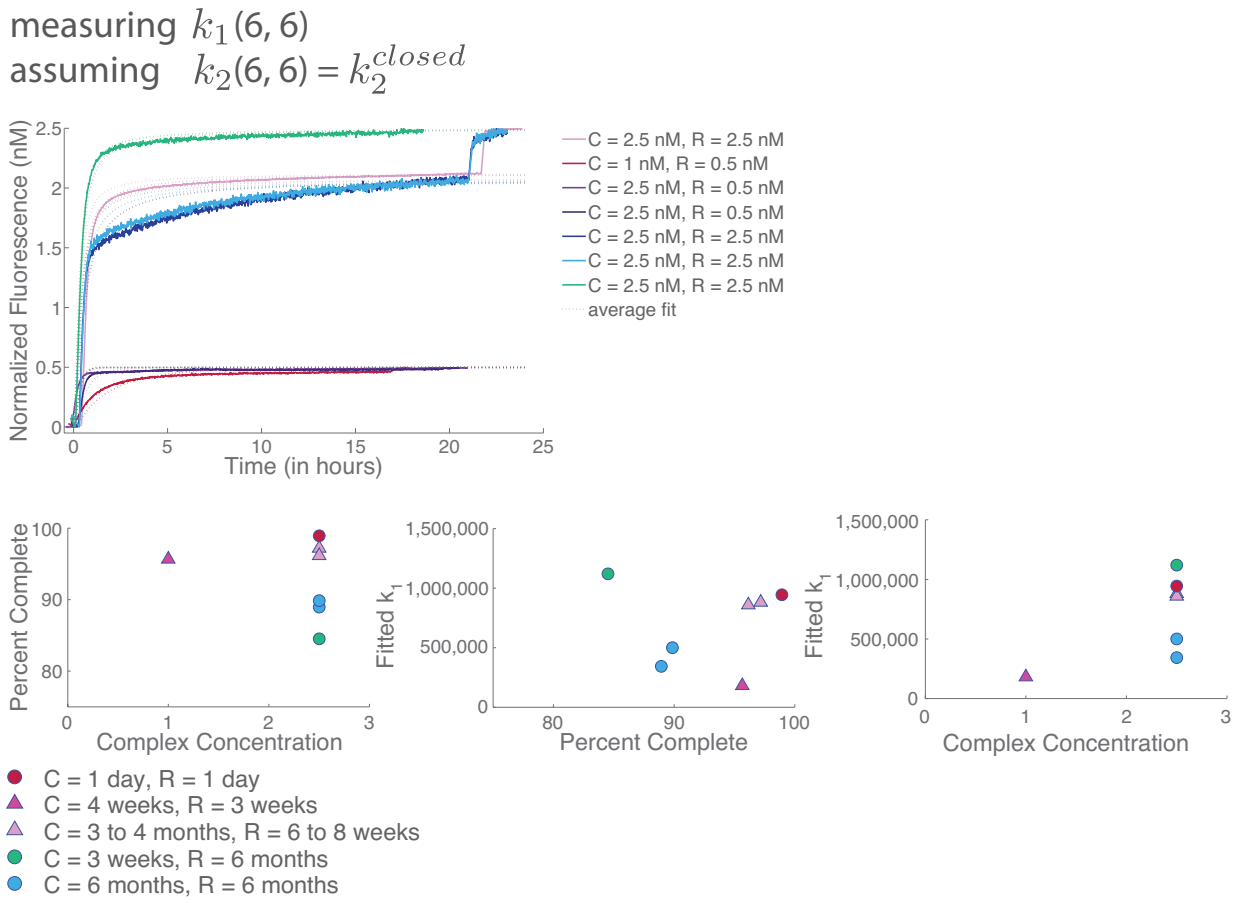


Figure C.7: We utilized a mean squared error (fminunc) function to fit k_1 for the $(m = 6, n = 6)$ reaction. Completion levels were assessed by batch below.

C.3.2 Displacement Strand Controls

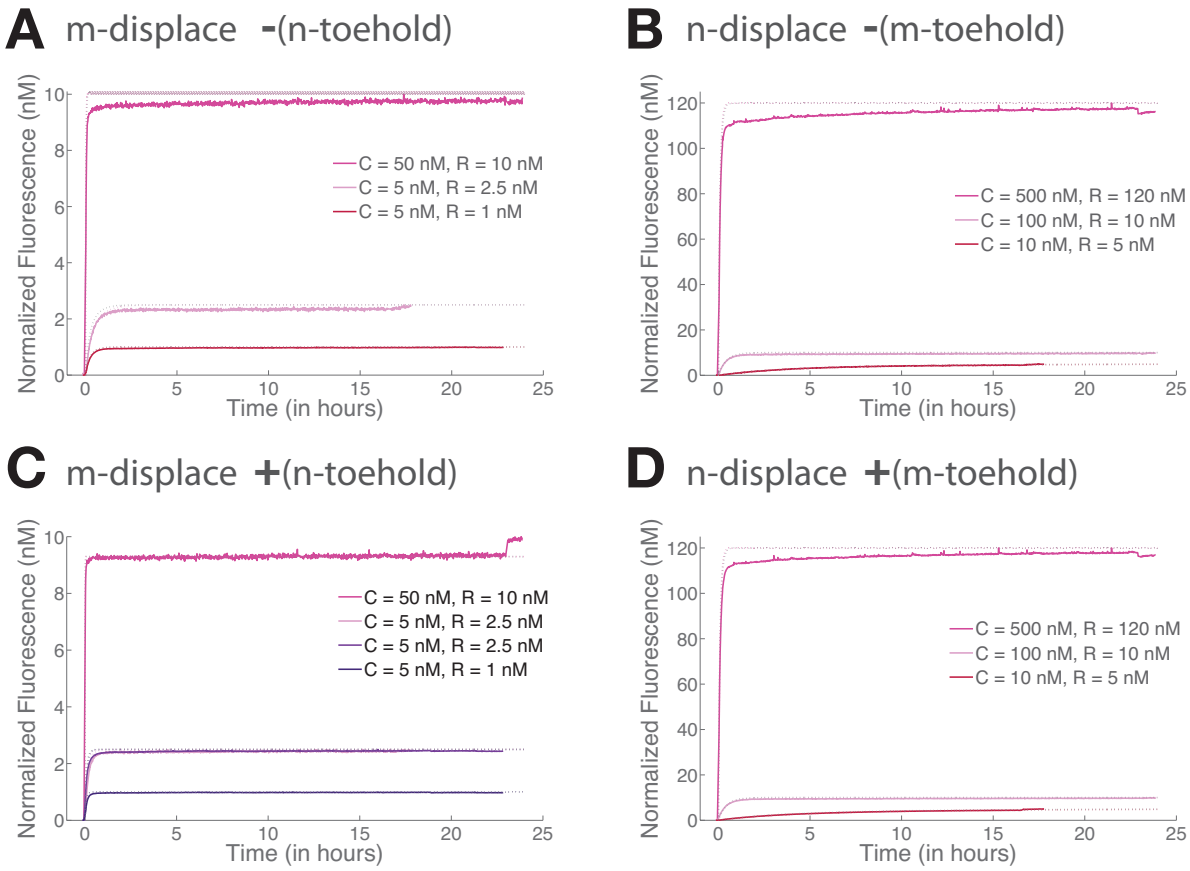


Figure C.8: (A) Reaction of m-displacement strand with a reporter complex missing the n toehold. (B) Reaction of n-displacement strand with reporter complex missing the m toehold. (C) Reaction of m-displacement strand with a reporter complex (n toehold present). (D) Reaction of n-displacement strand with a reporter complex (m toehold present).

Table C.5: Four-way Branch Migration Displacement Strand Control Concentrations

displacement strand	reporter m-toehold	reporter n-toehold	reporter concentration	displace concentration
m-displace	+	-	1 nM	5 nM
			2.5 nM	5 nM
			10 nM	50 nM
	+	+	1 nM	5 nM
			2.5 nM	5 nM
			10 nM	50 nM
n-displace	-	+	5 nM	10 nM
			10 nM	100 nM
			120 nM	500 nM
	+	+	5 nM	10 nM
			10 nM	100 nM
			120 nM	500 nM

C.3.3 Reverse Experiments

When additional toeholds are added to the opposite side of the complex and reporter, in addition to the m and n toeholds, a reverse reaction is possible (Figure C.9). This rate will depend on the length of these toeholds in addition to the length of m and n. We gauge the significance of a reverse reaction in the case where the m-product and n-product have one complementary toehold, y, of three bases in length (Figure C.9), using the concentrations found in Table C.7. In this case the reactions most likely to go backward are $(m = 0, n = 0)$, $(m = 0, n = 2)$, $(m = 2, n = 0)$.

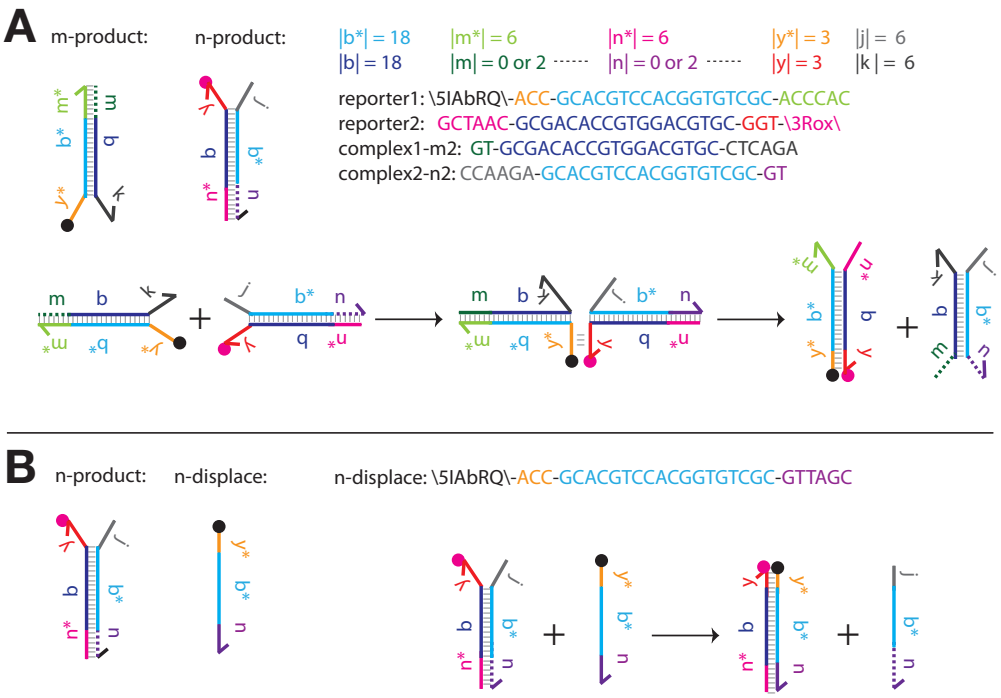


Figure C.9: (A) Experimental set-up to measure reverse reaction rates: We annealed and purified the product complexes (m-product and n-product) with a y and y' toehold domain of three base pairs and an extra toehold that is not complementary to anything in solution. This reverse reaction would be equivalent to a forward reaction with toeholds of length $(m = 0, n = 3)$. If the two complexes react and complete strand exchange, regenerating the original Reporter and Complex, we expect the overall fluorescence in solution to decrease. Sequences for DNA strands are color-coded by domain, \5IAbRQ\ indicates a 5' Iowa Black red quencher modification, and \3Rox\ indicates a 3' ROX fluorophore modification. (B) At the end of the experiment another strand of DNA is added into the solution in order to fully displace all unreacted complexes. In contrast to the experiments discussed in the main paper this displacement strand has a 5' quencher modification, thus when added in excess the fluorescence levels quickly decrease. As above, sequences are color-coded by domain.

Table C.7: Reverse Four-way Branch Migration Experimental Concentrations

toehold length	m-product concentration	n-product concentration
m = 0, n = 0	1 μM	500 nM
m = 0, n = 2	1 μM	500 nM
m = 2, n = 0	1 μM	500 nM

Table C.6: Reverse experiment sequences

reporter1	\5IAbRQ\-ACC-GCACGTCCACGGTGTCGC-ACCCAC
reporter2	GCTAAC-GCGACACCGTGGACGTGC-GGT-\3Rox\
complex-m2	GT-GCGACACCGTGGACGTGC- CTCAGA
complex-m0	GCGACACCGTGGACGTGC- CTCAGA
complex-n2	CCAAGA- GCACGTCCACGGTGTCGC-GT
complex-n0	CCAAGA- GCACGTCCACGGTGTCGC
n-displace	\5IAbRQ\-ACC-GCACGTCCACGGTGTCGC-GTTAGC

In these experiments the two product complexes (m-product and n-product) were annealed, each complex had a three-base-toehold on the opposite end of the complex from the m and n toehold sites. Because the product complexes separate the fluorophore quencher pair, the progress of this reaction is observed in reverse: we expect that if the reverse reaction occurs it will result in the reactants of our original experiments: the Reporter and Complex. In the newly formed Reporter, the fluorophore and quencher will be paired again resulting in an over-all decrease in fluorescence. At the end of the experiment another strand of DNA is added into the solution in order to fully displace all unreacted complexes. In contrast to the experiments discussed in the main paper this displacement strand has a 5' quencher modification, thus when added in excess the fluorescence levels quickly decrease (Figure C.9B).

The reverse reactions in our smallest toehold cases were negligible (Table C.8 and Figure C.10). Since these were the most energetically favorable of all of the reverse reactions, we are confident that the effect of a reverse reaction on our kinetics experiments is negligible.

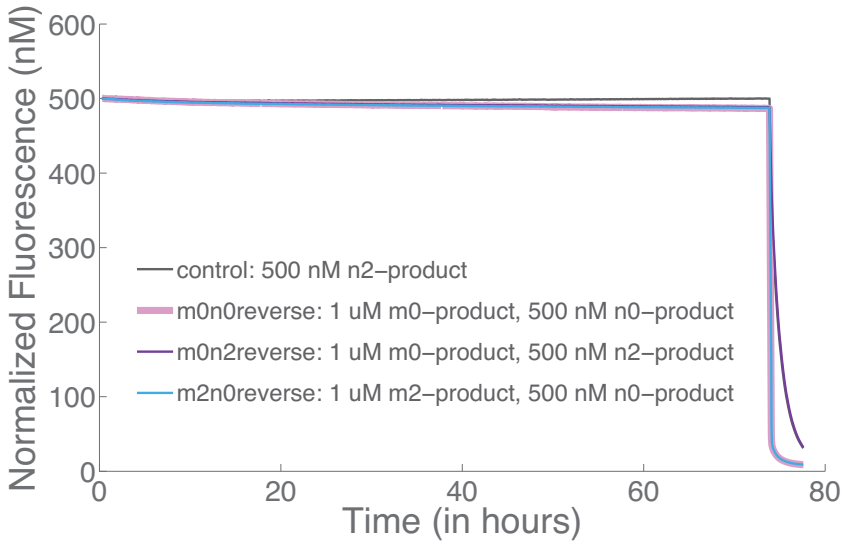


Figure C.10: Reversible four-way branch migration traces to fit the value of $k_r(0,3)$.

Fitted k_r Rates for reversible four-way branch migration

toehold length	average $k_r(0,3)$ (sec^{-1})
control	≈ 0
m0n0	0.11
m0n2	0.096
m2n0	0.093

Table C.8: Reverse reaction rate determined by linear fitting to experimental traces of reversible four-way branch migration. The control (500 nM n-product solution) showed a negligible change in fluorescence over three days resulting in an observed rate of: $k_r^{fit}(0,3) = -0.025\ (sec^{-1})$, which we interpret as zero within experimental error.

C.4 Reaction Coordinate

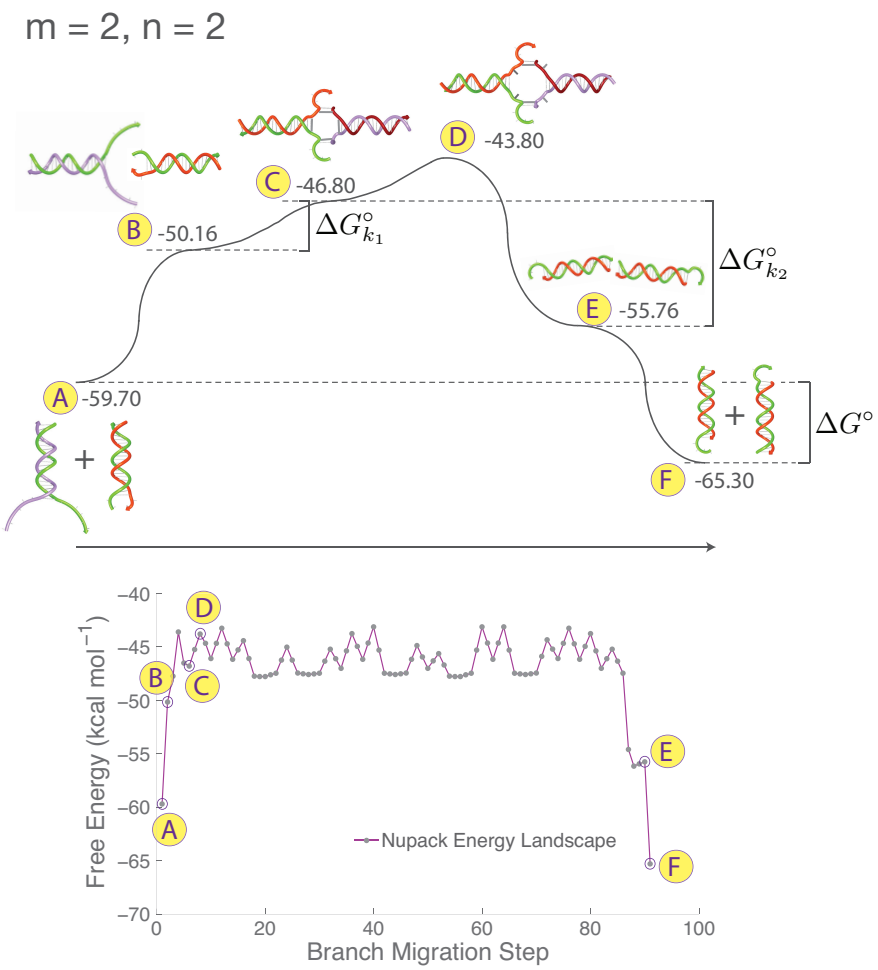


Figure C.11: A reaction coordinate for the open loop case where $(m = 2, n = 2)$. It is accompanied by a Nupack simulation of one trajectory of the thermodynamic energy landscape; each point indicates the formation or cleavage of a single hydrogen bond. Labels correspond to the indicated state in reaction coordinate. Equations are listed in Figure 5.4.

C.5 Trajectory Simulations

Simulated reaction rates (k_1^{sim}) were calculated using Multistrand [Schaeffer, 2012], an analysis tool that simulates the kinetics of multistranded DNA systems with single-basepair resolution utilizing the NUPACK energetics model. Sample size indicates the number of trajectories simulated.

Final values were normalized by computing a scaling factor by minimizing the mean multiplicative factor that best fit the raw multistrand results to the experimental results, which in this case was a factor of 20. Values that are upper-bounded indicate reactions that did not have a single forward result.

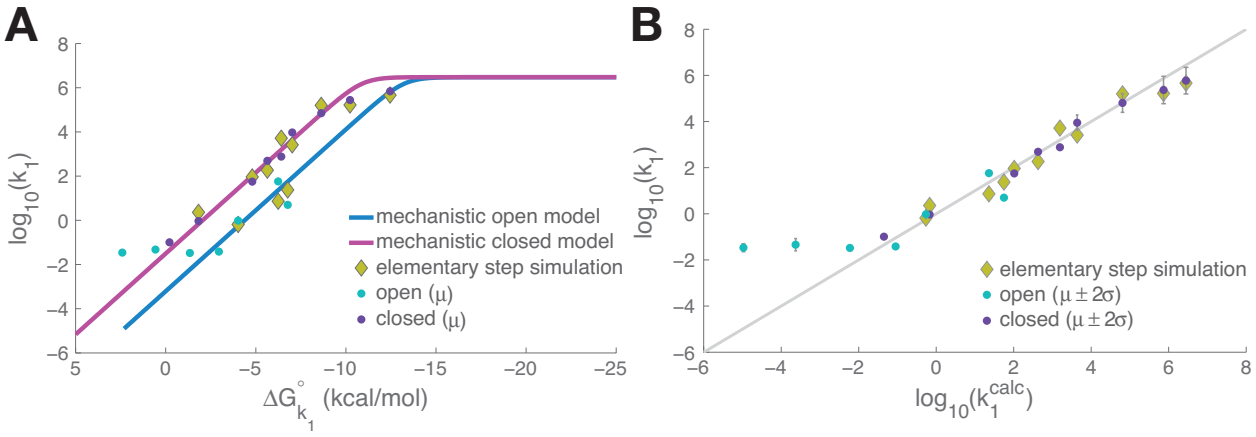


Figure C.12: Multistrand-determined reaction rates k_1^{sim} (yellow diamonds) are plotted against, experimentally fit and calculated rates. (A) Plot of $\Delta G^{\circ}_{k_1}$ versus $log_{10}(k_1)$compares our models for open (blue) and closed (magenta) loop toehold-mediated four-way branch migration to our experimentally fit mean k_1 rates; dots correspond to open (blue) and closed (purple) loop reactions. (B) Plot compares the $log_{10}(k_1)$ rate calculated by our model to the experimentally fit mean rates. Error bars show two standard deviations of error in our experimental measurements.

Bibliography

[Adleman, 1994] Adleman, L. (1994). "Molecular Computation of Solutions to Combinatorial Problems". *Science*, 266(5187):1021–1024.

[Adleman et al., 2001] Adleman, L., Cheng, Q., Goel, A., and Huang, M. (2001). "Running Time and Program Size for Self-Assembled Squares". In *Proceedings of the thirty-third annual ACM Symposium on Theory of Computing*, pages 740–748. ACM.

[Aldaye et al., 2008] Aldaye, F., Palmer, A., and Sleiman, H. (2008). "Assembling Materials with DNA as the Guide". *Science*, 321(5897):1795–1799.

[Alighieri, 2013] Alighieri, D. (2013). *The Divine Comedy: Inferno, Purgatorio, Paradiso (Penguin Classics Deluxe Edition)*. Penguin Classic.

[Alvarado and Tsonis, 2006] Alvarado, A. and Tsonis, P. (2006). "Bridging the Regeneration Gap: Genetic Insights from Diverse Animal Models". *Nature Reviews Genetics*, 7(11):873–884.

[Andersen et al., 2009] Andersen, E., Dong, M., Nielsen, M., Jahn, K., Subramani, R., Mamdouh, W., Golas, M., Sander, B., Stark, H., Oliveira, C., et al. (2009). "Self-Assembly of a Nanoscale DNA Box with a Controllable Lid". *Nature*, 459(7243):73–76.

[Antal, 2007] Antal, T. Krapivsky, P. L. (2007). "Molecular Spiders with Memory". *Phys. Rev. E*, 76:021121–021129.

[Arbuckle and Requicha, 2004] Arbuckle, D. and Requicha, A. (2004). "Active Self-Assembly". *IEEE International Conference on Robotics and Automation*, 1:896–901.

[Bath et al., 2005] Bath, J., Green, S., and Turberfield, A. (2005). "A Free-Running DNA Motor Powered by a Nicking Enzyme". *Angewandte Chemie*, 117(28):4432–4435.

[Bath and Turberfield, 2007] Bath, J. and Turberfield, A. (2007). "DNA Nanomachines". *Nature Nanotechnology*, 2(5):275–284.

[Baudrillard, 1994] Baudrillard, J. (1994). *Simulacra and Simulation*. Sheila Faria Glaser, trans. *Ann Arbor: University of Michigan Press*.

[Beaver, 1996] Beaver, D. (1996). "A Universal Molecular Computer". In Lipton, R. J. and Baum, E. B., editors, *DNA Based Computers*, pages 29–36. American Mathematical Society.

[Beck, 2011] Beck, V. (2011). Personal communication.

[Becker et al., 2006] Becker, F., Rapaport, I., and Rémila, E. (2006). "Self-Assemblying Classes of Shapes with a Minimum Number of Tiles, and in Optimal Time". In *FSTTCS: Foundations of Software Technology and Theoretical Computer Science*, volume 4337, pages 45–56. Springer.

[Bennett, 1982] Bennett, C. H. (1982). "The Thermodynamics of Computation–A Review". *International Journal of Theoretical Physics*, 21(12):905–940.

[Bonabeau et al., 1999] Bonabeau, E., Dorigo, M., and Theraulaz, G. (1999). *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press.

[Braitenberg, 1984] Braitenberg, V. (1984). *Vehicles: Experiments in Synthetic Psychology*. MIT Press.

[Breitbart et al., 1987] Breitbart, R., Andreadis, A., and Nadal-Ginard, B. (1987). "Alternative Splicing: A Ubiquitous Mechanism for the Generation of Multiple Protein Isoforms from Single Genes". *Annual Review of Biochemistry*, 56(1):467–495.

[Brooks, 1991] Brooks, R. (1991). "Intelligence without Representation". *Artificial intelligence*, 47(1-3):139–159.

[Buck et al., 2007] Buck, A., Campbell, C., Dickinson, P., Mountford, C., Stoquert, H., Terry, J., Evans, S., Keane, L., Su, T., Mount, A., et al. (2007). "DNA Nanoswitch as a Biosensor". *Analytical Chemistry*, 79(12):4724–4728.

[Buranachai et al., 2006] Buranachai, C., McKinney, S., Ha, T., et al. (2006). "Single Molecule Nanometronome". *Nano Lett*, 6(3):496–500.

[Butler et al., 2001] Butler, Z., Kotay, K., Rus, D., and Tomita, K. (2001). "Cellular Automata for Decentralized Control of Self-Reconfigurable Robots". *ICRA 2001 Workshop on Modular Self-Reconfigurable Robots*.

[Cardelli, 2011] Cardelli, L. (2011). "Strand algebras for DNA computing". *Natural Computing*, 10(1):407–428.

[Chen and Doty, 2012] Chen, H. and Doty, D. (2012). "Parallelism and Time in Hierarchical Self-Assembly". In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1163–1182. SIAM.

[Chen and Seeman, 1991] Chen, J. and Seeman, N. (1991). "Synthesis from DNA of a Molecule with the Connectivity of a Cube". *Nature*, 350(6319):631–633.

[Chirikjian, 1993] Chirikjian, G. (1993). "Metamorphic Hyper-Redundant Manipulators". *Proceedings of the JSME International Conference on Advanced Mechatronics*, pages 467–472.

[Choi, 2009] Choi, H. (2009). *Programmable In Situ Amplification for Multiplexed Bioimaging*. PhD thesis, California Institute of Technology.

[Churchman et al., 2005] Churchman, L., Ökten, Z., Rock, R., Dawson, J., and Spudich, J. (2005). "Single Molecule High-Resolution Colocalization of Cy3 and Cy5 Attached to Macromolecules Measures Intramolecular Distances through Time". *Proceedings of the National Academy of Sciences of the United States of America*, 102(5):1419–1423.

[Dabby and Chen, 2013a] Dabby, N. and Chen, H.-L. (2013a). "A Synthetic Linear Polymer that Grows Exponentially Fast". In preparation.

[Dabby and Chen, 2013b] Dabby, N. and Chen, H.-L. (2013b). "Active Self-Assembly of Simple Units Using an Insertion Primitive". In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1526–1536. SIAM.

[Dabby et al., 2013] Dabby, N., Chen, H.-L., Schaeffer, J., and Winfree, E. (2013). "Control of DNA Four-way Branch Migration Kinetics Using Toeholds". In submission to *Nucleic Acids Research*.

[Dabby, 2003] Dabby, N. L. (2003). *Poetry, Postmodernism and Contemporary Scientific Theory*. UC Berkeley English Department Undergraduate Thesis.

[Ding and Seeman, 2006] Ding, B. and Seeman, N. (2006). "Operation of a DNA Robot Arm Inserted into a 2D DNA Crystalline Substrate". *Science*, 314(5805):1583–1585.

[Dirks et al., 2007] Dirks, R., Bois, J., Schaeffer, J., Winfree, E., and Pierce, N. (2007). "Thermodynamic Analysis of Interacting Nucleic Acid Strands". *SIAM Review*, 49(1):65–88.

[Dirks and Pierce, 2004] Dirks, R. and Pierce, N. (2004). "Triggered Amplification by Hybridization Chain Reaction". *Proceedings of the National Academy of Sciences*, 101(43):15275–15278.

[Dorigo and Stützle, 2004] Dorigo, M. and Stützle, T. (2004). *Ant Colony Optimization*. Camebridge, MA, USA: MIT Press.

[Douglas et al., 2009] Douglas, S., Dietz, H., Liedl, T., Hogberg, B., Graf, F., and Shih, W. (2009). "Self-Assembly of DNA into Nanoscale Three-Dimensional Shapes". *Nature*, 459(7245):414–418.

[Duose et al., 2012] Duose, D., Schweller, R., Zimak, J., Rogers, A., Hittelman, W., and Diehl, M. (2012). "Configuring Robust DNA Strand Displacement Reactions for In Situ Molecular Analyses". *Nucleic Acids Research*, 40(7):3289–3298.

[Ehrig, 1979] Ehrig, H. (1979). "Introduction to the Algebraic Theory of Graph Grammars (A Survey)". *Lecture Notes in Computer Science*, 73:1–69.

[Eliot, 1995] Eliot, T. (1995). *The Love Song of J. Alfred Prufrock*. Harvard Vocarium Records.

[Elowitz and Leibler, 2000] Elowitz, M. and Leibler, S. (2000). "A Synthetic Oscillatory Network of Transcriptional Regulators". *Nature*, 403:335–338.

[Feller, 1968] Feller, W. (1968). An Introduction to Probability Theory and Its Applications, Vol. 1. New York: Wiley and Sons.

[Feynman, 1960] Feynman, R. (1960). "There's Plenty of Room at the Bottom". *Engineering and Science*, 23(5):22–36.

[Flamm et al., 2000] Flamm, C., Fontana, W., Hofacker, I., and Schuster, P. (2000). "RNA Folding at Elementary Step Resolution". *RNA*, 6(3):325–338.

[Foster et al., 2003] Foster, F., Zhang, M., Duckett, A., Cucevic, V., and Pavlin, C. (2003). "In Vivo Imaging of Embryonic Development in the Mouse Eye by Ultrasound Biomicroscopy". *Investigative Ophthalmology & Visual Science*, 44(6):2361–2366.

[Gajardo et al., 2002] Gajardo, A., Moreira, A., and Goles, E. (2002). "Complexity of Langton's Ant". *Discrete Applied Mathematics*, 117(1):41–50.

[Gillespie, 1977] Gillespie, D. (1977). "Exact Stochastic Simulation of Coupled Chemical Reactions". *The Journal of Physical Chemistry*, 81(25):2340–2361.

[Goldstein and Mowry, 2004] Goldstein, S. C. and Mowry, T. (2004). "Claytronics: A Scalable Basis for Future Robots". In *RoboSphere*.

[Green and Tibbetts, 1981] Green, C. and Tibbetts, C. (1981). "Reassociation Rate Limited Displacement of DNA Strands by Branch Migration". *Nucleic Acids Research*, 9(8):1905–1918.

[Green et al., 2008] Green, S., Bath, J., and Turberfield, A. (2008). "Coordinated Chemomechanical Cycles: A Mechanism for Autonomous Molecular Motion". *Physical Review Letters*, 101(23):238101–238105.

[Griffith, 2004] Griffith, S. (2004). *Growing Machines*. PhD thesis, MIT.

[Groß and Dorigo, 2008] Groß, R. and Dorigo, M. (2008). "Self-Assembly at the Macroscopic Scale". *Proceedings of the IEEE*, 96(9):1490–1508.

[Grzybowski et al., 2009] Grzybowski, B., Wilmer, C., Kim, J., Browne, K., and Bishop, K. (2009). "Self-Assembly: From Crystals to Cells". *Soft Matter*, 5(6):1110–1128.

[Gu et al., 2010] Gu, H., Chao, J., Xiao, S., and Seeman, N. (2010). "A Proximity-Based Programmable DNA Nanoscale Assembly Line". *Nature*, 465(7295):202–205.

[Harlow, 1999] Harlow, J. (1999). "Passage of an Iron Rod through the Head". *Journal of Neuropsychiatry & Clinical Neurosciences*, 11(2):281–283.

[Hasty et al., 2002] Hasty, J., McMillen, D., and Collins, J. (2002). "Engineered Gene Circuits". *Nature*, 420(6912):224–230.

[He et al., 2008] He, Y., Ye, T., Su, M., Zhang, C., Ribbe, A., Jiang, W., and Mao, C. (2008). "Hierarchical Self-Assembly of DNA into Symmetric Supramolecular Polyhedra". *Nature*, 452(7184):198–201.

[Hess, 2006a] Hess, H. (2006a). "Self-Assembly Driven by Molecular Motors". *Soft Matter*, 2(8):669–677.

[Hess, 2006b] Hess, H. (2006b). "Toward Devices Powered by Biomolecular Motors". *Science*, 312(5775):860–861.

[Heuvel and Dekker, 2007] Heuvel, M. G. L. v. d. and Dekker, C. (2007). "Motor Proteins at Work for Nanotechnology". *Science*, 317(5836):333–336.

[Holliday, 1964] Holliday, R. (1964). "A Mechanism for Gene Conversion in Fungi". *Genetical Research*, 5(2):282–304.

[Jacob et al., 1960] Jacob, F., Perrin, D., Sánchez, C., and Monod, J. (1960). "Operon: A Group of Genes with Expression Coordinated by an Operator". *CR Academy of Science Paris*, 250:1727–1729.

[Jones and Mataric, 2003] Jones, C. and Mataric, M. J. (2003). "From Local to Global Behavior in Intelligent Self-Assembly". In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 1, pages 721–726.

[Joyce, 1989] Joyce, G. (1989). "RNA Evolution and the Origins of Life". *Nature*, 338(6212):217–224.

[Jungmann et al., 2008] Jungmann, R., Liedl, T., Sobey, T., Shih, W., and Simmel, F. (2008). "Isothermal Assembly of DNA Origami Structures Using Denaturing Agents". *Journal of the American Chemical Society*, 130(31):10062–10063.

[Karsenti, 2008] Karsenti, E. (2008). "Self-Organization in Cell Biology: A Brief History". *Nature Reviews Molecular Cell Biology*, 9(3):255–262.

[Karymov et al., 2008] Karymov, M., Chinnaraj, M., Bogdanov, A., Srinivasan, A., Zheng, G., Olson, W., and Lyubchenko, Y. (2008). "Structure, Dynamics, and Branch Migration of a DNA Holliday Junction: A Single-Molecule Fluorescence and Modeling Study". *Biophysical Journal*, 95(9):4372–4383.

[Kay et al., 2007] Kay, E. R., Leigh, D. A., and Zerbetto, F. (2007). "Synthetic Molecular Motors and Mechanical Machines". *Angewandte Chemie International Edition*, 46:72–191.

[Ke et al., 2008] Ke, Y., Lindsay, S., Chang, Y., Liu, Y., and Yan, H. (2008). "Self-Assembled Water-Soluble Nucleic Acid Probe Tiles for Label-Free RNA Hybridization Assays". *Science*, 319(5860):180–183.

[Klavins et al., 2004] Klavins, E., Ghrist, R., and Lipsky, D. (2004). "Graph Grammars for Self-Assembling Robotic Systems". In *Proceedings of the International Conference on Robotics and Automation*, volume 5, pages 5293–5300.

[Kube and Zhang, 1993] Kube, C. and Zhang, H. (1993). "Collective Robotics: From Social Insects to Robots". *Adaptive Behavior*, 2(2):189–218.

[Kuczmarski et al., 2000] Kuczmarski, R., Ogden, C., Grummer-Strawn, L., Flegal, K., Guo, S., Wei, R., Mei, Z., Curtin, L., Roche, A., and Johnson, C. (2000). "CDC Growth Charts: United States". *Advance Data*, (314):1–27.

[Lederman et al., 2006] Lederman, H., Macdonald, J., Stefanovic, D., and Stojanovic, M. (2006). "Deoxyribozyme-Based Three-Input Logic Gates and Construction of a Molecular Full Adder". *Biochemistry*, 45(4):1194–1199.

[Li et al., 2000] Li, J., Zheng, W., Kwon, A., and Lu, Y. (2000). "In Vitro Selection and Characterization of a Highly Efficient Zn (II)-Dependent RNA-Cleaving Deoxyribozyme". *Nucleic Acids Research*, 28(2):481–488.

[Lindenmayer, 1987] Lindenmayer, A. (1987). "Models for Multicellular Development: Characterization, Inference and Complexity of L-Systems". *Trends, Techniques, and Problems in Theoretical Computer Science*, pages 138–168.

[Liu and Fletcher, 2009] Liu, A. and Fletcher, D. (2009). "Biology Under Construction: In Vitro Reconstitution of Cellular Function". *Nature Reviews Molecular Cell Biology*, 10(9):644–650.

[Liu et al., 1994] Liu, B., Leontis, N., and Seeman, N. (1994). "Bulged 3-Arm DNA Branched Junctions as Components for Nanoconstruction". *Nanobiology*, 3:177–188.

[Lu and Lieber, 2007] Lu, W. and Lieber, C. (2007). "Nanoelectronics from the Bottom Up". *Nature Materials*, 6(11):841–850.

[Lubrich et al., 2009] Lubrich, D., Green, S., and Turberfield, A. (2009). "Kinetically Controlled Self-Assembly of DNA Oligomers". *Journal of the American Chemical Society*, 131(7):2422–2423.

[Lubrich et al., 2008] Lubrich, D., Lin, J., and Yan, J. (2008). "A Contractile DNA Machine". *Angewandte Chemie International Edition*, 47(37):7026–7028.

[Luecke et al., 1999] Luecke, R., Wosilait, W., and Young, J. (1999). "Mathematical Modeling of Human Embryonic and Fetal Growth Rates". *Growth, Development, and Aging*, 63(1-2):49–59.

[Lund, 2008] Lund, K. (2008). *"Self-Assembling DNA Based Molecular Pegboards as Nanoscale Scaffolds and Prescriptive Molecular Landscapes"*. PhD thesis, Arizona State University.

[Lund et al., 2010] Lund, K., Manzo, A., Dabby, N., Michelotti, N., Johnson-Buck, A., Nangreave, J., Taylor, S., Pei, R., Stojanovic, M., Walter, N., et al. (2010). "Molecular Robots Guided by Prescriptive Landscapes". *Nature*, 465(7295):206–210.

[Mao et al., 2000] Mao, C., LaBean, T., Reif, J., and Seeman, N. (2000). "Logical Computation Using Algorithmic Self-Assembly of DNA Triple-Crossover Molecules". *Nature*, 407(6803):493–496.

[Mao et al., 1999] Mao, C., Sun, W., Shen, Z., and Seeman, N. (1999). "A DNA Nanomechanical Device Based on the B–Z Transition". *Nature*, 397:144–146.

[Mathews et al., 2004] Mathews, D., Disney, M., Childs, J., Schroeder, S., Zuker, M., and Turner, D. (2004). "Incorporating Chemical Modification Constraints into a Dynamic Programming Algorithm for Prediction of RNA Secondary Structure". *Proceedings of the National Academy of Sciences of the United States of America*, 101(19):7287–7292.

[McKinney et al., 2002] McKinney, S., Déclais, A., Lilley, D., and Ha, T. (2002). "Structural Dynamics of Individual Holliday Junctions". *Nature Structural Biology*, 10(2):93–97.

[McKinney et al., 2005] McKinney, S., Freeman, A., Lilley, D., and Ha, T. (2005). "Observing Spontaneous Branch Migration of Holliday Junctions One Step at a Time". *Proceedings of the National Academy of Sciences of the United States of America*, 102(16):5715–5720.

[Metropolis et al., 1953] Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., and Teller, E. (1953). "Equation of State Calculations by Fast Computing Machines". *The Journal of Chemical Physics*, 21(6):1087.

[Metzger et al., 2008] Metzger, R., Klein, O., Martin, G., and Krasnow, M. (2008). "The Branching Programme of Mouse Lung Development". *Nature*, 453(7196):745–750.

[Morris et al., 2001] Morris, D., Grealy, M., Leese, H., and Centre, G. R. (2001). *Cattle embryo growth, development and viability*. Grange Research Centre.

[Moulton et al., 2000] Moulton, V., Zuker, M., Steel, M., Pointon, R., and Penny, D. (2000). "Metrics on RNA Secondary Structures". *Journal of Computational Biology*, 7(1-2):277–292.

[Murata et al., 1994] Murata, S., Kurokawa, H., and Kokaji, S. (1994). "Self-Assembling Machine". In *Proceedings of the 1994 International Conference on Robotics and Automation*, pages 441–448.

[Muscat et al., 2011] Muscat, R., Bath, J., and Turberfield, A. (2011). "A Programmable Molecular Robot". *Nano Letters*, 11(3):982–987.

[Nagpal, 2008] Nagpal, R. (2008). "Programmable Pattern-Formation and Scale-Independence". In *Unifying Themes in Complex Systems IV*, pages 275–282. Springer.

[Omabegho et al., 2009] Omabegho, T., Sha, R., and Seeman, N. (2009). "A Bipedal DNA Brownian Motor with Coordinated Legs". *Science*, 324(5923):67–71.

[Panyutin and Hsieh, 1993] Panyutin, I. and Hsieh, P. (1993). "Formation of a Single Base Mismatch Impedes Spontaneous DNA Branch Migration". *Journal of Molecular Biology*, 230(2):413–424.

[Panyutin and Hsieh, 1994] Panyutin, I. and Hsieh, P. (1994). "The Kinetics of Spontaneous DNA Branch Migration". *Proceedings of the National Academy of Sciences*, 91(6):2021–2025.

[Pei et al., 2006] Pei, R., Taylor, S., Stefanovic, D., Rudchenko, S., Mitchell, T., and Stojanovic, M. (2006). "Behavior of Polycatalytic Assemblies in a Substrate-Displaying Matrix". *J. Am. Chem. Soc*, 128(39):12693–12699.

[Prusinkiewicz, 2004] Prusinkiewicz, P. (2004). "Modeling Plant Growth and Development". *Current Opinion in Plant Biology*, 7(1):79–83.

[Qian and Winfree, 2011] Qian, L. and Winfree, E. (2011). "Scaling Up Digital Circuit Computation with DNA Strand Displacement Cascades". *Science*, 332(6034):1196–1201.

[Qian et al., 2011] Qian, L., Winfree, E., and Bruck, J. (2011). "Neural Network Computation with DNA Strand Displacement Cascades". *Nature*, 475(7356):368–372.

[Rosa et al., 2006] Rosa, M. D., Goldstein, S., Lee, P., Campbell, J., and Pillai, P. (2006). "Scalable Shape Sculpting via Hole Motion: Motion Planning in Lattice-Constrained Modular Robots". In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, pages 1462–1468.

[Rothemund, 1996] Rothemund, P. (1996). "A DNA and Restriction Enzyme Implementation of Turing Machines". In Lipton, R. and Baum, E., editors, *DNA Based Computers*, pages 75–119. American Mathematical Society.

[Rothemund, 2006] Rothemund, P. (2006). "Folding DNA to Create Nanoscale Shapes and Patterns". *Nature*, 440(7082):297–302.

[Rothemund et al., 2004] Rothemund, P., Papadakis, N., and Winfree, E. (2004). "Algorithmic Self-Assembly of DNA Sierpinski Triangles". *PLoS Biology*, 2(12):e424.

[Rus et al., 2002] Rus, D., Butler, Z., Kotay, K., and Vona, M. (2002). "Self-Reconfiguring Robots". *Communications of the ACM*, 45(3):39–45.

[Rus and Vona, 2001] Rus, D. and Vona, M. (2001). "Crystalline Robots: Self-Reconfiguration with Compressible Unit Modules". *Autonomous Robots*, 10(1):107–124.

[Saffarian et al., 2004] Saffarian, S., Collier, I., Marmer, B., Elson, E., and Goldberg, G. (2004). "Interstitial Collagenase is a Brownian Ratchet Driven by Proteolysis of Collagen". *Science*, 306(5693):108–111.

[Saitou, 1999] Saitou, K. (1999). "Conformational Switching in Self-Assembling Mechanical Systems". *IEEE Transactions on Robotics and Automation*, 15(3):510–520.

[SantaLucia, 1998] SantaLucia, J. (1998). "A Unified View of Polymer, Dumbbell, and Oligonucleotide DNA Nearest-Neighbor Thermodynamics". *Proceedings of the National Academy of Sciences*, 95(4):1460–1465.

[SantaLucia Jr and Hicks, 2004] SantaLucia Jr, J. and Hicks, D. (2004). "The Thermodynamics of DNA Structural Motifs". *Annual Reviews of Biophysics and Biomolecular Structure*, 33:415–440.

[Santoro and Joyce, 1997] Santoro, S. and Joyce, G. (1997). "A General Purpose RNA-Cleaving DNA Enzyme". *Proceedings of the National Academy of Sciences*, 94(9):4262.

[Schaeffer, 2012] Schaeffer, J. (2012). *The Multistrand Simulator: Stochastic Simulation of the Kinetics of Multiple Interacting DNA Strands*. Master's thesis, California Institute of Technology.

[Seelig et al., 2006] Seelig, G., Soloveichik, D., Zhang, D., and Winfree, E. (2006). "Enzyme-Free Nucleic Acid Logic Circuits". *Science*, 314(5805):1585–1588.

[Seeman, 2005] Seeman, N. (2005). "DNA Enables Nanoscale Control of the Structure of Matter". *Quarterly Reviews of Biophysics*, 38(4):363–371.

[Seeman and Kallenbach, 1994] Seeman, N. and Kallenbach, N. (1994). "DNA Branched Junctions". *Annual Review of Biophysics and Biomolecular Structure*, 23(1):53–86.

[Shannon et al., 1956] Shannon, C., McCarthy, J., et al. (1956). *Automata Studies*. Princeton University Press.

[Sherman and Seeman, 2004] Sherman, W. and Seeman, N. (2004). "A Precisely Controlled DNA Biped Walking Device". *Nano Letters*, 4(7):1203–1207.

[Shin and Pierce, 2004] Shin, J. and Pierce, N. (2004). "A Synthetic DNA Walker for Molecular Transport". *Journal of the American Chemical Society*, 126(35):10834–10835.

[Siegwart, 2004] Siegwart, R. Nourbakhsh, I. R. (2004). *Introduction to Autonomous Mobile Robots*. MIT Press.

[Simmel and Yurke, 2002] Simmel, F. and Yurke, B. (2002). "A DNA-Based Molecular Device Switchable between Three Distinct Mechanical States". *Applied Physics Letters*, 80(5):883–885.

[Simon, 1996] Simon, H. (1996). *The Sciences of the Artificial*. MIT press.

[Sipser, 2006] Sipser, M. (2006). *Introduction to the Theory of Computation*. Thomson Course Technology.

[Siromoney, 1986] Siromoney, R. (1986). "Array languages and Lindenmayer Systems–A Survey". In *The Book of L*, pages 413–426. Springer.

[Smith and Schweitzer, 1996] Smith, W. and Schweitzer, A. (1996). "DNA Computers In Vitro and Vivo". In Lipton, R. and Baum, E., editors, *DNA Based Computers*, pages 121–185. American Mathematical Society.

[Soloveichik et al., 2010] Soloveichik, D., Seelig, G., and Winfree, E. (2010). "DNA as a Universal Substrate for Chemical Kinetics". *Proceedings of the National Academy of Sciences*, 107(12):5393–5398.

[Soloveichik and Winfree, 2005] Soloveichik, D. and Winfree, E. (2005). "Complexity of Self-Assembled Shapes". *DNA Computing*, pages 344–354.

[Stojanovic et al., 2002] Stojanovic, M., Mitchell, T., and Stefanovic, D. (2002). "Deoxyribozyme-Based Logic Gates". *Journal of the American Chemical Society*, 124(14):3555–3561.

[Stojanovic et al., 2003] Stojanovic, M., Stefanovic, D., et al. (2003). "A Deoxyribozyme-Based Molecular Automaton". *Nature biotechnology*, 21(9):1069–1074.

[Stupp, 2010] Stupp, S. (2010). "Self-Assembly and Biomaterials". *Nano Letters*, 10(12):4783–4786.

[Tataurov et al., 2008] Tataurov, A., You, Y., and Owczarzy, R. (2008). "Predicting Ultraviolet Spectrum of Single Stranded and Double Stranded Deoxyribonucleic Acids". *Biophysical Chemistry*, 133(1):66–70.

[Thompson et al., 1976] Thompson, B., Camien, M., and Warner, R. (1976). "Kinetics of Branch Migration in Double-Stranded DNA". *Proceedings of the National Academy of Sciences*, 73(7):2299–2303.

[Tian et al., 2005] Tian, Y., He, Y., Chen, Y., Yin, P., and Mao, C. (2005). "A DNAzyme that Walks Processively and Autonomously along a One-Dimensional Track". *Angewandte Chemie International Edition*, 44(28):4355–4358.

[Toffler, 1984] Toffler, A. (1984). *Future shock*. Bantam.

[Turing, 1936] Turing, A. M. (1936). "On Computable Numbers, with an Application to the Entscheidungsproblem". *Proceedings of the London Mathematical Society*, 42(2):230–265.

[Venkataraman et al., 2007] Venkataraman, S., Dirks, R., Rothemund, P., Winfree, E., and Pierce, N. (2007). "An Autonomous Polymerization Motor Powered by DNA Hybridization". *Nature Nanotechnology*, 2(8):490–494.

[Venter et al., 2001] Venter, J., Adams, M., Myers, E., Li, P., Mural, R., Sutton, G., Smith, H., Yandell, M., Evans, C., Holt, R., et al. (2001). "The Sequence of the Human Genome". *Science*, 291(5507):1304–1351.

[Von Neumann and Burks, 1966] Von Neumann, J. and Burks, A. W. (1966). *"Theory of Self-Reproducing Automata"*. University of Illinois Press.

[Walter et al., 2004] Walter, J. E., Welch, J. L., and Amato, N. M. (2004). "Distributed Reconfiguration of Metamorphic Robot Chains". *Distributed Computing*, 17(2):171–189.

[Walter et al., 2008] Walter, N., Huang, C., Manzo, A., and Sobhy, M. (2008). "Do-It-Yourself Guide: How To Use the Modern Single-Molecule Toolkit". *Nature Methods*, 5(6):475–489.

[Watson and Crick, 1953] Watson, J. and Crick, F. (1953). "Molecular Structure of Nucleic Acids". *Nature*, 171(4356):737–738.

[Werfel and Nagpal, 2007] Werfel, J. and Nagpal, R. (2007). "Towards a Common Comparison Framework for Global-to-Local Programming of Self-Assembling Robotic Systems". *IEEE Conference on Intelligent Robots and Systems*.

[Wetmur, 1976] Wetmur, J. (1976). "Hybridization and Renaturation Kinetics of Nucleic Acids". *Annual Review of Biophysics and Bioengineering*, 5(1):337–361.

[White et al., 2004] White, P., Kopanski, K., and Lipson, H. (2004). "Stochastic Self-Reconfigurable Cellular Robotics". *IEEE International Conference on Robotics and Automation*, 3:2888–2893.

[Whitesides and Grzybowski, 2002] Whitesides, G. and Grzybowski, B. (2002). "Self-Assembly at all Scales". *Science*, 295(5564):2418–2421.

[Win and Smolke, 2008] Win, M. N. and Smolke, C. D. (2008). "Higher-Order Cellular Information Processing with Synthetic RNA Devices". *Science*, 322(5900):456–460.

[Winfree, 1996] Winfree, E. (1996). "On the Computational Power of DNA Annealing and Ligation". In Lipton, R. and Baum, E., editors, *DNA Based Computers*, pages 199–221. American Mathematical Society, Providence, RI.

[Winfree, 2000] Winfree, E. (2000). "Algorithmic Self-Assembly of DNA: Theoretical Motivations and 2D Assembly Experiments". *Journal of Biomolecular Structure and Dynamics*, 17(Supplement 1):263–270.

[Winfree, 2012] Winfree, E. (2012). DNA design toolbox. URL http://www.dna.caltech.edu/dnadesign/.

[Winfree et al., 1998] Winfree, E., Liu, F., Wenzler, L., and Seeman, N. (1998). "Design and Self-Assembly of Two-Dimensional DNA Crystals". *Nature*, 394(6693):539–544.

[Woods et al., 2013] Woods, D., Chen, H.-L., Goodfriend, S., Dabby, N., Winfree, E., and Yin, P. (2013). "Active Self-Assembly of Algorithmic Shapes and Patterns in Polylogarithmic Time".

In *ITCS 2013: Innovations in Theoretical Computer Science*, pages 353–354, Berkeley, CA. ACM.

[Woolf, 1957] Woolf, V. (1957). A room of one's own. 1929. *New York: Harvest-Harcourt*.

[Yan et al., 2003] Yan, H., Park, S., Finkelstein, G., Reif, J., and LaBean, T. (2003). "DNA-Templated Self-Assembly of Protein Arrays and Highly Conductive Nanowires". *Science*, 301(5641):1882–1884.

[Yan et al., 2002] Yan, H., Zhang, X., Shen, Z., and Seeman, N. (2002). "A Robust DNA Mechanical Device Controlled by Hybridization Topology". *Nature*, 415(6867):62–65.

[Yang et al., 1998] Yang, X., Vologodskii, A., Liu, B., Kemper, B., and Seeman, N. (1998). "Torsional Control of Double-Stranded DNA Branch Migration". *Peptide Science*, 45(1):69–83.

[Yildiz and Selvin, 2005] Yildiz, A. and Selvin, P. (2005). "Fluorescence Imaging with One Nanometer Accuracy: Application to Molecular Motors". *Accounts of Chemical Research*, 38(7):574–582.

[Yim et al., 1997] Yim, M., Lamping, J., Mao, E., and Chase, J. G. (1997). "Rhombic Dodecahedron Shape for Self-Assembling Robots". *SPL TechReport P9710777, Xerox PARC*.

[Yim et al., 2007] Yim, M., Shen, W., Salemi, B., Rus, D., Moll, M., Lipson, H., Klavins, E., and Chirikjian, G. S. (2007). "Modular Self-Reconfigurable Robot Systems". *IEEE Robotics and Automation Magazine*, 14(1):43.

[Yin et al., 2008] Yin, P., Choi, H., Calvert, C., and Pierce, N. (2008). "Programming Biomolecular Self-Assembly Pathways". *Nature*, 451(7176):318–322.

[Yin et al., 2004] Yin, P., Yan, H., Daniell, X., Turberfield, A., and Reif, J. (2004). "A Unidirectional DNA Walker that Moves Autonomously along a Track". *Angewandte Chemie International Edition*, 43(37):4906–4911.

[Yurke and Mills, 2003] Yurke, B. and Mills, A. (2003). "Using DNA to Power Nanostructures". *Genetic Programming and Evolvable Machines*, 4(2):111–122.

[Yurke et al., 2000] Yurke, B., Turberfield, A., Mills, A., Simmel, F., and Neumann, J. (2000). "A DNA-Fuelled Molecular Machine Made of DNA". *Nature*, 406(6796):605–608.

[Zadeh et al., 2010] Zadeh, J., Steenberg, C., Bois, J., Wolfe, B., Pierce, M., Khan, A., Dirks, R., and Pierce, N. (2010). "NUPACK: Analysis and Design of Nucleic Acid Systems". *Journal of Computational Chemistry*, 32:170–173.

[Zadeh et al., 2011] Zadeh, J., Wolfe, B., and Pierce, N. (2011). "Nucleic Acid Sequence Design via Efficient Ensemble Defect Optimization". *Journal of Computational Chemistry*, 32:439–452.

[Zhang and Seelig, 2011] Zhang, D. and Seelig, G. (2011). "Dynamic DNA Nanotechnology Using Strand-Displacement Reactions". *Nature Chemistry*, 3(2):103–113.

[Zhang et al., 2007] Zhang, D., Turberfield, A., Yurke, B., and Winfree, E. (2007). "Engineering Entropy-Driven Reactions and Networks Catalyzed by DNA". *Science*, 318(5853):1121.

[Zhang and Winfree, 2009] Zhang, D. and Winfree, E. (2009). "Control of DNA Strand Displacement Kinetics Using Toehold Exchange". *Journal of the American Chemical Society*, 131(47):17303–17314.