

Design and Engineering of Pattern Formation in Gene Expression in Escherichia coli

Justin Hsia



Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2016-32

<http://www.eecs.berkeley.edu/Pubs/TechRpts/2016/EECS-2016-32.html>

May 1, 2016

Copyright © 2016, by the author(s).
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

**Design and Engineering of Pattern Formation in Gene Expression in
*Escherichia coli***

by

Justin Ezekiel Hsia

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Engineering – Electrical Engineering and Computer Sciences

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Murat Arcak, Chair
Professor Michel M. Maharbiz
Professor Adam P. Arkin

Summer 2015

Design and Engineering of Pattern Formation in Gene Expression in
Escherichia coli

Copyright 2015
by
Justin Ezekiel Hsia

Abstract

Design and Engineering of Pattern Formation in Gene Expression in *Escherichia coli*

by

Justin Ezekiel Hsia

Doctor of Philosophy in Engineering – Electrical Engineering and Computer Sciences

University of California, Berkeley

Professor Murat Arcaç, Chair

Synthetic biology aims to develop new biological systems and devices, from the modification of existing pathways to the construction of entirely new genetic circuits. The role of the engineer in synthetic biology is to apply engineering principles to the design and analysis of proposed systems. Building biological systems *de novo* are the best way to demonstrate our successes on this front and so far has yielded biological devices such as synthetic promoters, toggle switches, oscillators, and logic gates. Here we aim to push the current boundaries of synthetic biology and study the principles behind engineering pattern formation in ensembles of *E. coli* cells. Motivated by the study of morphogenesis, we hope to develop synthetic systems with the high-minded goals of one day engineering molecular differentiation or even multicellularity. Pattern formation will be a critical part of these goals and brings an additional focus on cell-to-cell communication and signaling molecules. Here we examine two different communication mechanisms, quorum sensing and contact-based signaling, and see what types of patterns we can achieve.

Using quorum sensing, we focus on diffusion-driven instability (Turing patterning), where a homogeneous steady state of an ensemble of cells is destabilized in the presence of diffusion. This is made possible by the conflicting interactions of the internal dynamics of the cells and the normalizing effect of diffusion between them. The work in this area thus far has centered around activator-inhibitor network theory and to date has yet to yield a biological experimental demonstration. Here we analyze the Turing mechanism and propose a new network which we call a “quenched oscillator” system and demonstrate its ability to produce diffusion-driven instability. We then propose a synthetic implementation and present work towards a partial implementation. In the process, we use zinc finger proteins (ZFPs) and small RNAs (sRNAs) to construct new synthetic inverters to put together in a ring oscillator for use in a quenched oscillator system.

Interest in contact-based signaling has risen recently with the discovery of a contact-dependent inhibition (CDI) system in *E. coli*. While a synthetic contact-mediated communication channel has not yet been achieved, its realization will provide a huge boost in engineering possibilities, particularly for multicellular applications. Here we develop an

analytical framework based on graph theory for analyzing lateral inhibition networks, a category that CDI falls under, for the existence and stability of equitable patterns. Without an actual CDI system to use, we develop what we call a “compartmental lateral inhibition” system using diffusible molecules and engineered communication channels to simulate contact-mediated signaling for verification of our patterning analysis. The current state of our synthetic implementation is presented, highlighting experimental setup details that may prove useful for future applications in engineered multicellular ensembles.

This dissertation is dedicated everyone who has steadfastly supported me and given me the strength to see this remarkable journey to its completion:

To my parents, for shaping me into the grounded and hard-working person I am today and for giving me the space the find my own direction.

To UFO, for teaching me to laugh at and enjoy the unpredictability of life.

To AiR and Vacuum, for keeping music in my life.

And to **Cal**, for teaching me everything I know.

Contents

Contents	ii
List of Figures	vi
List of Tables	viii
1 Introduction	1
1.1 Preface	1
1.2 Synthetic Biology	1
1.3 Pattern Formation	2
1.3.1 Motivation	2
1.3.2 Diffusion-based Signaling	3
1.3.3 Contact-based Signaling	4
2 Turing Patterning	5
2.1 Diffusion-Driven Instability	5
2.2 Reaction-Diffusion System Analysis	6
2.2.1 Additive D-Stability	7
2.2.2 Conditions for Turing Patterning	8
2.3 Activator-Inhibitor Theory	9
2.3.1 Activator-Inhibitor Example	10
2.3.2 Activator-Inhibitor Difficulties	13
2.4 Quenched Oscillator Theory	15
2.4.1 Quenched Oscillator Toy Model Analysis	16
2.4.2 Discussion of the Quenched Oscillator Toy Model	20
2.5 Quenched Oscillator Implementation with an Existing Oscillator	21
2.5.1 Quenched Oscillator PDE Model and Analysis	22
2.5.2 Model Parameter Selection and PDE Simulation	25
2.5.3 Parameter Selection for Stochastic Simulations	32
2.5.4 Stochastic Model and Simulation	33
2.5.5 Discussion of Quenched Oscillator Implementation	41
2.6 Newer Synthetic Technologies Improve Quenched Oscillator Design	42

2.6.1	Zinc Finger Protein Technology	42
2.6.2	Hybrid sRNA-Repressor Topology for Increased Cooperativity	43
2.6.3	Zinc Finger Protein Oscillator	46
2.6.4	ZFP-sRNA Quenched Oscillator	47
2.7	ZFP-sRNA Inverter Experimental Results	53
2.7.1	Experimental Design	53
2.7.2	Parts Selection	54
2.7.3	Plasmid Design	57
2.7.4	Inverter Results	59
2.8	Turing Patterning Discussion	62
2.8.1	Implications of the Novel Architecture	63
2.8.2	Inverter Performance	64
2.8.3	Turing Patterning Future Work	66
2.9	Turing Patterning Materials and Methods	67
2.9.1	Computational	67
2.9.2	PDE Simulations	68
2.9.3	Stochastic Simulations	68
2.9.4	Discrete Cosine Transforms	68
2.9.5	Construction of Plasmids	69
2.9.6	Experimental Conditions and Procedure	72
3	Lateral Inhibition	73
3.1	Lateral Inhibition and Contact-Dependent Inhibition	73
3.1.1	CDI as a Transport System	74
3.2	Lateral Inhibition System Analysis	75
3.2.1	Lateral Inhibition Model	76
3.2.2	Lateral Inhibition Patterns	77
3.3	Compartmental Lateral Inhibition System	78
3.3.1	Compartmental Lateral Inhibition Analysis	81
3.3.2	Existence and Proof of Contrasting Patterns	83
3.3.3	Patterning Region	86
3.4	Lateral Inhibition Experimental Results	91
3.4.1	Experimental Design	91
3.4.2	Parts Selection	94
3.4.3	Plasmid Design	97
3.4.4	Parts Testing Results	98
3.4.5	Compartmental Lateral Inhibition Partial Results	101
3.5	Lateral Inhibition Discussion	103
3.5.1	Implications of Compartmental Lateral Inhibition	103
3.5.2	Lateral Inhibition Future Work	103
3.6	Lateral Inhibition Materials and Methods	105
3.6.1	Computational	105

3.6.2	Construction of Plasmids	106
3.6.3	Experimental Conditions and Procedure	108
3.6.4	Development of Experimental Assay	109
3.6.5	Robotics Protocols	113
4	Conclusion	116
	Bibliography	118
A	Sample Code	128
A.1	Analysis Code - Quenched Oscillator	128
A.2	Analysis Code - Quenched Oscillator Function	129
A.3	Analysis Code - Extract Parameters	135
A.4	PDE Code - MATLAB Simulation Script	136
A.5	PDE Code - Equations	138
A.6	PDE Code - Parameters List	139
A.7	PDE Code - Steady State Function	140
A.8	SSC Code - Simulation Reaction File	141
A.9	SSC Code - Simulation Parameter File	142
A.10	SSC Code - SSC Simulation Script	144
A.11	SSC Code - Config File	145
A.12	SSC Code - Generate Initial Condition	145
A.13	SSC Code - Submit Jobs	147
A.14	SSC Code - Base Batch Job File	148
A.15	SSC Code - Format SSC Output File	149
A.16	SSC Code - Read Output File into MATLAB	150
A.17	SSC Code - Read Output File Helper	153
A.18	Discrete Cosine Transform Code	154
A.19	Excel Macro - Adjust Data	155
A.20	Excel Macro - Aggregate Replicates	157
B	Full SSC Simulation Results	160
C	Selected Sequence Info	163
C.1	Inv Plasmid 1: precursor (pJH4-21)	163
C.2	Inv Plasmid 1: ZFP16-59, s04 (pJH7-17)	166
C.3	Inv Plasmid 1: ZFP16-57, s05 (pJH9-56)	169
C.4	Inv Plasmid 1: ZFP16-56, sLS (pJH4-64)	171
C.5	Inv Plasmid 2: J23108, a04 (pJH5-56)	174
C.6	Inv Plasmid 2: J23118, a05 (pJH6-26)	176
C.7	Inv Plasmid 2: J23106, aLS (pWH39-27)	177
C.8	Inv Control: positive ind (pJH4-37)	178

C.9 LatInh Testing: las sender (pJH9-22)	181
C.10 LatInh Testing: lux receiver Amp (pJH5-4)	182
C.11 LatInh Testing: reporter Amp (pJH5-29)	185
C.12 LatInh Testing: lux receiver Cm (pJH9-35)	186
C.13 LatInh Plasmid 1: reporter Cm (pJH4-22)	189
C.14 LatInh Plasmid 2: cell type B (pJH5-66)	191
C.15 LatInh Plasmid 3: cell type A (pJH9-43)	193
C.16 LatInh Plasmid 3: cell type B (pJH9-44)	196

List of Figures

2.1	The canonical activator-inhibitor system	9
2.2	Example of experimentally unfeasible activator-inhibitor system	10
2.3	Activator-inhibitor subsystem instability range	15
2.4	The quenched oscillator system	16
2.5	Toy model 1 PDE sim - diffusion and growth	18
2.6	Toy model 1 PDE sim - diffusion and decay	19
2.7	Toy model 1 PDE sim - no diffusion	19
2.8	Negative feedback diagram for diffusion-driven instability of toy model 1	20
2.9	Bifurcation diagram for toy model 1	21
2.10	First proposed quenched oscillator implementation	22
2.11	Quenched oscillator 1 PSet1 PDE sim - diffusion and growth	28
2.12	Quenched oscillator 1 PSet1 PDE sim - diffusion and decay	29
2.13	Quenched oscillator 1 PSet1 PDE sim - no diffusion	30
2.14	Comparison of steady-state concentrations for Parameter Sets 1 and 2	32
2.15	Quenched oscillator 1 PSet2 PDE sim - diffusion and growth.	34
2.16	Quenched oscillator 1 PSet2 PDE sim - excited wave numbers.	35
2.17	Quenched oscillator 1 PSet2 PDE sim - single cell with diffusion	37
2.18	Quenched oscillator 1 PSet2 PDE sim - single cell without diffusion	38
2.19	Quenched oscillator 1 PSet2 stochastic sim - single cell with diffusion	38
2.20	Quenched oscillator 1 PSet2 stochastic sim - single cell without diffusion	39
2.21	Quenched oscillator 1 PSet2 stochastic sim - plots and DCTs	40
2.22	Diagram of structure of zinc finger proteins	43
2.23	Diagram of sRNA thresholding	43
2.24	Graphical interpretation of ZFP-sRNA inverter gain	45
2.25	Schematic of the ZFP-sRNA oscillator	46
2.26	ZFP-sRNA oscillator PDE and stochastic sims	48
2.27	Modified quenched oscillator system	48
2.28	Toy model 2 PDE sim - diffusion and growth	50
2.29	Toy model 2 PDE sim - diffusion and decay	51
2.30	Toy model 2 PDE sim - no diffusion	51
2.31	Bifurcation diagram for toy model 2	52
2.32	Second proposed quenched oscillator implementation	52

2.33	ZFP-sRNA Inverter Schematic	53
2.34	Effect of sRNA on ZFP inverter	54
2.35	Orthogonality and recognition sequences of chosen ZFPs	55
2.36	Inverter data with different Anderson constitutive promoters	56
2.37	Plasmid designs for ZFP-sRNA inverters	57
2.38	Plasmid designs for two ZFP-sRNA inverters in series	58
2.39	Plasmid designs to test viability of two inverters in series	59
2.40	Inverter data for 3 ZFPs and 3 sRNAs	60
2.41	Visualization of Inverter Induction Fold	61
2.42	Visualization of Inverter Maximum Output	61
2.43	Inverter data showing effect of 2nd operon location	62
2.44	Inverter data showing effect of sense region 5'-UTR	62
3.1	Toy model of a lateral inhibition system in a line of cells	74
3.2	Diagram of the parts in the natural CDI system in <i>E. coli</i>	75
3.3	Conserved regions of CdiA chimaeras	76
3.4	Example of steady state patterns on a 2-D mesh	78
3.5	Example of network geometries for compartmental lateral inhibition system	79
3.6	Diagram of components in compartmental lateral inhibition system	80
3.7	Network diagram of compartmental lateral inhibition system model	82
3.8	Typical steady-state I/O maps for compartmental lateral inhibition system	85
3.9	Compartmental lateral inhibition system model split into network blocks	87
3.10	Patterning region for compartmental lateral inhibition	90
3.11	Parts testing plasmids for compartmental lateral inhibition	92
3.12	Initial selection of receptor protein constitutive promoter	96
3.13	Testing plasmid designs for compartmental lateral inhibition	97
3.14	Plasmid designs for compartmental lateral inhibition	98
3.15	Combined plasmid design for compartmental lateral inhibition	99
3.16	Receivers with AHL induction data	99
3.17	Senders and receivers colonies on agar	100
3.18	“Matching pair” experiment data verifying TetR behavior	101
3.19	Receiving sensitivity experiment data in channels	102
3.20	Cell type biased experiment data in channels	102
3.21	Layout and dimensions for Insert 1	110
3.22	Layout and dimensions for Insert 2	111
3.23	Robotics setup for channel device	114
B.1	Stochastic simulation for Parameter Set 2 in line of cells with diffusion (1)	161
B.2	Stochastic simulation for Parameter Set 2 in line of cells with diffusion (2)	162

List of Tables

2.1	Base parameter values for activator-inhibitor system	13
2.2	Acceptable ranges and chosen parameter values used for simulations	27
2.3	Steady-state concentrations for parameter sets	31
2.4	System characteristics for parameter sets	31
2.5	Full reaction set for stochastic simulations	36
2.6	Comparison of gains for different inverters	46
2.7	Synthetic promoters used in ZFP-sRNA inverters	55
2.8	Plasmids used in ZFP-sRNA inverters	70
2.9	Oligonucleotides used in the construction of the ZFP-sRNA inverter plasmids	71
3.1	Parameter values for compartmental lateral inhibition system simulation	89
3.2	Candidate AHLs for Lateral Inhibition	95
3.3	Plasmids used in compartment lateral inhibition	105
3.4	Oligonucleotides used in the construction of the lateral inhibition plasmids	107

Acknowledgments

If someone had described my graduate work to my younger self, I would have called him or her crazy. My graduate career has been unexpected and transformative and none of that would have been possible without the help of the dedicated individuals around me who helped mentor and teach me. I have been blessed to maintain a consistent research focus throughout my graduate studies and to discover that I love teaching, and so I owe a special debt of gratitude to the following people who helped me along the way:

Research:

I want to thank Murat Arcak for being an extremely understanding advisor, giving me the opportunity to explore this new field, and providing me with much-needed direction throughout the years. Special thanks for indulging my desire to teach.

I want to thank my longtime mentor William Holtz, who taught me everything I know about wet lab work and whose earlier tribulations paved the way for this dissertation and my ventures into synthetic biology. May your legacy forever live in JBEI lore.

Many thanks as well to Michel Maharbiz for livening up meetings with your infectious enthusiasm and ridiculous humor. I hope to one day replicate your unique ability to dream big while staying both grounded and focused.

Thanks to Adam Arkin and Jay Keasling for their advice and interest in my projects and allowing me to utilize their lab resources.

Thanks to Ana Sophia Rufino Ferreira for her huge contributions to the lateral inhibition project and willingness to dig into some of the bio with me.

Thanks to research collaborators and others I have interacted with throughout the years: My fellow group members Yusef Shafi and Sam Coogan for their help with the administrative of grad school and theory work. Daniel Huang and David Chen for bio-related work and ideas.

Teaching:

I want to thank Dan Garcia for being an awesome role model as a lecturer and for giving me the opportunity to be a summer instructor.

I want to thank Claire Tomlin and Bharathwaj Muthuswamy for sparking my interest in teaching and giving me the opportunity to learn about course design in EE128.

And finally thanks to my teaching staff and students over the years for being my guinea pigs. Hopefully the experience was as worthwhile for you as it was for me.

Chapter 1

Introduction

1.1 Preface

Despite its simple-sounding name, synthetic biology is a wondrously convoluted interdisciplinary field. It captures the attention of biologists, chemists, physicists, engineers, and computer scientists alike and yet we sometimes still struggle to come to an agreement over its very definition. Each discipline brings with it its own unique set of skills and views and this can lead to harmonious collaboration as well as clashes in opinions. But this is what makes synthetic biology both exciting and frustrating – the field is still ripe with promise, but every bump along the road is humbling and forces you to learn something new.

And so here I offer up a definition of synthetic biology as broad as the field itself: **“the manipulation or creation of biological systems for ‘useful’ purposes.”** This means vastly different things to different people and my own internal definition continues to evolve as I learn more and more. Working in this field has given me a newfound appreciation for the fact that most living things just... work, somehow. And despite the constant setbacks, I still get giddy at the thought of the possibilities in this field for the future. It has been so much fun to push the boundaries of what we want to build even if the basics of forward engineering of biology are still being investigated. We are getting there little by little and I’m proud and thankful to have played a small part of that.

1.2 Synthetic Biology

Synthetic biology aims to develop new biological systems and devices, from the modification of existing pathways to the construction of entirely new genetic circuits. While the concept has been around for the better part of a century, the characterization of a growing number of pathways, a growing library of synthetic parts, the decreasing costs of sequencing and DNA synthesis, exciting new technologies, and increased funding have facilitated a boom in the field since just before the turn of the century. The collaboration between biologists and engineers has produced a very broad range of work that have been categorized into

numerous fields and subfields. It is difficult to impossible to encompass the breadth of synthetic biology succinctly, but the manipulation or creation of biological systems can be used for any number of goals [71] such as biofuels [113], disease prevention [135], or mimicking electronic circuitry [96].

From an engineering perspective, the last example is of particular interest. Indeed, a large amount of effort has been placed on seeing what we can currently achieve in biology in terms of computation and the construction of basic elements such as switches [48, 64], oscillators [42, 126], and logic gates [4, 132, 128]. The role of the engineer in synthetic biology is to apply engineering principles to the design and analysis of proposed systems. Building biological systems *de novo* are the best way to demonstrate our successes on this front. However, the engineering principles of standardization, abstraction, and functional composition [43] have proven to be much more difficult to achieve in the realm of biology. In particular, the main shortcoming of the engineering work in synthetic biology is the mismatch between modeling and reality. Unlike the mechanical and digital or analog electronic realms, biology tends to be much more complicated and less controlled. The workings of a single cell, much less an organism, are extremely complicated and any desired engineered behavior could be interfered with in unforeseen and unpredictable ways.

Much work is currently being done to try to overcome these limitations on complexity and construction [21]. Much of the engineering effort in synthetic biology can be divided into the following rough areas: system and interaction modeling [10, 97, 2], simulation tools [14, 85, 67], functional composition [34], engineering robustness [76], parts optimization [41], *in vitro* experiments [75], and continued *de novo* circuit construction. Our work is focused on this last area. We continue to pursue the design of evermore complicated and exciting systems with the understanding that future results in these other areas will continue to make our designs more and more feasible. We aim for a practical approach with projects where we can provide a significant theoretical contribution to synthetic biology while still targeting reasonable experimental implementations.

1.3 Pattern Formation

1.3.1 Motivation

The focus of this work is on pattern formation in gene expression. Like most engineering-focused projects in synthetic biology, we have a biology-based motivating example plus a more grandiose engineering goal. Our motivating example here is developmental biology and the study of morphogenesis. At some point during development, multicellular organisms transform from a clump of identical embryonic cells to an organism with specialized cells that have undergone differentiation. Much of how that happens is still unknown but there must be signaling that allows cells to determine their fate based on their neighbors or global spatial position. While the biologists continue to study the exact mechanisms that cause this to happen in nature, we will study the underlying principles behind pattern formation

to try to contribute to the understanding of patterning systems and to construct tractable model systems.

From the engineering perspective, patterning systems are of particular interest to the outlandish goal of one day engineering multicellularity, that is, synthesizing biological systems on the order of organisms. While this may or may not be achievable, pattern formation is a critical part of one or more of the so-called “enabling technologies” for engineered multicellularity, particularly molecular differentiation [91]. More realistically, pattern formation is an area that is ripe for results and is more interesting and more complex than most of the synthetic genetic circuits achieved thus far exactly because of its multicellular nature. This brings an additional focus on cell-to-cell communication [20] and signaling molecules that will be examined in this work.

The two methods of cell-to-cell communication examined here are quorum sensing [133] and contact-based signaling [57]. Because these mechanisms are very different from each other, they allow for different patterns to emerge. We choose to base our work in *E. coli* for a number of reasons. It is a model organism that is well-sequenced and easy to manipulate (both chromosomally and with plasmids); it lacks much of the complicated background machinery in eukaryotes that might unintentionally interfere with our work; and there exist quorum sensing and contact-based signaling mechanisms that are known to work in *E. coli*. And finally, a demonstration of engineered multicellularity will be that much more meaningful if we can achieve it in a single-celled organism like *E. coli* instead of co-opting an existing multicellular system.

1.3.2 Diffusion-based Signaling

Quorum sensing depends on the secretion or diffusion of a signaling molecule that is sensed by other cells. Because it is diffusion-based, the signaling can reach over relatively long distances and can be effective at a variety of different cell densities. Our work here will be focused on a mathematical phenomenon known as diffusion-driven instability, or Turing patterning [129], where the conditions are such that the interaction between the internal dynamics of the cells and the normalizing effect of diffusion between cells causes the spatial homogenous steady state to be destabilized. This phenomenon has been known for over half a century and yet no synthetic demonstration has been yet achieved in biology. In Chapter 2 we propose a novel architecture that produces spatio-temporal Turing patterning. We work through the design and analysis of this new network and present results on a partial implementation. The primary goal of this chapter is to provide a circuit architecture which can be implemented with relative ease by practitioners and which provides an alternative implementation strategy for reaction-diffusion pattern generation in synthetic multicellular systems. During our partial implementation, we create new synthetic inverters using zinc finger proteins and small RNAs that may prove useful to others in the synthetic biology community.

1.3.3 Contact-based Signaling

Contact-based signaling relies on ligands and receptors that are expressed on the outer membrane of cells. Because of this mechanical restriction, cells must be in close proximity for these signaling pathways to be activated, and thus the signals only reach very short distances but tend to have stronger interactions. In Chapter 3 we study a phenomenon known as lateral inhibition, where a cell can reduce the activity of its neighbors. Because of the competitive inhibition between neighbors, we find that it produces more “on-off” type patterning. We describe a graph theoretic approach to analyzing potentially large contact networks for the existence and stability of these “fine-grained” patterns. In place of an actual contact-based system, we propose a synthetic circuit we call a compartmental lateral inhibition system that used diffusible molecules to demonstrate these types of patterns and work towards its laboratory implementation. The primary goal of this chapter is to provide a theoretical framework in preparation for the future when a synthetic contact-based signaling becomes available. Such systems have been found in nature, but have yet to be co-opted to transport transcriptional factors of our choosing, so the compartmental lateral inhibition system serves as an intermediate demonstration.

Chapter 2

Turing Patterning

2.1 Diffusion-Driven Instability

A particularly well-studied mechanism for pattern formation is diffusion-driven instability, originally proposed by Turing ([129]), where a homogeneous steady state is destabilized in the presence of diffusion.

Attempts have been made to build synthetic gene networks that generate spatio-temporal patterns in gene expression mediated by diffusible signals ([27, 122, 12, 13, 88]). To obtain pattern generation, these efforts have relied either on the external spatio-temporal manipulation of the cell's chemical environment ([27, 122, 88]) or the precise positioning of cells containing different gene networks that secrete or respond to diffusible signals ([12, 13]). To date, there have been no experimental demonstrations of a robust, tunable system which can break symmetry and spontaneously generate predictable gene expression patterns (spatio-temporal inhomogeneities) as in the Turing mechanism. What is specifically lacking in the community is an experimentally tractable model system for studying spontaneous pattern formation. Such a system would catalyze the engineering of complex cellular ensembles, ranging from engineered microbial communities ([12, 88]) to auto-differentiating multicellular systems.

In the synthetic biology community, efforts to achieve spontaneous generation of spatial patterns in gene expression have been centered around networks similar to the one originally proposed by Turing ([129]): two diffusible species (usually termed an *activator* and an *inhibitor*) interact with each other via chemical reactions that produce positive and negative interactions as in Figure 2.1 in Section 2.3. For an appropriate range of kinetic parameters and diffusion constants, these topologies produce spatial or spatio-temporal patterns spontaneously from a homogeneous initial condition perturbed by small variations in concentration due to stochastic effects. However, this type of architecture has proven very difficult to implement using genetic networks because: (a) Turing instability requires that the steady state occur in the linear regime of the activator-inhibitor interactions away from saturation, and severely restricts the parameter range to meet the instability criteria; (b) when using

systems with two diffusible components, either the diffusion constants ([129]) or the uptake rates ([127]) must be sufficiently different to allow unstable spatial modes, and significant differences are difficult to engineer; (c) the addition of intermediate protein steps to Turing’s two-molecule activator-inhibitor model further restricts the parameter set for patterning, and (d) stochasticity plays a significant role in the behavior of these systems, but most analyses rely on continuum partial differential equation (PDE) models, making it difficult to reconcile theoretical predictions with observed experimental results.

Although the activator-inhibitor model is the canonical example of a system demonstrating Turing instability, many other possible network structures exist. Indeed, the essential structural requirement for the emergence of the Turing phenomenon is that the network contain an unstable subsystem, which is stabilized by a feedback loop. The diffusion of molecules participating in this feedback loop then unleashes the inherent instability and allows growth of spatial modes. In the activator-inhibitor network, the activator plays the role of the unstable subsystem and the inhibitor provides the stabilizing feedback. Although it is well known that the Turing mechanism is not restricted to the activator-inhibitor network (e.g. see [36] for Turing instability conditions for general reaction-diffusion models), to the best of our knowledge, no other biologically plausible network has been proposed. Systems that contain more than two species have been studied, but their reactions conform to the essential structure of the activator-inhibitor paradigm ([93]).

Here we break away from the activator-inhibitor model and propose a new network which we call a “quenched oscillator” system. This system uses one diffusible component and an oscillator circuit serving as the unstable subsystem that is quenched by a second feedback loop, as depicted in Figure 2.4 in Section 2.4. To our knowledge, this is the first demonstration that oscillator-driven gene networks can exhibit Turing instability and spatial patterning of gene expression across fields of cells. Moreover, the network can be implemented with a variety of published oscillator circuits ([42, 126]) using known genes and promoters. It is important to stress that the mechanism pursued here – displaying Turing instability – is fundamentally different from the traveling wave trains and spiral waves in diffusively coupled oscillators ([136, 98]). Although we employ an oscillator as a subsystem, the full system is not an oscillator, instead exhibiting a stable steady state as in the Turing mechanism. Moreover, the oscillator subsystem lacks a diffusible molecule. The proposed architecture bears resemblance to the diffusively coupled repressilator model in [47], where a second loop is integrated with the repressilator to incorporate a diffusible molecule. However, their loop does not quench the oscillator, but simply enables communication between cells to ensure synchronization, which is contrary to the pattern formation task studied here.

2.2 Reaction-Diffusion System Analysis

Turing pattern formation arises in reaction-diffusion systems where stability of a steady state in the reaction system does not imply stability of the homogeneous steady state in the presence of diffusion [129]. We will consider the situation where the cells are closely packed

and study the continuous reaction-diffusion system

$$\frac{\partial}{\partial t}c(t, \xi) = f(c(t, \xi)) + D\nabla_{\xi}^2c(t, \xi) \quad (2.1)$$

over the spatial domain Ω with smooth boundary $\partial\Omega$ subject to zero-flux (Neumann) boundary conditions. Here $c(t, \xi)$ is the vector of species concentrations that depends on time t and spatial variable $\xi \in \Omega$, f is the vector field of reaction rates, $D \succeq 0$ is a diagonal matrix of diffusion coefficients, and ∇^2 is the vector Laplacian. The Neumann boundary condition states that $\nabla c(t, \xi) \cdot n(\xi) = 0$, $\forall \xi \in \partial\Omega$, where $n(\xi)$ is the outward normal vector.

We let $J = \frac{\partial f}{\partial c}|_{c=c^*}$ denote the Jacobian linearization about the steady state c^* . The dynamical behavior of the reaction-diffusion system is determined from the matrices $J + \lambda_k D$, where λ_k are the eigenvalues of the Laplacian operator ∇^2 on the given spatial domain, and the subscripts $k = 1, 2, 3, \dots$ denote the wave numbers [25]. For example, on a one-dimensional domain $\Omega = [0, L]$, $\lambda_k = -(\pi k/L)^2$. If the matrix $J + \lambda_k D$ is Hurwitz, then the corresponding wave decays to zero asymptotically in time. If $J + \lambda_k D$ is unstable, then the corresponding wave grows. In Turing's condition for pattern formation, matrix J is stable, implying convergence to steady-state in the absence of diffusion, but $J + \lambda_k D$ is unstable for one or more wave numbers $k \geq 1$, implying the growth of these waves due to diffusion.

2.2.1 Additive D-Stability

A matrix stability concept that rules out Turing pattern formation is *additive D-stability* [49, 70], defined below. We are interested in this concept because necessary conditions for additive D-stability, when negated, serve as sufficient conditions for Turing instability.

Definition 2.1. *A matrix J is called additively D-stable if $J - \bar{D}$ is Hurwitz for all diagonal $\bar{D} \succeq 0$.*

For Turing pattern formation, we need $J + \lambda_k D$ to become unstable for some λ_k . Because $\lambda_k \leq 0$ for Neumann eigenvalues [51] and all diffusion coefficients are non-negative, $J + \lambda_k D$ matches the $J - \bar{D}$ format of additive D-stability with $\bar{D} = -\lambda_k D$. Thus, if J is additively D-stable, Turing pattern formation is not possible.

We will make use of the following necessary condition for additive D-stability and its proof to observe an essential structural property for Turing instability:

Theorem 2.1. *A necessary condition for additive D-stability of the matrix J is that J and all of its principal submatrices be stable.*

Proof. A result similar to Theorem 2.1 has been proven in [104]. Here, we present an alternative proof that makes explicit the structure of \bar{D} that renders $J - \bar{D}$ unstable. Observing this structure will be helpful in designing a network that exhibits Turing instability.

We proceed by contradiction and suppose that $J \in \mathbb{R}^{n \times n}$ contains an unstable principal submatrix J_r of size $r \leq n$, and show that we can recursively construct a diagonal matrix $\bar{D} \succeq 0$ such that $J - \bar{D}$ is unstable.

If $r = n$, then J is an unstable matrix and we can choose $\bar{D} = 0$. If $r < n$, we assume, without loss of generality, that J_r is a leading principal submatrix. Taking the leading principal submatrix of size $r + 1$:

$$J_{r+1} = \begin{bmatrix} J_r & b_r \\ c_r & a_{r+1} \end{bmatrix},$$

we claim that we can find $\bar{D}_{r+1} = \text{diag}\{0, \dots, 0, d_{r+1}\}$ such that $J_{r+1} - \bar{D}_{r+1}$ is unstable. To see this, let $d_{r+1} = 1/\epsilon$ and note from standard singular perturbation arguments [79] that, as $\epsilon \rightarrow 0$, one of the eigenvalues of $J_{r+1} - \bar{D}_{r+1}$ approaches $-\infty$, while the remaining r approach the eigenvalues of J_r . Since J_r is unstable, then by an appropriately large choice of d_{r+1} we can make $J_{r+1} - \bar{D}_{r+1}$ unstable.

Now we can similarly define

$$J_{r+2} = \begin{bmatrix} J_{r+1} & b_{r+1} \\ c_{r+1} & a_{r+2} \end{bmatrix}$$

and $\bar{D}_{r+2} = \text{diag}\{0, \dots, 0, d_{r+1}, d_{r+2}\}$, and render

$$J_{r+2} - \bar{D}_{r+2} = \begin{bmatrix} J_{r+1} - \bar{D}_{r+1} & b_{r+1} \\ c_{r+1} & a_{r+2} - d_{r+2} \end{bmatrix}$$

unstable by an appropriately large choice of d_{r+2} . We can then recursively apply this procedure until we make $J_n - \bar{D}_n = J - \bar{D}$ unstable using $\bar{D} = \text{diag}\{0, \dots, 0, d_{r+1}, \dots, d_n\}$. \square

2.2.2 Conditions for Turing Patterning

We can utilize the above findings into the following conditions for finding biological networks that should produce Turing patterning:

Condition 1: The network must contain an unstable subsystem. [63]

Condition 2: This subsystem must be stabilized by the rest of the system so that J is stable.

Condition 3: The diffusion matrix D must be such that $J + \lambda_k D$ is unstable for some wave number $k \geq 1$.

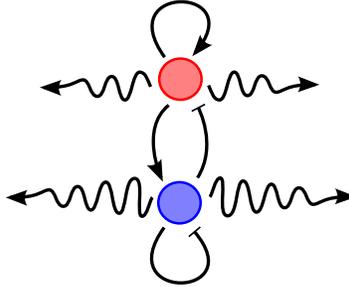


Figure 2.1: The canonical two-component “activator-inhibitor” Turing system. The top component is the activator (in pink) and the bottom component is the inhibitor (in blue), both diffusible.

Since Condition 1 breaks the necessary condition set forth in Theorem 2.1, these conditions are merely sufficient to show Turing patterning and alternative design methodologies to achieve Turing pattern formation may exist. If Conditions 1 and 2 are met, then a matrix D that satisfies Condition 3 can be constructed following the recursive procedure in the proof of Theorem 2.1.

2.3 Activator-Inhibitor Theory

We show that the conditions for Turing patterning discussed above encompass the canonical two-component activator-inhibitor system in Figure 2.1. The linearization and diffusion matrices for this system have the form:

$$J = \begin{bmatrix} j_{11} & j_{12} \\ j_{21} & j_{22} \end{bmatrix}, \quad D = \begin{bmatrix} d_1 & 0 \\ 0 & d_2 \end{bmatrix}, \quad d_i \geq 0,$$

where $j_{11} > 0$ so that component 1 is the unstable “activator,” and $j_{22} < 0$ so that component 2 is the stabilizing “inhibitor.” The activator thus serves as the unstable subsystem to disprove additive D-stability. We assume the spatial domain is $\Omega = [0, \pi]$ with zero-flux boundary conditions such that the eigenfunctions are $\cos(k\xi)$ with eigenvalues $\lambda_k = -k^2$.

Condition 1: This condition is met since we define $j_{11} > 0$.

Condition 2: For stability of the full reaction network, we need:

$$j_{11} + j_{22} < 0 \text{ and } j_{11}j_{22} - j_{12}j_{21} > 0 \tag{2.2}$$

so that $\det(\lambda I - J) = \lambda^2 - (j_{11} + j_{22})\lambda + (j_{11}j_{22} - j_{12}j_{21})$ has both roots in the left half-plane. The first part of (2.2) in combination with Condition 1 confirm that j_{11} and j_{22} must have opposite signs with the negative (stable) quantity of larger magnitude.

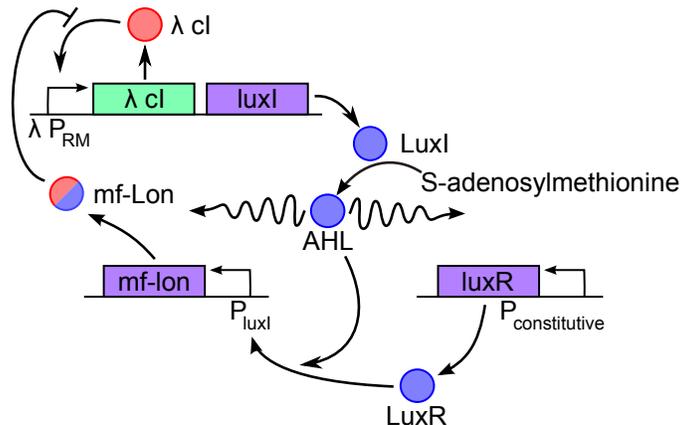


Figure 2.2: One proposed synthetic implementation of an activator-inhibitor network using existing components. The species λcI activates itself as the unstable subsystem and also activates a longer feedback loop using the membrane-diffusible signaling molecule acyl-homoserine lactone (AHL). This system proved experimentally infeasible due to parameter constraints.

This matches our intuition that the feedback loop has to be strong enough to stabilize the overall system.

Condition 3: For diffusion-driven instability of the k^{th} spatial mode we need:

$$(j_{11}j_{22} - j_{12}j_{21}) - k^2(j_{11}d_2 + j_{22}d_1) + k^4d_1d_2 < 0 \quad (2.3)$$

so that $\det(\lambda I - (J + \lambda_k D)) = \lambda^2 - (j_{11} + j_{22} - k^2(d_1 + d_2))\lambda + (j_{11}j_{22} - j_{12}j_{21}) - k^2(j_{11}d_1 + j_{22}d_2) + k^4d_1d_2$ has at least one unstable root. Note in (2.3) that the first quantity is positive by (2.2) and the third quantity is positive by definition, so instability is only possible if $j_{11}d_2 + j_{22}d_1 > 0$. This leads to the further condition that $d_2 > d_1$.

The proof of Theorem 2.1 implies that instability is achieved when $d_2 > 0$ is large enough and this is verified by the condition above. The proof also implies that $d_1 = 0$ is admissible for Turing instability.

2.3.1 Activator-Inhibitor Example

We attempted to find an experimentally feasible activator-inhibitor system, but it proved to be difficult to do with existing components (as discussed in Section 2.3.2). Although many different systems were proposed and tested without success, we will present the system shown in Figure 2.2 as an illustrative example of the difficulties involved.

Here we use the positive auto-regulating loop from λ phage where λCI is a transcriptional activator of the promoter P_{RM} . This is one of the simplest (and smallest) unstable subsystems

available. The feedback loop consists of *Vibrio fischeri* quorum sensing genes *luxI* and *luxR* and disrupts the auto-regulating loop by targeted degradation of λ CI via the Lon protease found in *Mesoplasma florum* (*mf-Lon*) [54]. Although this Turing patterning example did not work out, the use of degradation tags here led directly to the creation of a new bistable switch [64].

We represent the dynamics of this system with the following set of partial differential equations:

$$\begin{aligned}
\frac{\partial}{\partial t} m_C &= V_{P_{RM}} N C \left(\frac{1}{1 + (K_C/p_C)^{n_C}} + \ell_{P_{RM}} \right) - \gamma_m m_C \\
\frac{\partial}{\partial t} p_C &= \epsilon_C m_C - \gamma_C p_C - \frac{k_{cat} p_{Lon}}{1 + K_M/p_C} \\
\frac{\partial}{\partial t} m_I &= V_{P_{RM}} N C \left(\frac{1}{1 + (K_C/p_C)^{n_C}} + \ell_{P_{RM}} \right) - \gamma_m m_I \\
\frac{\partial}{\partial t} p_I &= \epsilon_I m_I - \gamma_I p_I \\
\frac{\partial}{\partial t} p_A &= v_3 p_I - k_f p_A (p_R - p_{RA}) + k_r p_{RA} - \gamma_A p_A + d_{AHL} \nabla^2 p_A \\
\frac{\partial}{\partial t} p_{RA} &= k_f p_A (p_R - p_{RA}) - k_r p_{RA} \\
\frac{\partial}{\partial t} m_{Lon} &= V_{P_{LuxI}} N C \left(\frac{1}{1 + (K_{RA}/p_{RA})^{n_{RA}}} + \ell_{P_{LuxI}} \right) - \gamma_m m_{Lon} \\
\frac{\partial}{\partial t} p_{Lon} &= \epsilon_{Lon} m_{Lon} - \gamma_{Lon} p_{Lon},
\end{aligned} \tag{2.4}$$

where m_i are mRNA concentrations, p_i are protein concentrations, V_i are velocity constants, N is the copy number, K_i are dissociation constants, n_i are Hill coefficients, ℓ_i are leakage rates normalized to V_i , γ_i are degradation rates, and ϵ_i are protein translational rates. The parameters are subscripted according to their corresponding species ($C = [\lambda \text{ cI}]$, $I = [luxI]$, $A = [AHL]$, $R = [luxR]$, $RA = [luxR\text{-AHL complex}]$, $Lon = [mf\text{-lon}]$) except for velocity and leakage constants, which are subscripted by promoter. The variable p_R is the total amount of LuxR protein in the system, which is assumed constant, thus the amount of free LuxR is represented by $p_R - p_{RA}$. The parameter C is the concentration level generated by a single molecule in an *E. coli* cell and d_{AHL} is the diffusion coefficient of AHL. Note that there would also be a fluorescent protein on the P_{RM} operon as the reporter, but since its concentration would be proportional to λ CI and LuxI we omit it from our model.

Jacobian linearization of the reaction equations about the steady-state ($\bar{m}_C, \bar{p}_C, \bar{m}_I, \bar{p}_I$,

$\bar{p}_A, \bar{p}_{RA}, \bar{m}_{Lon}, \bar{p}_{Lon}$) yields:

$$J = \left[\begin{array}{cc|cccccc} -\gamma_m & b_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ \epsilon_C & -a_2 & 0 & 0 & 0 & 0 & 0 & -c_8 \\ \hline 0 & b_2 & -\gamma_m & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \epsilon_I & -\gamma_I & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & v_3 & -a_5 & a_6 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_5 & -a_6 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & b_6 & -\gamma_m & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \epsilon_{Lon} & -\gamma_{Lon} \end{array} \right], \quad (2.5)$$

where:

$$\begin{aligned} c_5 &= k_f(p_R - \bar{p}_{RA}), \quad c_8 = \frac{k_{cat}}{1 + K_M/\bar{p}_C}, \\ b_2 &= V_{PRM}NC \cdot \frac{n_C}{\bar{p}_C} \cdot \frac{(K_C/\bar{p}_C)^{n_C}}{(1 + (K_C/\bar{p}_C)^{n_C})^2}, \quad b_6 = V_{PLuxI}NC \cdot \frac{n_{RA}}{\bar{p}_{RA}} \cdot \frac{(K_{RA}/\bar{p}_{RA})^{n_{RA}}}{(1 + (K_{RA}/\bar{p}_{RA})^{n_{RA}})^2}, \\ a_2 &= \gamma_C + k_{cat} \cdot \frac{\bar{p}_{Lon}}{\bar{p}_C} \cdot \frac{K_M/\bar{p}_C}{(1 + K_M/\bar{p}_C)^2}, \quad a_5 = \gamma_A + c_5, \quad a_6 = k_f\bar{p}_A + k_r. \end{aligned}$$

Note that all of these terms are non-negative.

Condition 1: For the unstable subsystem we require:

$$\gamma_m a_2 - \epsilon_C b_2 < 0 \quad (2.6)$$

so that the characteristic polynomial of the 2×2 upper-left principal submatrix of J , given by $\det(\lambda I - J_u) = (\lambda + \gamma_m)(\lambda + a_2) - \epsilon_C b_2$, has one real root in the right half-plane.

Condition 2: The eigenvalues of J are the roots of:

$$\det(sI - J) = \det(sI - J_u)(s + \gamma_m)^2(s + \gamma_C)(s + \gamma_I)[s^2 + (a_5 + a_6)s + \gamma_A a_6] + F(s + \gamma_m), \quad (2.7)$$

where $F \triangleq v_3 \epsilon_I \epsilon_{Lon} b_2 b_6 c_5 c_8$ characterizes the feedback strength. F must be a value such that all of the eigenvalues of J are stable.

Condition 3: For $\Omega = [0, L]$, $\lambda_k = -(k\pi/L)^2$ for eigenfunctions $\cos(\frac{k\pi}{L}x)$. Here $D = \text{diag}\{0, 0, 0, 0, d_{AHL}, 0, 0, 0\}$. $J + \lambda_k D$ looks identical to J except for the AHL entry of the diagonal, which is now defined as $\hat{a}_5 \triangleq c_5 + \gamma_A - \lambda_k d_{AHL}$. This leads to:

$$\begin{aligned} \det(sI - (J + \lambda_k D)) &= \det(sI - J_u)(s + \gamma_m)^2(s + \gamma_C)(s + \gamma_I)[(s + \hat{a}_5)(s + a_6) - c_5 a_6] \\ &\quad + F(s + \gamma_m), \end{aligned} \quad (2.8)$$

which needs to yield unstable roots for some $k \geq 1$.

Table 2.1: Base parameter values for activator-inhibitor system

Variable	Description	Units	Parameter Value
γ_C	Degradation rate of CI	s^{-1}	2.89×10^{-4}
γ_I	Degradation rate of LuxI	s^{-1}	2.89×10^{-4}
γ_A	Degradation rate of AHL	s^{-1}	2×10^{-3} [33]
γ_{Lon}	Degradation rate of <i>mf</i> -Lon	s^{-1}	7×10^{-4} [59]
γ_m	Degradation rate of mRNA	s^{-1}	2.89×10^{-3} [112]
$V_{P_{RM}}$	mRNA production velocity rate for P_{RM}	s^{-1}	0.3 [109]
$V_{P_{LuxI}}$	mRNA production velocity rate for P_{LuxI}	s^{-1}	0.26 [24]
N	Plasmid copy number for SC101*		5
C	Concentration of a single protein/mRNA in a typical bacterium	M	1.5×10^{-9} [81]
K_C	Disassociation constant of CI to P_{RM}	M	2.5×10^{-8} [56]
K_{RA}	Disassociation constant of LuxR-AHL complex to P_{LuxI}	M	1.5×10^{-9} [24]
K_M	Michaelis constant for <i>mf</i> -Lon	M	3.7×10^{-6} [54]
n_C	Hill coefficient for P_{RM}		2 [56, 16]
n_{RA}	Hill coefficient for P_{LuxI}		2 [24]
$\ell_{P_{RM}}$	mRNA leakage of P_{RM} promoter normalized to velocity rate		1/10 [95]
$\ell_{P_{LuxI}}$	mRNA leakage of P_{LuxI} promoter normalized to velocity rate		1/167 [24]
ϵ_C	Translation rate for CI	s^{-1}	4.5×10^{-5}
ϵ_I	Translation rate for LuxI	s^{-1}	4.5×10^{-5}
ϵ_{Lon}	Translation rate for <i>mf</i> -Lon	s^{-1}	3.5×10^{-5}
v_3	Catalytic rate of LuxI to AHL	s^{-1}	0.01335 [119]
k_{cat}	Catalytic rate of <i>mf</i> -Lon	s^{-1}	0.071 [54]
k_f	Forward rate of LuxR-AHL binding	$M^{-1}s^{-1}$	1×10^9 [137]
k_r	Reverse rate of LuxR-AHL binding	s^{-1}	50 [1, 24, 137]
p_R	Constitutive level of total LuxR in the system	M	1×10^{-8}
d_{AHL}	Diffusion constant of AHL	m^2s^{-1}	1.667×10^{-12} [12]

2.3.2 Activator-Inhibitor Difficulties

The biggest difficulty with these systems is invariably the search for a reasonable set of parameters well within the patterning region. The analysis shown above allows us to direct our parameter search better and to come to conclusions about a system's feasibility. We will start with the base parameter values shown in Table 2.1, which were taken from the literature when possible.

Making the P_{RM} - λ *cI* subsystem in Figure 2.2 unstable is nontrivial. For simplicity, let us first examine Condition 1 above with $k_{cat} = 0$. In this case the instability condition simplifies to $X \triangleq \frac{\epsilon_C b_2}{\gamma_m \gamma_C} < 1$ and the steady-state reduces to:

$$\bar{m}_C = \frac{\gamma_C}{\epsilon_C} \bar{p}_C, \quad \frac{\gamma_m \gamma_C}{\epsilon_C} \bar{p}_C = V_{P_{RM}} N_C \left(\frac{1}{1 + (K_C/\bar{p}_C)^{n_C}} + \ell_{P_{RM}} \right).$$

This means that we can solve for the steady-state value by looking for positive real solutions to the equation:

$$\bar{p}_C^{n_C+1} - \frac{V_{P_{RM}} \epsilon_C}{\gamma_m \gamma_C} (1 + \ell_{P_{RM}}) \bar{p}_C^{n_C} + K_C^{n_C} \bar{p}_C - \ell_{P_{RM}} \frac{V_{P_{RM}} \epsilon_C}{\gamma_m \gamma_C} K_C^{n_C} = 0.$$

For $n_C = 2$, this becomes a cubic equation. Using the base values in Table 2.1 (still with $k_{cat} = 0$), we get a single real solution with $X = 0.0656 < 1$, so the subsystem is stable. Now we wish to vary parameters in order make the subsystem unstable. Substituting the steady-state expression into X , we arrive at the following expression:

$$X = n_C \cdot \frac{(K_C/\bar{p}_C)^{n_C}}{1 + (K_C/\bar{p}_C)^{n_C}} \cdot \frac{1}{1 + \ell_{P_{RM}}(1 + (K_C/\bar{p}_C)^{n_C})}.$$

We can see that the first fraction lies between 0 and 1 and the second fraction lies between 0 and $1/(1 + \ell_{P_{RM}})$, so $X_{\max} = n_C/(1 + \ell_{P_{RM}})$. Since the instability condition is $X > 1$, we need at minimum $n_C > 1$. Also, note that both fractions are functions of K_C/\bar{p}_C and work against each other, i.e. the first fraction is a decreasing function in \bar{p}_C while the second fraction is an increasing function in \bar{p}_C . We can analytically solve for the value of \bar{p}_C to maximize X and we get:

$$\bar{p}_C^* = \sqrt[n_C]{1 + 1/\ell_{P_{RM}}} \quad (2.9)$$

The value of X as a function of \bar{p}_C can be seen in Figure 2.3. Notice that the range of \bar{p}_C for instability is quite small. We can choose a single parameter to vary in order to set \bar{p}_C at a desired value to ensure that $X > 1$. Here we choose ϵ_C since in practice we can alter the ribosome binding site (RBS) of the λ cI gene on the P_{RM} operon. We solve for $\epsilon_C^* = 1.5364 \times 10^{-5}$ to set X to its peak value (while disconnected).

Note that at this new parameter value, all three steady-state values of \bar{p}_C are real and positive. Two of them are stable and only one meets the instability criterion. Multiple equilibria are a common occurrence in activator-inhibitor designs and complicate the analysis, especially as we begin to rely more and more on numerical analysis.

In this particular system design, we can see that even when disconnected, our subsystem is only unstable in a small range of values (Figure 2.3). The threshold for instability also increases in the overall system and the instability condition gets much more complicated:

$$1 + \frac{k_{cat}}{\gamma_C} \cdot \frac{\bar{p}_{Lon}}{\bar{p}_C} \cdot \frac{K_M/\bar{p}_C}{(1 + K_M/\bar{p}_C)^2} < X$$

$$X = n_C \frac{\gamma_I}{\gamma_C} \cdot \frac{\epsilon_C}{\epsilon_I} \cdot \frac{\bar{p}_I}{\bar{p}_C} \cdot \frac{(K_C/\bar{p}_C)^{n_C}}{1 + (K_C/\bar{p}_C)^{n_C}} \cdot \frac{1}{(1 + \ell_{P_{RM}}) + \ell_{P_{RM}}(K_C/\bar{p}_C)^{n_C}}$$

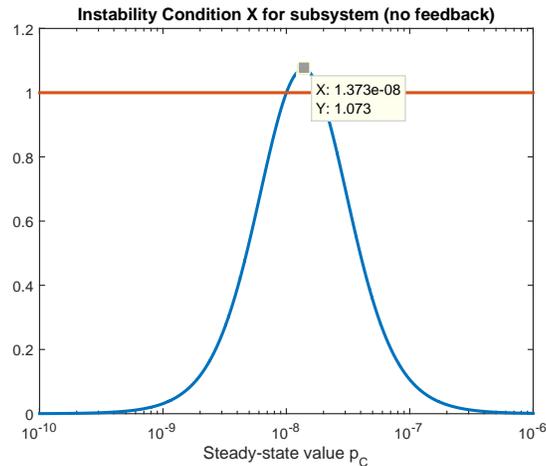


Figure 2.3: How the instability condition X (blue line) varies with respect to the steady-state value \bar{p}_C for the disconnected ($k_{cat} = 0$) subsystem J_u . X must exceed 1 (red line) for the subsystem to be unstable. Shown is the maximum value 1.073 of X given our choices for n_C , K_C , and ℓ_{PRM} . Starting with a base set of parameters, we can choose one (e.g. ϵ_C) to vary in order to get a desired \bar{p}_C .

The threshold is now ≥ 1 and we can no longer easily solve the expression of X for particular parameter values due to the presence of both \bar{p}_C and \bar{p}_I . Similarly, the expression to stabilize the entire system does not take a nice form. This unfortunately leaves us in the situation where we would need to do a parameter search across dozens of variables with little intuition.

This example system exhibits a few problematic characteristics that complicate the analysis. First, our chosen subsystem is only unstable in a small neighborhood around the base set of values, meaning parameter variation in an experimental implementation might easily derail the system. Secondly, the method of feedback directly affects the instability of the subsystem. In this case, the targeted degradation using *mf*-Lon shows up directly in the second diagonal entry $-a_2$, which is part of $\det(sI - J_u)$. We addressed both of these concerns with our later designs.

2.4 Quenched Oscillator Theory

We now present a new network architecture that is capable of generating Turing patterns (Figure 2.4). It consists of a ring oscillator loop (in pink) that serves as the unstable subsystem and a second loop that “quenches” the oscillations and stabilizes the full system. The quenching loop contains a diffusible molecule (in blue), which means that when the corresponding diffusion coefficient is large, the diffusion matrix has the destabilizing structure proposed in the proof of Theorem 2.1.

For stability of the full system, it is essential that the quenching loop have a smaller

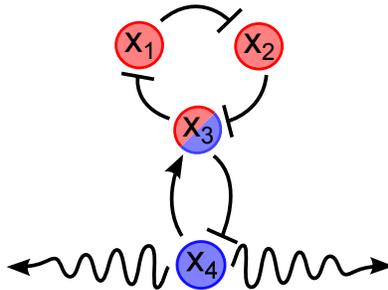


Figure 2.4: The “quenched oscillator” system. The quenching loop with the diffusible molecule (in blue) stabilizes the unstable oscillator loop (in pink). Diffusion then unleashes the inherent instability and allows growth of spatial modes with high wave numbers.

phase lag than the oscillator loop. Smaller phase lag can be achieved with fewer reaction steps or with faster degradation rates in the second loop.

Even though an oscillator has a steady limit cycle, the linearization of its steady state is unstable, thus satisfying Turing Condition 1. In isolation (without diffusion or in a small, enclosed environment), a single cell will approach a steady state over time. However, the embedded oscillator subsystem will approach the limit cycle in the absence of the diffusible molecule. So intuitively we can see that patterning arises from individual cells in a larger ensemble alternating between approaching either the steady state or the limit cycle based on the relative presence or absence of the diffusible molecule in the quenching loop.

2.4.1 Quenched Oscillator Toy Model Analysis

To demonstrate pattern formation with this new architecture, we consider the following “toy” model, which exhibits both an oscillator loop (x_1, x_2, x_3) and a quenching loop (x_3, x_4) . This model represents the set of reactions within a single cell of an ensemble. The spatial component comes from inter-cellular communication, here achieved via diffusion. Species without a spatial component are confined within the membrane of each cell, while diffusible species can travel between cells.

$$\begin{aligned}
 \frac{\partial}{\partial t}x_1 &= \frac{v_3}{1+x_3^p} - x_1 \\
 \frac{\partial}{\partial t}x_2 &= \frac{v_1}{1+x_1^p} - x_2 \\
 \frac{\partial}{\partial t}x_3 &= \frac{v_2}{1+x_2^p} + \frac{v_4x_4^p}{1+\alpha x_4^p} - x_3 \\
 \frac{\partial}{\partial t}x_4 &= \frac{v_3}{1+x_3^p} - x_4 + d_4 \frac{\partial^2 x_4}{\partial \xi^2},
 \end{aligned} \tag{2.10}$$

where the concentrations x_i , $i = 1, \dots, 4$, and all other variables and parameters are non-dimensional. In particular, the time variable t is scaled to bring the degradation constants

(assumed to be identical for each species for simplicity) to one, and the one-dimensional length variable ξ is scaled so that the spatial domain is $\Omega = [0, \pi]$, as would be dictated by the boundaries of a likely experimental environment (microfluidic device). We assume only the fourth species (in blue in Figure 2.4) is diffusible and is subject to zero-flux boundary conditions, meaning there is no diffusion at the ends of the line of cells at $\xi = 0$ and $\xi = \pi$. Because the fourth species is diffusible, this architecture is able to exhibit diffusion-driven instability for a large enough diffusion coefficient d_4 . The choice of spatial domain was made for the sake of simplicity and higher-dimensional domains will only change the form of the Laplacian eigenvalues and modes in the analysis below.

Jacobian linearization of the reaction equations about the steady-state $(\bar{x}_1, \bar{x}_2, \bar{x}_3, \bar{x}_4)$ yields:

$$J = \left[\begin{array}{ccc|c} -1 & 0 & -b_3 & 0 \\ -b_1 & -1 & 0 & 0 \\ 0 & -b_2 & -1 & b_4 \\ \hline 0 & 0 & -b_3 & -1 \end{array} \right], \quad (2.11)$$

where:

$$b_1 = \frac{pv_1\bar{x}_1^{p-1}}{(1 + \bar{x}_1^p)^2}, b_2 = \frac{pv_2\bar{x}_2^{p-1}}{(1 + \bar{x}_2^p)^2}, b_3 = \frac{pv_3\bar{x}_3^{p-1}}{(1 + \bar{x}_3^p)^2}, b_4 = \frac{pv_4\bar{x}_4^{p-1}}{(1 + \alpha\bar{x}_4^p)^2}$$

are all non-negative quantities.

Condition 1: For the oscillator subsystem, we require:

$$B \triangleq b_1b_2b_3 > 8 \quad (2.12)$$

so that the characteristic polynomial of the 3×3 upper-left principal submatrix of J , given by $\det(\lambda I - J_{osc}) = (\lambda + 1)^3 + B$, has a pair of complex conjugate roots in the right half-plane.

Condition 2: For stability of the full reaction network, we need:

$$C \triangleq b_3b_4 > \frac{B - 8}{2} \quad (2.13)$$

so that $\det(\lambda I - J) = (\lambda + 1)[(\lambda + 1)^3 + B + C(\lambda + 1)]$ has all roots in the left half-plane.

Condition 3: For diffusion-driven instability of the k^{th} spatial mode $\cos(k\xi)$, there must be right half-plane roots of the polynomial:

$$\det(\lambda I - (J - k^2 \text{diag}\{0, 0, 0, d_4\})) = (\lambda + 1)[(\lambda + 1)^3 + B + C(\lambda + 1)] + k^2d_4[(\lambda + 1)^3 + B], \quad (2.14)$$

where d_4 is the diffusion coefficient. Indeed, when the product $k^2 d_4$ is sufficiently large, three roots of (2.14) approach those of $(\lambda + 1)^3 + B$, which contain right-half plane roots due to (2.12). This means that the inhomogeneous modes $\cos(k\xi)$ grow in time if $k^2 d_4$ exceeds the threshold for instability of the polynomial (2.14).

The parameters $p = 3$, $v_1 = v_2 = v_3 = 8$, $v_4 = 0.2$, and $\alpha = 0.1$ in the system (2.10) satisfy conditions (2.12)-(2.13) with $B = 10.0398$, $C = 1.6928$, and the polynomial (2.14) becomes unstable when $k^2 d_4 > 5.6397$. PDE Simulations with $d_4 = 4$ indeed exhibit growth of the spatial inhomogeneity when the steady state is perturbed by adding the second wave ($k = 2$) with amplitude steady state $\pm 33\%$ peak-to-peak to $x_1(0, \xi)$ (Figure 2.5). The PDE system does not include noise, so a perturbation must manually be added to the system for cells to leave steady state. This Turing behavior is contrasted to the decay of the initial inhomogeneity for wave numbers below the instability threshold ($k = 1$, $d_4 = 4$ in Figure 2.6) and in the absence of diffusion ($k = 2$, $d_4 = 0$ in Figure 2.7).

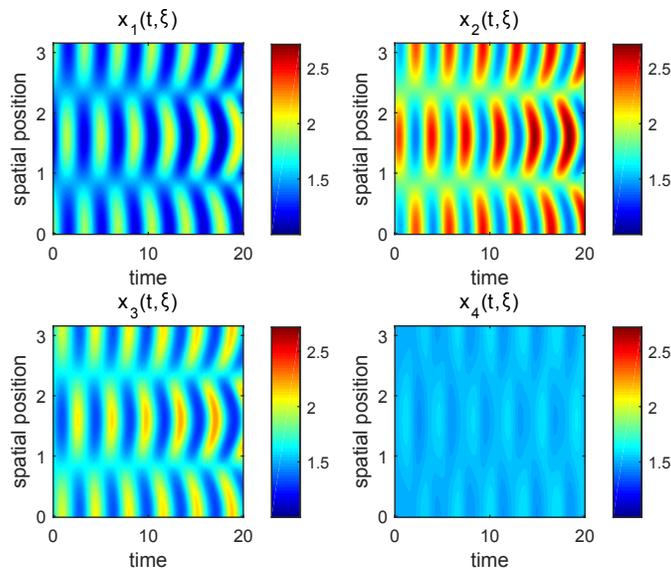


Figure 2.5: Solution of (2.10) on $\Omega = [0, \pi]$ with diffusion and growth. Using parameters $p = 3$, $v_1 = v_2 = v_3 = 8$, $v_4 = 0.2$, $\alpha = 0.1$. Here $d_4 = 4$ and $k = 2$ (wavelength π). Perturbation in x_1 of amplitude steady state $\pm 33\%$ peak-to-peak causes the inhomogeneity to grow as $k^2 d_4 = 16 > d_{thresh} = 5.6397$. Colorbar scale of concentrations normalized across all species so the growing fluctuations in $x_4(t, \xi)$ are difficult to see because of their small amplitudes.

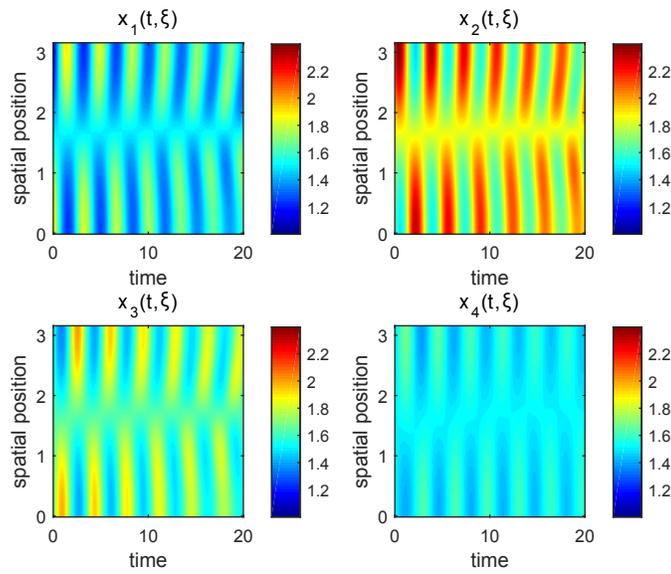


Figure 2.6: Solution of (2.10) on $\Omega = [0, \pi]$ with diffusion and decay. Using parameters $p = 3$, $v_1 = v_2 = v_3 = 8$, $v_4 = 0.2$, $\alpha = 0.1$. Here $d_4 = 4$ and $k = 1$ (wavelength 2π). Perturbation in x_1 of amplitude steady state $\pm 33\%$ peak-to-peak decays towards the steady state as $k^2 d_4 = 4 < d_{thresh} = 5.6397$.

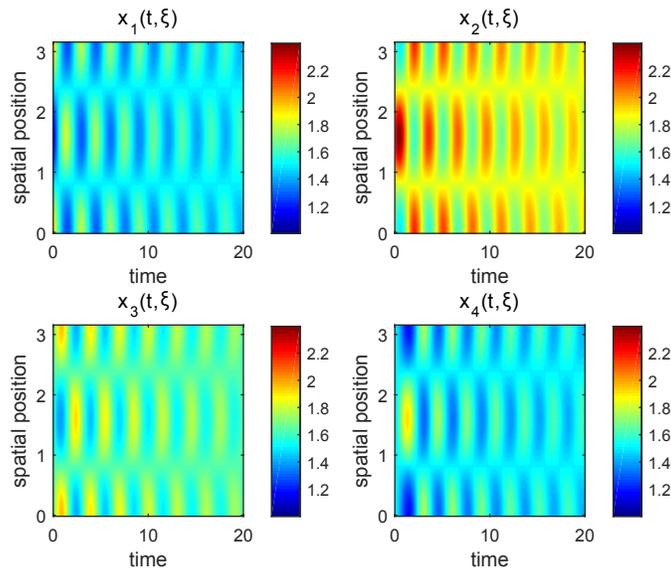


Figure 2.7: Solution of (2.10) on $\Omega = [0, \pi]$ without diffusion. Using parameters $p = 3$, $v_1 = v_2 = v_3 = 8$, $v_4 = 0.2$, $\alpha = 0.1$. Here $d_4 = 0$ and $k = 2$ (wavelength π). Perturbation in x_1 of amplitude steady state $\pm 33\%$ peak-to-peak decays towards the steady state as all cells are stable.

2.4.2 Discussion of the Quenched Oscillator Toy Model

The quenching loop needs to have a negative feedback structure, so interactions involving x_4 can be either x_3 inhibiting x_4 , which activates x_3 (as shown in Figure 2.4), or x_3 activating x_4 , which inhibits x_3 . The one shown was chosen to simplify the mathematics (reuse of b_3) as well as for the practical reason that in this case we can use the same promoter to produce x_1 and x_4 .

To see why roots of (2.14) approach those of the oscillator, we can rewrite it as follows:

$$1 + k^2 d_4 \frac{[(\lambda + 1)^3 + B]}{(\lambda + 1)[(\lambda + 1)^3 + B + C(\lambda + 1)]} = 1 + k^2 d_4 \frac{\det(\lambda I - J_{osc})}{\det(\lambda I - J)} = 0,$$

which is in the form of the standard negative feedback system shown in Figure 2.8. As the feedback gain $k^2 d_4$ increases, three poles of the system given by $\det(\lambda I - J)$ approach the zeros of the system given by $\det(\lambda I - J_{osc})$. If (2.12) and (2.13) are satisfied, then the system eigenvalues without diffusion all lie in the left half-plane and two of them approach values in the right half-plane as $k^2 d_4 \rightarrow \infty$, as shown in Figure 2.9. We will call the threshold for instability where system eigenvalues cross the imaginary axis d_{thresh} , which will determine the minimum wave number for instability for a particular parameter set.

In more general terms on the 1-D spatial domain $\Omega = [0, L]$, Condition 3 is met when:

$$(k\pi/L)^2 d_4 > d_{thresh}. \quad (2.15)$$

This implies that for diffusion-driven patterning with this quenched oscillator system, we need a large diffusion coefficient, a large wave number, or a small spatial domain. This expression can be rewritten in terms of the spatial wavelength ω_x as:

$$\omega_x^2 < 4\pi^2 d_4 / d_{thresh}. \quad (2.16)$$

This maximum unstable wavelength is a convenient formulation because it applies to any chosen spatial domain size.

In addition, we know that the system eigenvalues approach values with a non-zero imaginary component, meaning that any patterning will have an oscillatory component. While this may not match what many biologists expect of Turing patterning (e.g. Figure 2A in

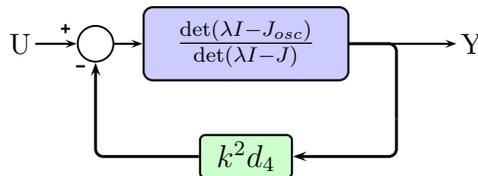


Figure 2.8: Equation 2.14 shown in standard negative feedback form with feedback gain $k^2 d_4$. As the feedback gain increases, three of the four poles of the plant given initially by $\det(\lambda I - J)$ will approach the three zeros of the plant given by $\det(\lambda I - J_{osc})$.

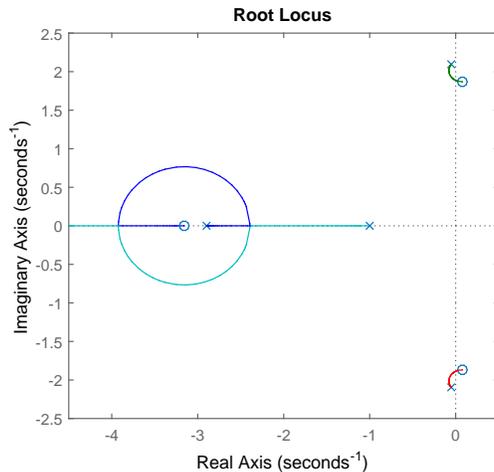


Figure 2.9: Bifurcation diagram for parameters $n = 3$, $v_1 = v_2 = v_3 = 8$, $v_4 = 0.2$, $\alpha = 0.1$. The positions of the system eigenvalues without diffusion are shown with \times 's and their limits as the gain $k^2 d_4 \rightarrow \infty$ are shown with o 's. For this parameter set, Turing patterning is achieved once two eigenvalues cross the imaginary axis when $k^2 d_4 > d_{thresh} = 5.6397$.

[80]), it fits with the body of literature that studies the relationship between Hopf bifurcations and Turing bifurcations [86]. As was shown in that paper, the quenched oscillator system is the case where a spatio-temporally oscillating solution is obtained in the presence of only a Turing bifurcation.

2.5 Quenched Oscillator Implementation with an Existing Oscillator

We first proposed a network that could be synthesized from existing components. Consider the system of two interconnected loops shown in Figure 2.10. The first (top) loop is the repressilator [42], which is a ring oscillator comprised of three pairs of transcriptional repressors (TetR, λ cI, LacI) and promoters ($P_{LtetO-1}$, λP_R , $P_{LlacO-1}$), which match up with the three-component oscillator of the toy model (x_1 - x_2 - x_3). The second (bottom) feedback loop consists of *V. fischeri* quorum sensing genes *luxI* and *luxR*. The *luxI* gene is regulated by the $P_{LtetO-1}$ promoter, and is transcribed in the absence of TetR. LuxI is the autoinducer synthase that catalyzes the formation of the membrane-diffusible signaling molecule 3-oxohexanoyl-homoserine lactone (3OC6HSL), which is a well-known member of a class of signaling molecules called *N*-acyl homoserine lactones (AHLs). For simplicity here we will refer to 3OC6HSL as AHL. AHL binds to the constitutively produced receptor protein, LuxR. The LuxR-AHL complex forms a homodimer that binds to the P_{LuxI} promoter and activates transcription. TetR production closes the second loop by repressing the second

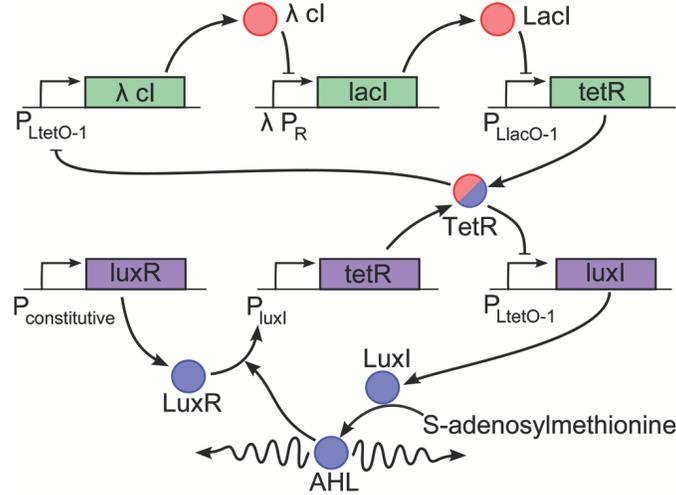


Figure 2.10: Our first proposed synthetic implementation of the network in Figure 2.4 using existing components. Two feedback loops are interconnected by shared production and sensing of the transcriptional repressor *tetR*. The genes in the first feedback loop are depicted in green and the genes in the second feedback loop are shown in purple. The second loop contains the membrane-diffusible signaling molecule 3-oxohexanoyl-homoserine lactone (3OC6HSL), which we refer to simply as AHL.

$P_{LtetO-1}$ promoter. This quenching loop is much longer than that of the toy model (x_3-x_4), but still contains a single diffusible molecule and we ensure that it has smaller phase delay than the oscillator loop by using faster degradation rates. Even though the bottom loop has a single inhibitory interaction, this loop does not oscillate because the phase delay is small. The two loops interact through TetR and the first loop ceases to oscillate in the presence of the second loop.

2.5.1 Quenched Oscillator PDE Model and Analysis

We represent the dynamics of the network in Figure 2.10 with the following set of partial differential equations:

$$\begin{aligned} \frac{\partial}{\partial t} m_C &= V_{P_{LtetO-1}} N_C C \left(\frac{1}{1 + (p_T/K_T)^{n_T}} + \ell_{P_{LtetO-1}} \right) - \gamma_{mO} m_C \\ \frac{\partial}{\partial t} p_C &= \epsilon_C m_C - \gamma_C p_C \\ \frac{\partial}{\partial t} m_{TO} &= V_{P_{LlacO-1}} N_{TO} C \left(\frac{1}{1 + (p_L/K_L)^{n_L}} + \ell_{P_{LlacO-1}} \right) - \gamma_{mO} m_{TO} \\ \frac{\partial}{\partial t} p_T &= \epsilon_{TO} m_{TO} + \epsilon_{TQ} m_{TQ} - \gamma_T p_T \end{aligned}$$

$$\begin{aligned}
\frac{\partial}{\partial t} m_L &= V_{P_R} N_L C \left(\frac{1}{1 + (p_C/K_C)^{n_C}} + \ell_{P_R} \right) - \gamma_{mO} m_L \\
\frac{\partial}{\partial t} p_L &= \epsilon_L m_L - \gamma_L p_L \\
\frac{\partial}{\partial t} m_I &= V_{P_{LtetO-1}} N_I C \left(\frac{1}{1 + (p_T/K_T)^{n_T}} + \ell_{P_{LtetO-1}} \right) - \gamma_{mQ} m_I \\
\frac{\partial}{\partial t} p_I &= \epsilon_I m_I - \gamma_I p_I \\
\frac{\partial}{\partial t} A &= v_3 p_I - k_f A (p_R - p_{RA}) + k_r p_{RA} - \gamma_A A + d_{AHL} \nabla^2 A \\
\frac{\partial}{\partial t} p_{RA} &= k_f A (p_R - p_{RA}) - k_r p_{RA} \\
\frac{\partial}{\partial t} m_{TQ} &= V_{P_{LuxI}} N_{TQ} C \left(\frac{1}{1 + (K_{RA}/p_{RA})^{n_{RA}}} + \ell_{P_{LuxI}} \right) - \gamma_{mQ} m_{TQ}, \tag{2.17}
\end{aligned}$$

where m_i are mRNA concentrations, p_i are protein concentrations, V_i are velocity constants, N_i are copy numbers, K_i are dissociation constants, n_i are Hill coefficients, ℓ_i are leakage rates normalized to V_i , γ_i are degradation rates, and ϵ_i are protein translational rates. The parameters are subscripted according to their corresponding species ($C=[cI]$, $T=[tetR]$, $L=[lacI]$, $I=[luxI]$, $A=[AHL]$, $R=[luxR]$, $RA=[luxR\text{-}AHL \text{ complex}]$) except for velocity and leakage constants, which are subscripted by promoter, and copy numbers, which are subscripted by the gene being transcribed. The concentration of the mRNA for *tetR* is split into those produced by the oscillator loop (O) and the quenching loop (Q). The variable p_R is the total amount of LuxR protein in the system, which is assumed constant, thus the amount of free LuxR is represented by $p_R - p_{RA}$. The parameter C is the concentration level generated by a single molecule in an *E. coli* cell and d_{AHL} is the diffusion coefficient of AHL. We take $\gamma_C = \gamma_T = \gamma_L \triangleq \gamma_p$. The system is subject to zero-flux boundary conditions on the one-dimensional spatial domain $\Omega = [0, L]$.

The linearization of the reaction terms in (2.17) at steady state ($\bar{m}_C, \bar{p}_C, \bar{m}_{TO}, \bar{p}_T, \bar{m}_L, \bar{p}_L, \bar{m}_I, \bar{p}_I, \bar{A}, \bar{p}_{RA}, \bar{m}_{TQ}$) yields the Jacobian matrix:

$$J = \left[\begin{array}{c|c} J_{11} & J_{12} \\ \hline J_{21} & J_{22} \end{array} \right] = \left[\begin{array}{cccccc|cccccc} -\gamma_{mO} & 0 & 0 & -b_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \epsilon_C & -\gamma_C & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\gamma_{mO} & 0 & 0 & -b_6 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \epsilon_{TO} & -\gamma_T & 0 & 0 & 0 & 0 & 0 & 0 & \epsilon_{TQ} \\ 0 & -b_2 & 0 & 0 & -\gamma_{mO} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \epsilon_L & -\gamma_L & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & -b_{42} & 0 & 0 & -\gamma_{mQ} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \epsilon_I & -\gamma_I & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & v_3 & -a_9 & a_{10} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_9 & -a_{10} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & b_{10} & -\gamma_{mQ} \end{array} \right],$$

where we use the parameters $\alpha_C \triangleq \bar{p}_C/K_C$, $\alpha_T \triangleq \bar{p}_T/K_T$, $\alpha_L \triangleq \bar{p}_L/K_L$, and $\alpha_A \triangleq k_f \bar{A}/k_r$ to obtain the off-diagonal entries:

$$\begin{aligned}
c_9 &= \frac{k_f p_R}{1 + \alpha_A}, \\
a_9 &= c_9 + \gamma_A, \\
a_{10} &= k_r(1 + \alpha_A), \\
b_2 &= V_{P_R} V_L C \frac{n_C \alpha_C^{(n_C-1)}}{K_C(1 + \alpha_C^{n_C})^2}, \\
b_4 &= V_{P_{LtetO-1}} N_C C \frac{n_T \alpha_T^{(n_T-1)}}{K_T(1 + \alpha_T^{n_T})^2}, \\
b_{42} &= V_{P_{LtetO-1}} N_I C \frac{n_T \alpha_T^{(n_T-1)}}{K_T(1 + \alpha_T^{n_T})^2}, \\
b_6 &= V_{P_{LlacO-1}} N_{TO} C \frac{n_L \alpha_L^{(n_L-1)}}{K_L(1 + \alpha_L^{n_L})^2}, \\
b_{10} &= V_{P_{LuxI}} N_{TQ} C \frac{n_{RA} \left(\frac{K_{RA}}{p_R}\right) \left(\frac{K_{RA}}{p_R} \frac{1 + \alpha_A}{\alpha_A}\right)^{n_{RA}}}{p_R \left(1 + \left(\frac{K_{RA}}{p_R} \frac{1 + \alpha_A}{\alpha_A}\right)^{n_{RA}}\right)^2}.
\end{aligned}$$

We note that J_{11} is the linearization matrix for the first loop, which corresponds to the standard repressilator system (*cI-lacI-tetR*).

Condition 1: The eigenvalues of J_{11} are the roots of:

$$\det(sI - J_{11}) = (s + \gamma_{mO})^3 (s + \gamma_p)^3 + \epsilon_C \epsilon_{TO} \epsilon_L b_2 b_4 b_6.$$

It can be shown [42] that instability of J_{11} is achieved when:

$$\frac{(\beta + 1)^2}{\beta} < \frac{3X^2}{4 + 2X} \quad (2.18)$$

where $\beta \triangleq \gamma_p/\gamma_{mO}$ and $X \triangleq -\frac{1}{\gamma_p \gamma_{mO}} \sqrt[3]{\epsilon_C \epsilon_{TO} \epsilon_L b_2 b_4 b_6}$.

Substituting steady-state expressions and rearranging, we arrive at the following expression for X :

$$\begin{aligned}
X^3 &= -n_C n_T n_L \frac{\alpha_{RA}}{1 + \alpha_{RA}} \frac{\alpha_C^{n_C}}{1 + \alpha_C^{n_C}} \frac{1}{1 + \ell_{P_R} (1 + \alpha_C^{n_C})} \frac{\alpha_T^{n_T}}{1 + \alpha_T^{n_T}} \frac{1}{1 + \ell_{P_{LtetO-1}} (1 + \alpha_T^{n_T})} \\
&\quad \times \frac{\alpha_L^{n_L}}{1 + \alpha_L^{n_L}} \frac{1}{1 + \ell_{P_{LlacO-1}} (1 + \alpha_L^{n_L})}, \quad (2.19)
\end{aligned}$$

where the additional variable $\alpha_{RA} \geq 0$ is defined by the relation:

$$\frac{1}{1 + \alpha_{RA}} \triangleq \frac{\epsilon_{TQ} V_{P_{LuxI}} N_{TQ} C}{\gamma_p \gamma_{mQ} \alpha_T K_T} \left(\frac{1}{1 + \left(\frac{K_{RA}}{p_R} \frac{1 + \alpha_A}{\alpha_A}\right)^{n_{RA}}} + \ell_{P_{LuxI}} \right).$$

Condition 2: The eigenvalues of J are the roots of:

$$\begin{aligned} \det(sI - J) &= \det(sI - J_{11})(s + \gamma_I)(s + \gamma_{mQ})^2[(s + a_9)(s + a_{10}) - c_9 a_{10}] \\ &\quad + F(s + \gamma_{mO})^3(s + \gamma_p)^2, \end{aligned} \quad (2.20)$$

where $F \triangleq v_3 \epsilon_I \epsilon_T Q c_9 b_{42} b_{10}$ characterizes the feedback strength. F must be a value such that all of the eigenvalues of J are stable.

Substituting steady-state expressions and rearranging, we arrive at the following expression for F :

$$\begin{aligned} F &= \gamma_T \gamma_I \gamma_A \gamma_{mQ}^2 k_r n_T n_{RA} \frac{1}{1 + \alpha_{RA}} \frac{\left(\frac{K_{RA}}{p_R} \frac{1 + \alpha_A}{\alpha_A}\right)^{n_{RA}}}{1 + \left(\frac{K_{RA}}{p_R} \frac{1 + \alpha_A}{\alpha_A}\right)^{n_{RA}}} \frac{1}{1 + \ell_{P_{LuxI}} \left(1 + \left(\frac{K_{RA}}{p_R} \frac{1 + \alpha_A}{\alpha_A}\right)^{n_{RA}}\right)} \\ &\quad \times \frac{\alpha_T^{n_T}}{1 + \alpha_T^{n_T}} \frac{1}{1 + \ell_{P_{LtetO-1}} (1 + \alpha_T^{n_T})}. \end{aligned} \quad (2.21)$$

Condition 3: For $\Omega = [0, L]$, $\lambda_k = -(k\pi/L)^2$ for eigenfunctions $\cos(\frac{k\pi}{L}x)$. Here $D = \text{diag}\{0, 0, 0, 0, 0, 0, 0, 0, d_{AHL}, 0, 0\}$. $J + \lambda_k D$ looks identical to J except for the AHL entry of the diagonal, which is now defined as $-\hat{a}_9 \triangleq -c_9 - \gamma_A + \lambda_k d_{AHL}$. This leads to:

$$\begin{aligned} \det(sI - (J + \lambda_k D)) &= \det(sI - J_{11})(s + \gamma_I)(s + \gamma_{mQ})^2[(s + \hat{a}_9)(s + a_{10}) - c_9 a_{10}] \\ &\quad + F(s + \gamma_{mO})^3(s + \gamma_p)^2, \end{aligned} \quad (2.22)$$

which yields unstable roots for large enough $|\lambda_k d_{AHL}|$.

2.5.2 Model Parameter Selection and PDE Simulation

Taking some parameters to be fixed (e.g. those that are difficult to manipulate experimentally), we can use the above results to guide our choices for mutable values to adjust $|X|$ and F such that Turing Conditions 1 and 2 are satisfied and experimentally-reasonable wave numbers become unstable. We reduce the number of parameters to look at in the system by treating the steady state values α_C , α_T , α_L , α_A , and α_{RA} as new parameters to be adjusted such that $|X|$ is large enough for instability of J_{11} and F is large enough to stabilize the over-all system, then solve for the original set of parameters. For example, $\alpha_C^* = \sqrt[2n]{1 + 1/\ell_{P_R}}$ will maximize $|X|$ for that variable and increasing k_r and k_f proportionally will increase F without affecting the steady-state values.

To show the viability of this system for experimental implementation, we modeled the system behavior using parameter values from the literature that fit the constraints found in the analysis (“Value for PDE Simulation (Parameter Set 1)” column of Table 2.2). Accepted literature values were used whenever possible. When literature values were not available,

acceptable estimates were made in order to keep the analysis within the realm of experimental plausibility. The parameter values can be split to three groups: known, estimated, and prescribed values. “Known values” are parameters with measured literature values, including protein-DNA dissociation constants, mRNA production rates for given promoters, and promoter leakage levels. These values are considered fixed and cannot be readily changed.

“Estimated values” can be experimentally changed within reasonable ranges, which are also given in Table 2.2. Half-lives of proteins can be anywhere between minutes to hours and can be controlled by adding or changing their *ssrA* tags [66, 84]. Most proteins in our system have half-lives in the tens of minutes. Half-lives of mRNA usually fall in the order of minutes [112] and can be altered by changing the secondary structure of the mRNA. The half-life of AHL is measured to be approximately 24-48 hours [45], but can be sped up to the order of minutes in the presence of the enzyme AiiA [33]. In this study, a steady-state concentration of AiiA is assumed to set the AHL half-life at 15 minutes. The copy numbers for the plasmids was assumed to be low, so a value of 5 is used to represent the averaged plasmid copy number for the complete field of cells.

The “prescribed” parameters consist of the value of constitutively-produced LuxR protein, assumed constant, and the translation rates of mRNA. The steady-state value of a protein can be fixed by adjusting its production rate and degradation rate and should fall in the range between 1 nM to 1 mM in a cell. The translation rate of proteins can be sped up or slowed down by changing the ribosome binding site. This generally yields about 10 proteins per mRNA transcript [42, 74]. In this analysis, the protein translation rates were the only parameters which were readily changed. Finding protein translation rates for this system to meet the Turing conditions for patterning was the big challenge of this analysis. The known and estimated values created tight constraints for the translation rates.

Due to computational constraints on the stochastic simulations, we restricted our spatial domain to a line of 100 cells ($100 \mu m$). The literature value for diffusion allowed molecules of AHL to traverse this entire spatial domain very rapidly, obscuring the patterning visually, so we reduced the diffusion constant. Experimentally the diffusion constant will effectively change based on the medium of diffusion, but the same effect can also be achieved by increasing the spatial domain.

Expected steady-state values and system characteristics can also be found in Tables 2.3 and 2.4. We ran PDE simulations in MATLAB with and without AHL diffusion using an initial perturbation in p_C of amplitude steady state $\pm 33\%$ peak-to-peak and wavelength $100 \mu m$, which was predicted to be unstable (Figures 2.11 and 2.13). The imprinted wave grows with diffusion and it decays without diffusion (Figure 2.12), exhibiting similar behavior to that of the toy model (Figures 2.5-2.7).

While the simulation results produce spatio-temporal patterning as desired, the expected experimental behavior will be impacted by stochastic properties that stem from concentrations in our system approaching a few molecules per cell. Taking the concentration of a

*Estimated from [1, 24, 137].

†Estimated from [12].

Table 2.2: Acceptable ranges and chosen parameter values used for simulations

Parameter	Range	Value for PDE Simulation (Parameter Set 1)	Value for Stochastic Simulation (Parameter Set 2)
γ_C	$\gamma_x \leq 1 \times 10^{-2} \text{ s}^{-1}$ [84]	$2.89 \times 10^{-4} \text{ s}^{-1}$	$2.89 \times 10^{-3} \text{ s}^{-1}$
γ_T		$2.89 \times 10^{-4} \text{ s}^{-1}$	$2.89 \times 10^{-3} \text{ s}^{-1}$
γ_L		$2.89 \times 10^{-4} \text{ s}^{-1}$	$2.89 \times 10^{-3} \text{ s}^{-1}$
γ_I		$1.16 \times 10^{-3} \text{ s}^{-1}$	$2.89 \times 10^{-2} \text{ s}^{-1}$
γ_A	$\gamma_A \sim 2 \times 10^{-3} \text{ s}^{-1}$ [33]	$7.70 \times 10^{-4} \text{ s}^{-1}$	$2.89 \times 10^{-2} \text{ s}^{-1}$
γ_{mO}	$3.5 \times 10^{-4} \leq \gamma_m$ $\gamma_m \leq 2.3 \times 10^{-2} \text{ s}^{-1}$ [112]	$5.78 \times 10^{-4} \text{ s}^{-1}$	$5.78 \times 10^{-3} \text{ s}^{-1}$
γ_{mQ}		$5.78 \times 10^{-3} \text{ s}^{-1}$	$5.78 \times 10^{-2} \text{ s}^{-1}$
$V_{PLtetO-1}$	$V_x \leq 4 \text{ s}^{-1}$ [68]	0.3 s^{-1} [90]	4 s^{-1}
$V_{PLlacO-1}$		0.23 s^{-1} [90]	4 s^{-1}
V_{PR}		0.06 s^{-1} [109]	4 s^{-1}
V_{PLuxI}		0.26 s^{-1} [24]	4 s^{-1}
N_C	$1 \leq N_x \leq 30$	5	4
N_{TO}		5	4
N_L		5	4
N_I		5	4
N_{TQ}		5	4
C		$1.5 \times 10^{-9} \text{ M}$ [81]	$1.5 \times 10^{-9} \text{ M}$
K_C	$1 \times 10^{-13} \leq K_x$ $K_x \leq 1 \times 10^{-7} \text{ M}$ [115, 56]	$2.5 \times 10^{-8} \text{ M}$ [56]	$3 \times 10^{-9} \text{ M}$
K_T		$1.786 \times 10^{-10} \text{ M}$ [92]	$3 \times 10^{-9} \text{ M}$
K_L		$1 \times 10^{-13} \text{ M}$ [115]	$3 \times 10^{-9} \text{ M}$
K_{RA}		$1.5 \times 10^{-9} \text{ M}$ [24]	$6.75 \times 10^{-9} \text{ M}$
n_C		2 [56, 16]	2
n_T		2 [16]	2
n_L		2 [16]	2
n_{RA}		2 [24]	2
$\ell_{PLtetO-1}$		1/5050 [90]	1/1000
$\ell_{PLlacO-1}$		1/620 [90]	1/1000
ℓ_{PR}		1/131 [37]	1/1000
ℓ_{PLuxI}		1/167 [24]	1/1000
ϵ_C	$\epsilon_x \leq 5.78 \times 10^{-3} \text{ s}^{-1}$ [74]	$4.470 \times 10^{-4} \text{ s}^{-1}$	$3.712 \times 10^{-4} \text{ s}^{-1}$
ϵ_{TO}		$2.269 \times 10^{-6} \text{ s}^{-1}$	$1.856 \times 10^{-4} \text{ s}^{-1}$
ϵ_L		$2.113 \times 10^{-9} \text{ s}^{-1}$	$3.712 \times 10^{-4} \text{ s}^{-1}$
ϵ_I		$2.655 \times 10^{-5} \text{ s}^{-1}$	$3.815 \times 10^{-2} \text{ s}^{-1}$
ϵ_{TQ}		$6.224 \times 10^{-6} \text{ s}^{-1}$	$1.856 \times 10^{-3} \text{ s}^{-1}$
v_3		0.01335 s^{-1} [119]	0.01 s^{-1}
k_f	$1 \times 10^4 \leq k_f \leq 5 \times 10^9$ [137]	$1 \times 10^9 \text{ M}^{-1} \text{ s}^{-1}$ [137]	$5 \times 10^9 \text{ M}^{-1} \text{ s}^{-1}$
k_r	$k_r \geq 5 \times 10^{-4} \text{ s}^{-1*}$	50 s^{-1*}	15 s^{-1}
PR		$1 \times 10^{-8} \text{ M}$	$1.8 \times 10^{-8} \text{ M}$
d_{AHL}		$1.667 \times 10^{-12} \text{ m}^2 \text{ s}^{-1 \dagger}$	$1.667 \times 10^{-12} \text{ m}^2 \text{ s}^{-1}$

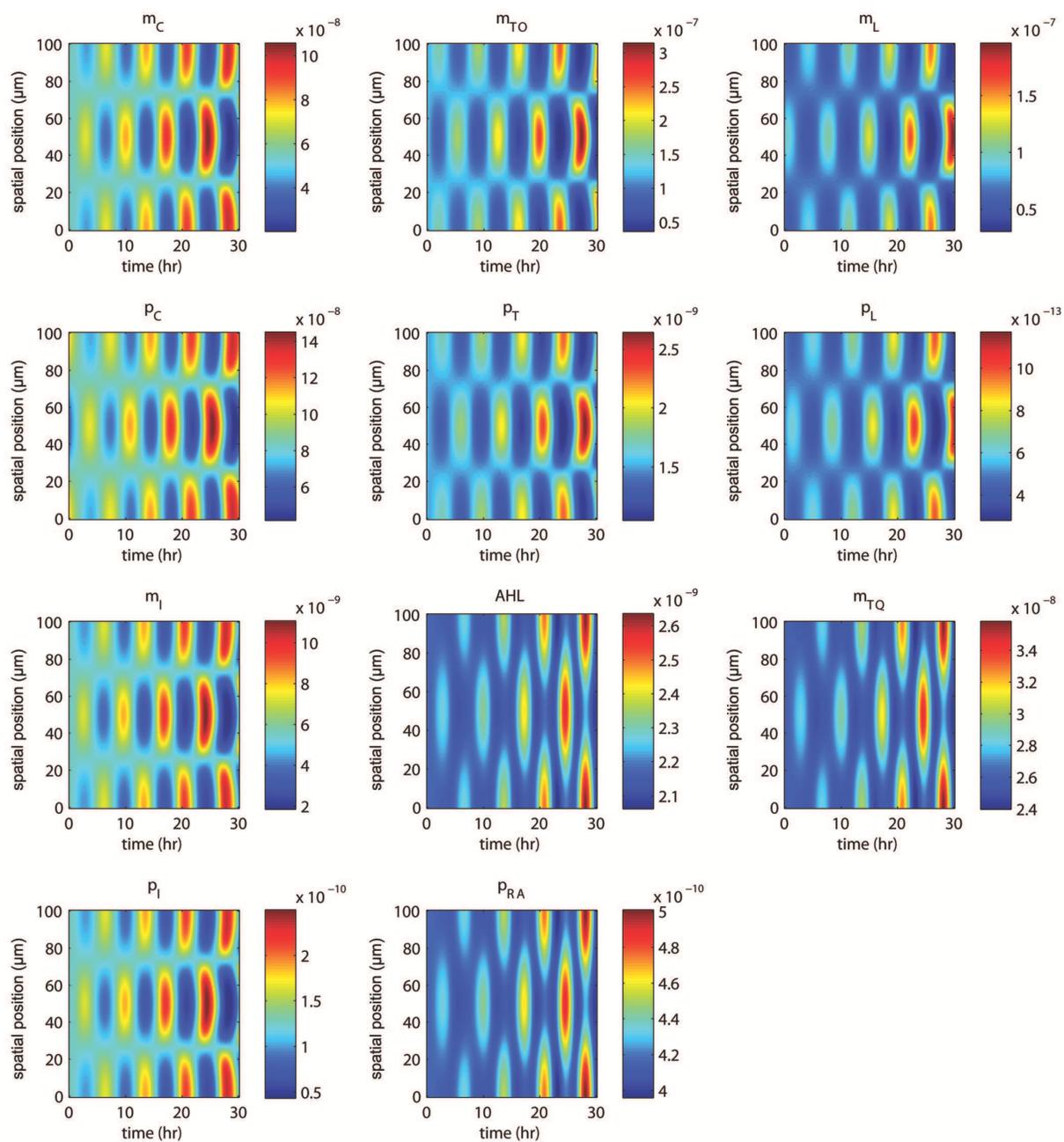


Figure 2.11: PDE simulation results for Parameter Set 1 in line of cells with diffusion and growth. Here $d_{AHL} = 1.667 \times 10^{-12} \text{ m}^2/\text{s}$, $L = 100 \mu\text{m}$, and $k = 2$ (wavelength $100 \mu\text{m}$). Concentrations (colorbar) given in M. Perturbation in p_C of amplitude steady state $\pm 33\%$ peak-to-peak leads to growth of the inhomogeneity.

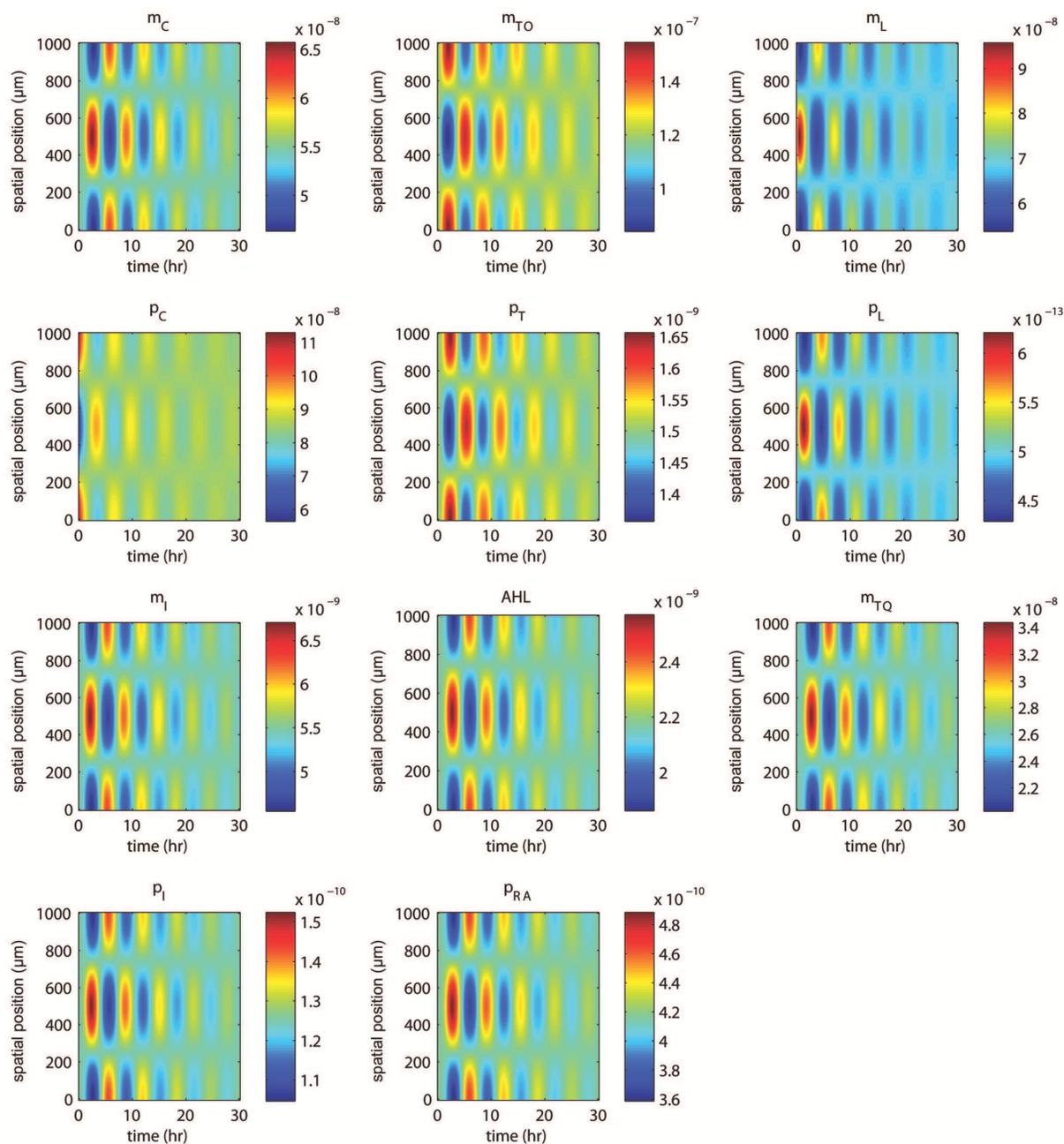


Figure 2.12: PDE simulation results for Parameter Set 1 in line of cells with diffusion and decay. Here $d_{AHL} = 1.667 \times 10^{-12} \text{ m}^2/\text{s}$, $L = 1000 \text{ } \mu\text{m}$, and $k = 2$ (wavelength $1000 \text{ } \mu\text{m}$). Concentrations (colorbar) given in M. Perturbation in p_C of amplitude steady state $\pm 33\%$ peak-to-peak decays over time. To achieve a stable wavelength ($> 832.3 \text{ } \mu\text{m}$), we had to increase the spatial domain.

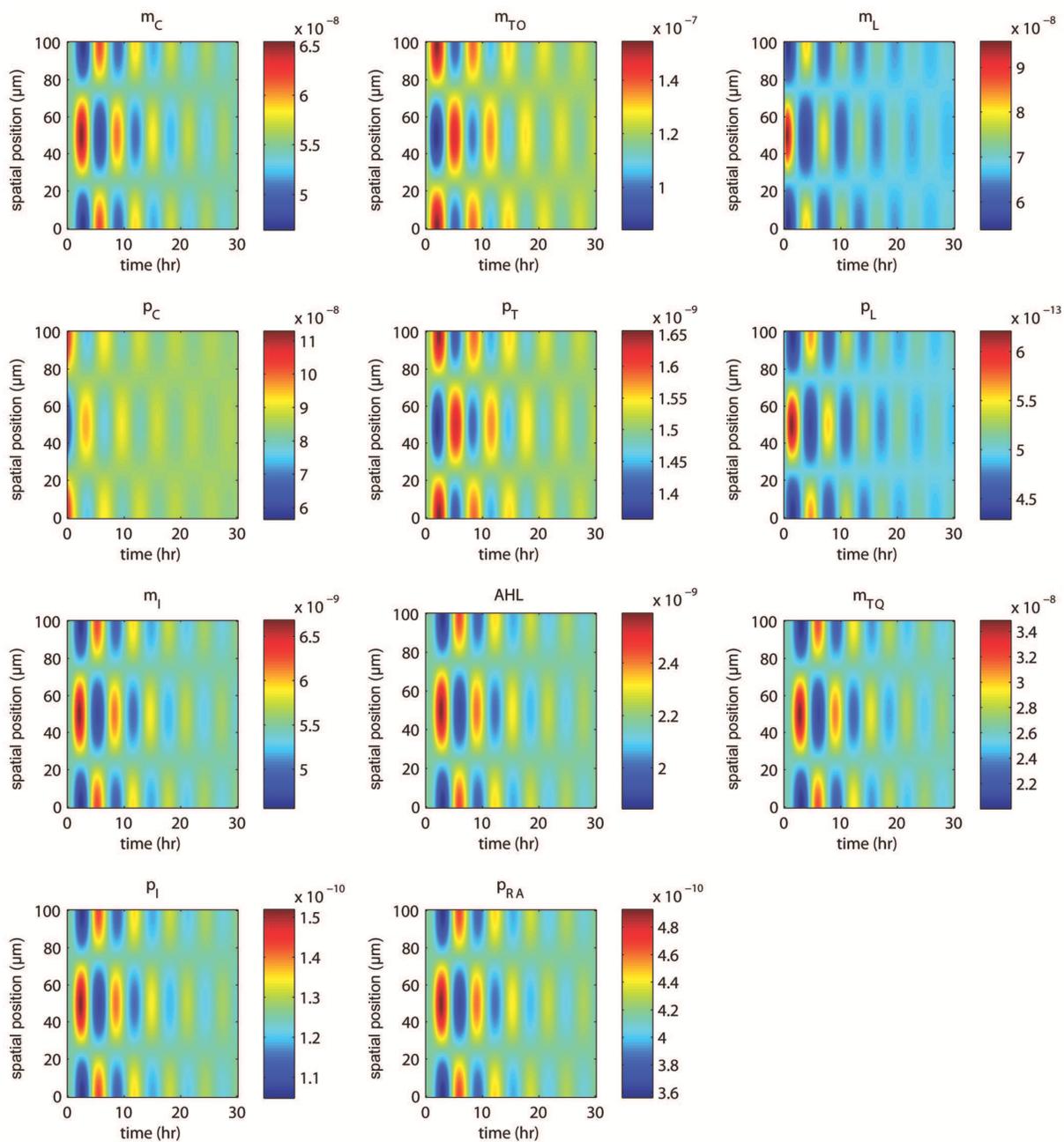


Figure 2.13: PDE simulation results for Parameter Set 1 in line of cells without diffusion. Here $d_{AHL} = 1.667 \times 10^{-12} \text{ m}^2/\text{s}$, $L = 100 \text{ } \mu\text{m}$, and $k = 2$ (wavelength $100 \text{ } \mu\text{m}$). Concentrations (colorbar) given in M. Perturbation in p_C of amplitude steady state $\pm 33\%$ peak-to-peak decays over time.

Table 2.3: Steady-state concentrations given by the analysis for parameter sets given in Table 2.2.

Species	Steady-state Concentration for PDE Simulation (Parameter Set 1)	Steady-state Concentration for Stochastic Simulation (Parameter Set 2)
m_C	5.478×10^{-8} M	1.314×10^{-7} M
p_C	8.474×10^{-8} M	1.687×10^{-8} M
m_{TO}	1.200×10^{-7} M	1.314×10^{-7} M
p_T	1.506×10^{-9} M	1.687×10^{-8} M
m_L	6.828×10^{-8} M	1.314×10^{-7} M
p_L	4.992×10^{-13} M	1.687×10^{-8} M
m_I	5.478×10^{-9} M	1.314×10^{-8} M
p_I	1.254×10^{-11} M	1.734×10^{-8} M
A	2.174×10^{-9} M	6.000×10^{-9} M
p_{RA}	4.166×10^{-10} M	1.200×10^{-8} M
m_{TQ}	2.619×10^{-8} M	1.314×10^{-8} M

Table 2.4: System characteristics given by the analysis for parameter sets given in Table 2.2.

Instability Measurement	Value for PDE Simulation (Parameter Set 1)	Value for Stochastic Simulation (Parameter Set 2)
Instability threshold d_{thresh}	9.5×10^{-4}	2.657×10^{-2}
Maximum unstable wavelength	832.3 μm	49.77 μm
Minimum unstable wave number k for $L = 100\mu\text{m}$	all are unstable	5
Minimum unstable wave number k for $L = 1000\mu\text{m}$	3	41

single molecule in an *E. coli* cell to be 1.5 nM [81], a number of steady-state values fall near or below this threshold (Figure 2.14), particularly p_L and p_I . This implies that: a) stochastic simulations are necessary for examining experimental plausibility, and b) Parameter Set 1 would need to be modified to produce pattern due to the behavior of certain species in our system being dominated by noise. In this limit, stochastic models better capture the behavior of *in vivo* systems because of their inability to respond to unrealistic concentration changes of less than one molecule per cell.

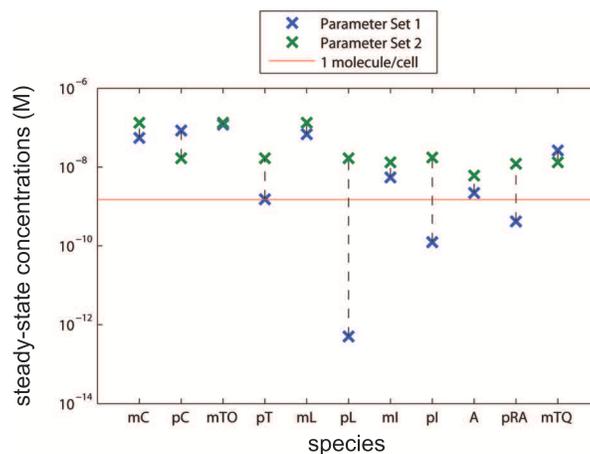


Figure 2.14: Comparison of steady-state concentrations for Parameter Sets 1 and 2. A number of steady-state concentrations for Parameter Set 1 lie near or below the threshold of 1 molecule/*E. coli* cell (red line). Parameter Set 2 has been chosen such that all steady-state concentrations lie above this threshold.

2.5.3 Parameter Selection for Stochastic Simulations

Even with suboptimal parameter values and parts, we have demonstrated that spatio-temporal patterning can be obtained in PDE simulation. Here we show that with other parameter values that are still biologically realistic, we can improve the system performance to also produce patterning in a discrete, stochastic environment.

This refers to the parameters given in the “Value for Stochastic Simulation (Parameter Set 2)” column of Table 2.2. With Parameter Set 1 based on physically characterized parts currently used in synthetic biology, we were unable to produce patterning on the stochastic simulator. The goal was then to find a parameter set that deviates slightly from the accepted literature values in order to boost steady-state concentrations to produce patterning. All of these values are physically possible based on information in the literature (see references in Table 2.2) and in the future parts are likely to be found that match our chosen parameter values.

The protein binding affinity, Hill coefficient, and degradation constants in the oscillator loop were made to be equal for all three components as in [42]. The protein binding affinity is at a concentration of two proteins. Even though the literature values show a binding affinity concentration of less than 1 protein, multiple binding sites and greater than unity plasmid copy numbers mean that the effective binding affinity should be higher than 1 protein. This immediately takes care of the extremely low steady-state value of p_L .

The steady-state concentration of AHL in Parameter Set 1 was too low to develop steep enough gradients necessary for diffusion-driven patterning. It is known that crosstalk exists between different AHL systems. In the deterministic analysis, we used the *V. fischeri* AHL system. The circuitry was changed to a mixed AHL system for stochastic simulation. An

autoinducer synthase that is unmatched to the *lux* promoter is used. This way the binding affinity of the unmatched AHL to LuxR is much smaller than in the matched case, thereby boosting the steady-state concentration of AHL. RhlI from *P. aeruginosa* is one such example of an autoinducer synthase that may be used with the P_{LuxI} promoter ([106, 105]).

The steady-state concentrations for Parameter Set 2 can also be seen in Figure 2.14 and do not fall below 6 nM (4 molecules/cell). We also verified the desired system behavior of this parameter set in PDE simulation (Figure 2.15). The imprinted wave $k = 6$ was chosen because it falls above the minimum unstable wave number of $k = 5$ for this parameter set ($(k\pi/L)^2 d_{AHL} > 2.66 \times 10^{-2}$). This new parameter set results in growth of additional wave numbers other than the imprinted one, highlighting the nonlinear nature of our system (Figure 2.16). These effects arise when oscillations start to reach near-maximal amplitudes and would likely be seen for Parameter Set 1 if the simulations were run for a much longer time.

Using this realistic but “relaxed” set of parameters, patterns were then observed in the stochastic simulations.

2.5.4 Stochastic Model and Simulation

We developed a set of reactions for stochastic simulation that, using the law of mass action and the quasi-steady-state approximation, would exactly match our set of PDE equations. The full set of reactions used in our stochastic simulations can be found in Table 2.5. For a Hill coefficient of exactly two, we assumed that each promoter had a single binding site to which only the dimerized form of the appropriate activator or inhibitor could bind. The table of reactions follows the species and variable naming scheme used in the PDE set found in (2.17) with the addition of dimers p_{x2} and promoters Pr_x . The dissociation constants K_x are combinations of the on and off dimerization and binding constants. In general,

$$K_x = \sqrt{\frac{k_{\text{doff}_x} k_{\text{off}_x}}{k_{\text{don}_x} k_{\text{on}_x}}}.$$

Stochastic simulations of the network were performed using the Stochastic Simulator Compiler (SSC) v0.6 [85]. The output from SSC was reformatted with custom Perl scripts and then plotted in MATLAB. SSC handles concentrations in units of molecules, so all parameter values were scaled appropriately, but the output values are converted to units of molarity here for ease of comparison. Reported values for protein concentrations are the totals of all forms of the protein: monomer, dimer, and bound to promoter. We represented cells with cubes of edge length $1 \mu\text{m}$. For single cell simulations, the cell was located at the center of a volume of $100 \times 1 \times 1 \mu\text{m}$. All multi-cell simulations consisted of a line containing 100 directly adjacent cells.

To compare the behavior of PDE and stochastic simulations, we first ran single cell simulations to verify that the general expected behavior was maintained. While not indicative of the system’s ability to generate pattern, these simulations allow us to draw comparisons between our PDE and stochastic models. To observe both an oscillating cell and a quenched cell, we used a single cell in the center of a long, empty volume. Without AHL diffusion,

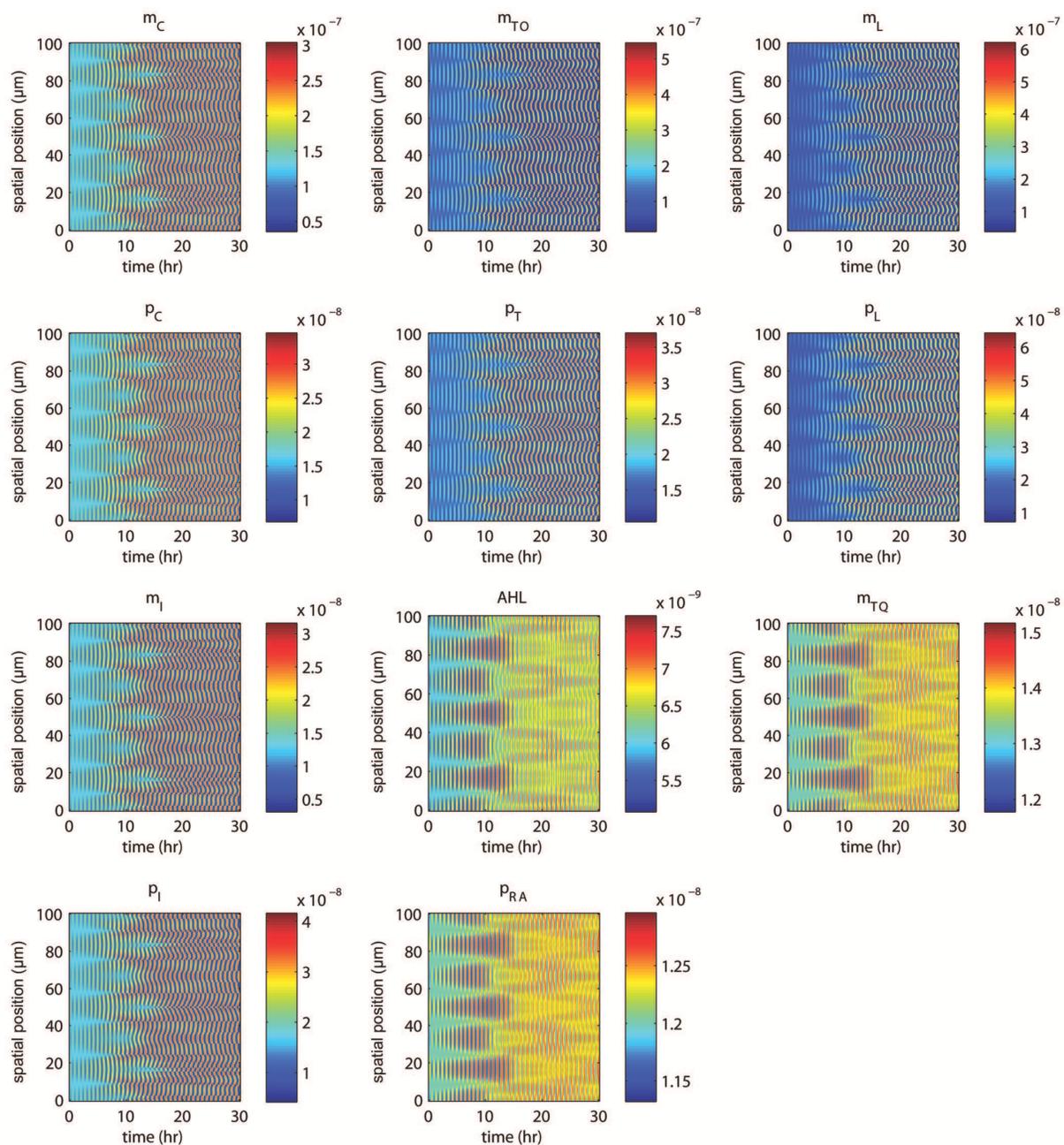


Figure 2.15: PDE simulation results for Parameter Set 2 in line of cells with diffusion and growth for comparison with Figure 2.11. Here $d_{AHL} = 1.667 \times 10^{-12} \text{ m}^2/\text{s}$, $L = 100 \mu\text{m}$, and $k = 6$ (wavelength $33.3 \mu\text{m}$). Concentrations (colorbar) given in M. Perturbation in p_C of amplitude steady state $\pm 33\%$ peak-to-peak leads to growth of the inhomogeneity and higher wave numbers.

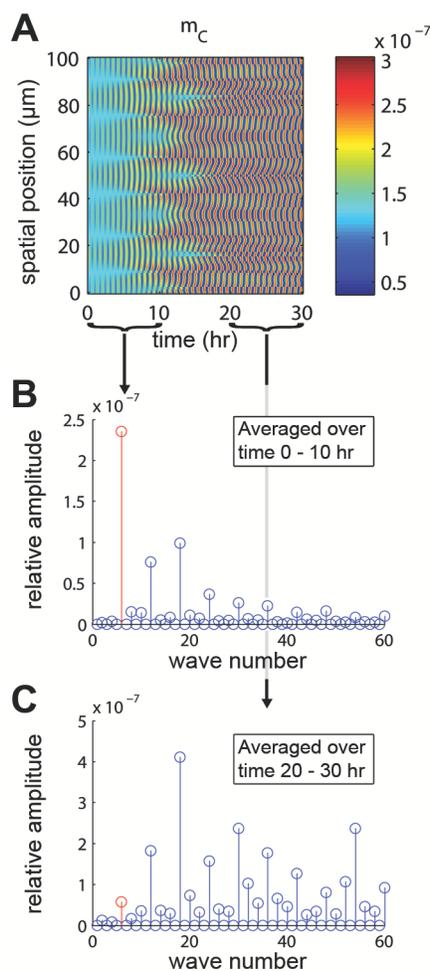


Figure 2.16: (A) PDE simulation results for cI mRNA using Parameter Set 2 repeated from Figure 2.15). Concentrations (colorbar) given in M. Parameter Set 2 oscillates much faster than Parameter Set 1, and because of this we observe more interesting behavior within the 30-hr simulation window once the oscillations reach their maximum amplitude. In particular, the imprint initially grows, but then the energy moves into higher harmonics as time goes on. (B) Discrete cosine transform (DCT) of cI mRNA over the window of 0-10 hr. The imprinted wave (shown in red) dominates and grows. (C) Over the window 20-30 hr, higher harmonics have begun to dominate.

Table 2.5: Full reaction set for stochastic simulations. Kinetic rate constants chosen to validate quasi-steady-state approximation and to match Parameter Set 2 of Table 2.2.

Reaction Type	Reactions
Dimerization	$p_x + p_x \xrightleftharpoons[k_{\text{doff}_x}]{k_{\text{don}_x}} p_{x2}$ for $x \in \{C, L, RA, T\}$
Transcription factor-promoter binding	$Pr_x + p_{x2} \xrightleftharpoons[k_{\text{off}_x}]{k_{\text{on}_x}} Pr_x p_{x2}$ for $x \in \{C, L, RA\}$ $Pr_x + p_{T2} \xrightleftharpoons[k_{\text{off}_T}]{k_{\text{on}_T}} Pr_x p_{T2}$ for $x \in \{TO, TQ\}$
Maximal transcription	$Pr_C \xrightarrow{V_{PR}} Pr_C + m_L$ $Pr_L \xrightarrow{V_{PlacO-1}} Pr_L + m_{TO}$ $Pr_{TO} \xrightarrow{V_{PLtetO-1}} Pr_{TO} + m_C$ $Pr_{TQ} \xrightarrow{V_{PLtetO-1}} Pr_{TQ} + m_I$ $Pr_{RAPRA2} \xrightarrow{V_{PLuxI}} Pr_{RAPRA2} + m_{TQ}$
Leaky transcription	$Pr_C p_{C2} \xrightarrow{\ell_{PR} V_{PR}} Pr_C p_{C2} + m_L$ $Pr_L p_{L2} \xrightarrow{\ell_{PlacO-1} V_{PlacO-1}} Pr_L p_{L2} + m_{TO}$ $Pr_{TO} p_{T2} \xrightarrow{\ell_{PLtetO-1} V_{PLtetO-1}} Pr_{TO} p_{T2} + m_C$ $Pr_{TQ} p_{T2} \xrightarrow{\ell_{PLtetO-1} V_{PLtetO-1}} Pr_{TQ} p_{T2} + m_I$ $Pr_{RA} \xrightarrow{\ell_{PLuxI} V_{PLuxI}} Pr_{RA} + m_{TQ}$
Translation	$m_x \xrightarrow{\epsilon_x} p_x$ for $x \in \{C, I, L\}$ $m_{Tx} \xrightarrow{\epsilon_{Tx}} p_T$ for $x \in \{O, Q\}$
AHL production	$p_I \xrightarrow{v_3} p_I + A$
LuxR-AHL binding	$p_R + A \xrightleftharpoons[k_{\text{boff}_{AR}}]{k_{\text{bon}_{AR}}} p_{RA}$
Degradation	$p_x \xrightarrow{\gamma_x} \emptyset$ for $x \in \{C, T, L, I\}$ $m_x \xrightarrow{\gamma_{mO}} \emptyset$ for $x \in \{C, L, TO\}$ $m_x \xrightarrow{\gamma_{mQ}} \emptyset$ for $x \in \{I, TQ\}$ $A \xrightarrow{\gamma_A} \emptyset$
Diffusion	A in cell $x \xrightarrow{d_{AHL}} A$ in cell $x \pm 1$

the cell remains isolated and we expect oscillations to decay to the steady state. With diffusion, AHL diffuses into the empty volume and weakens the quenching loop, meaning oscillations are expected to grow. Both PDE and stochastic simulations confirmed these expectations (Figures 2.17-2.20). The simulations exhibited similar behavior but oscillations in the stochastic simulations are slower and more irregular, due to stochasticity and our modeling assumption that the dimerization and binding reactions are at equilibrium in the PDE model. Oscillations in the stochastic simulations are significantly slower – about 5 times slower in the decaying case, and 10 times slower in the growing case – which lead us to choose faster degradation rates for Parameter Set 2. In a cell without diffusion, stochasticity keeps the system oscillating at a small amplitude with occasional “firing events,” where a few cycles of increased oscillation amplitude occur before the system settles again. Both PDE and stochastic simulations exhibit the same phase relationship between the proteins in the oscillator loop and a slower period of oscillation when growing as opposed to decaying (Figures 2.17-2.20).

As expected, stochastic simulations with Parameter Set 1 in a line of cells were unable to produce patterning due to the low steady-state concentration values (results not shown), but did yield some insights. In particular, any initial imprint we imposed would very rapidly (< 0.5 hr) decay into noise, likely due to low copy numbers. With only four or five promoter binding sites per cell and the fact that almost all of them are bound in steady state, a large change in a single species of the system is unlikely to be able to propagate quickly enough throughout the system due to the bottlenecks at the promoter binding sites. Thus

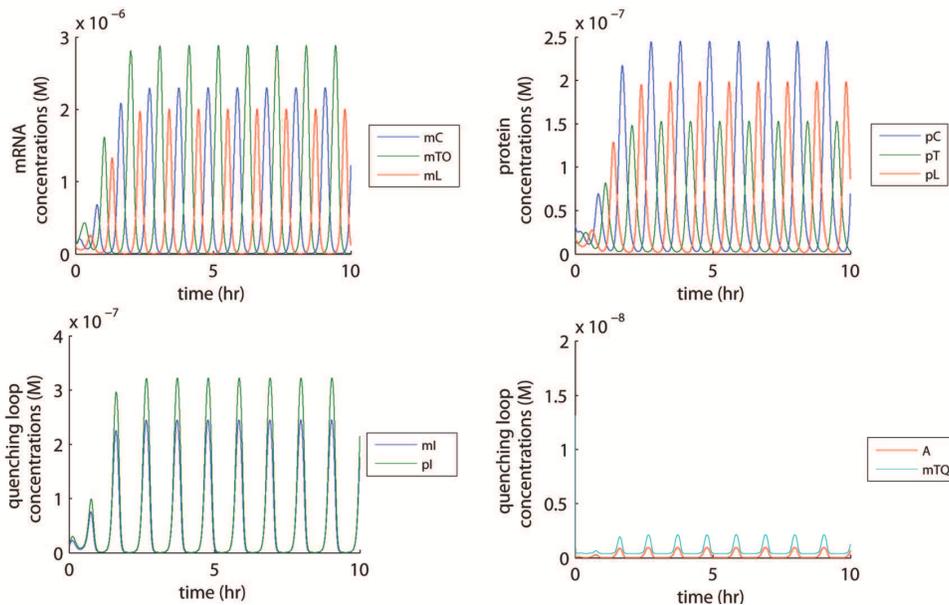


Figure 2.17: PDE simulation results for Parameter Set 2 in single cell with diffusion ($d_{AHL} = 1.667 \times 10^{-12}$ m²/s). Initial perturbation in p_C of twice the steady state value causes growing oscillations until stable limit cycle is reached.

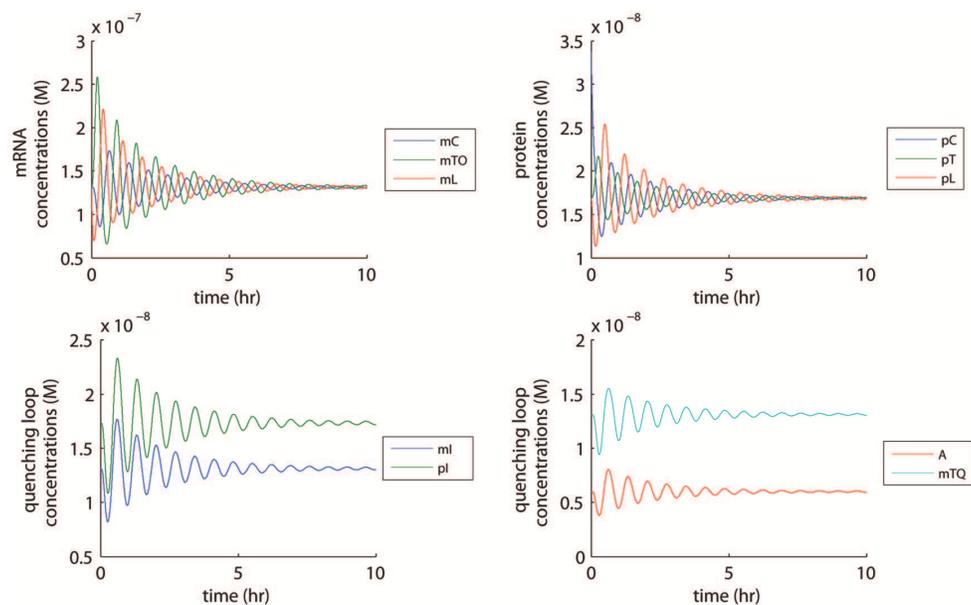


Figure 2.18: PDE simulation results for Parameter Set 2 in single cell without diffusion ($d_{AHL} = 0 \text{ m}^2/\text{s}$). Initial perturbation in p_C of twice the steady state value causes decaying oscillations, which asymptotically approach the steady state.

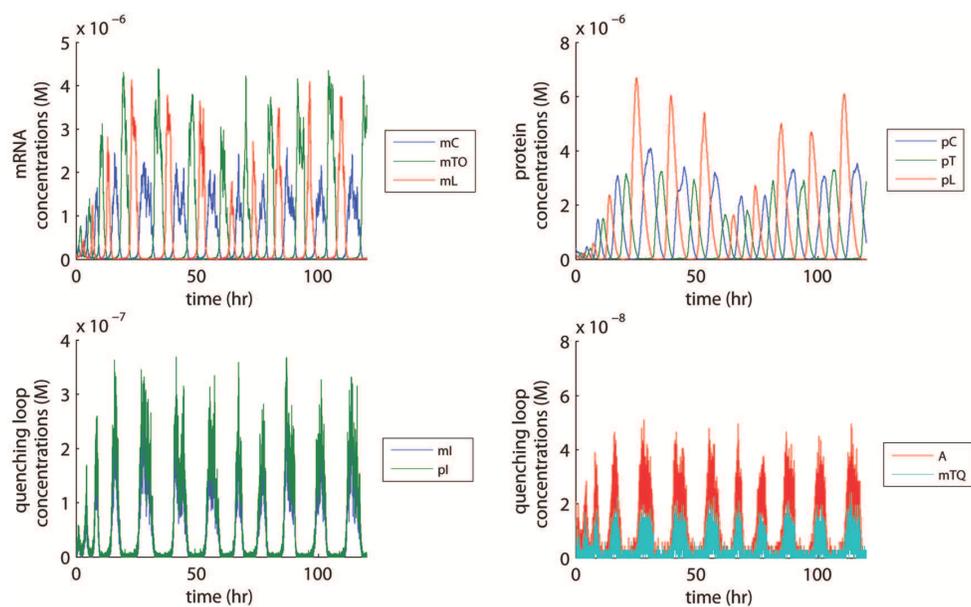


Figure 2.19: Stochastic simulation results for Parameter Set 2 in single cell with diffusion ($d_{AHL} = 1.667 \times 10^{-12} \text{ m}^2/\text{s}$). Initial perturbation in p_C of twice the steady state value rounded to nearest molecule causes growing oscillations that eventually exhibit relatively stable period and amplitude.

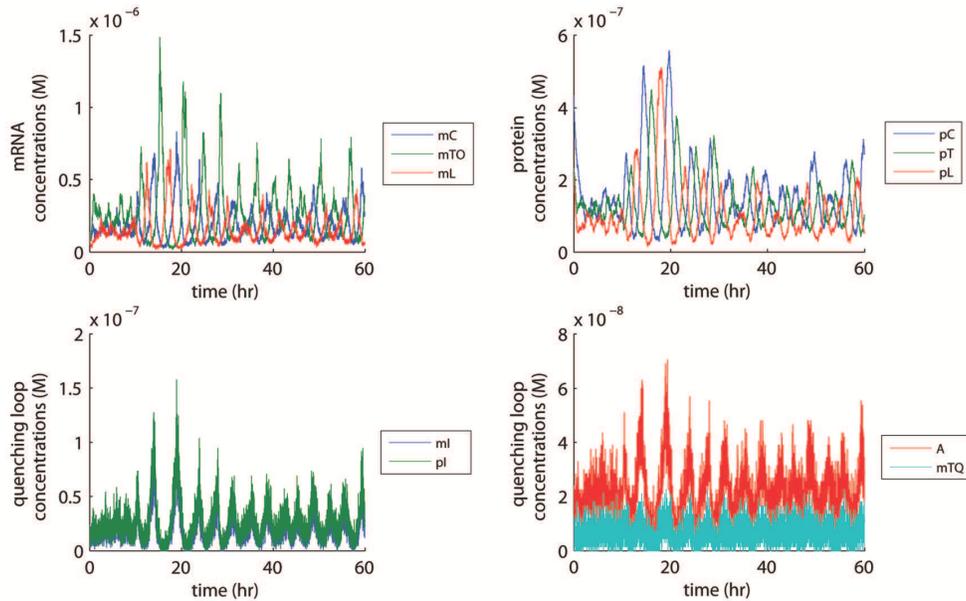


Figure 2.20: Stochastic simulation results for Parameter Set 2 in single cell without diffusion ($d_{AHL} = 0 \text{ m}^2/\text{s}$). Initial perturbation in p_C of twice the steady state value rounded to nearest molecule causes sustained oscillations of short period and small amplitude. Occasional “firing events” eventually settle.

we avoided imprinting and used the ability of the stochasticity in our system to naturally excite high wave numbers.

Indeed, stochastic simulations with Parameter Set 2 in a line of cells exhibit growing oscillations and eventually produce spatio-temporal patterns (Figure 2.21). Large amplitude oscillations emerge around 20 hours and an obvious pattern emerges as time goes on. Visually, patterning is most evident in AHL due to the effects of diffusion. Without diffusion, no spatial patterns emerge with single cell oscillations occurring randomly (results not shown).

To quantify the patterns produced by our system, we use the *discrete cosine transform* (DCT) to check the relative presence of the different emerging wave numbers. All wave numbers higher than a threshold ($k \geq 5$ for Parameter Set 2) should grow in the presence of noise according to our analysis, but a number of factors, including stochasticity and the discrete nature of only having 100 cells in our simulations, prevent them from growing uniformly. The exact wave numbers vary from simulation to simulation, but the averaged DCT over time frames late in simulations (beyond the “start-up” phase) always shows a number of spikes that are prominent across most species in the system (see Supplementary information B). The exceptions to this are AHL and subsequent species in the quenching loop, where diffusion acts as a low-pass filter and attenuates high wave numbers. This filtering effect is what accounts for the visual “bleeding” effect of diffusion.

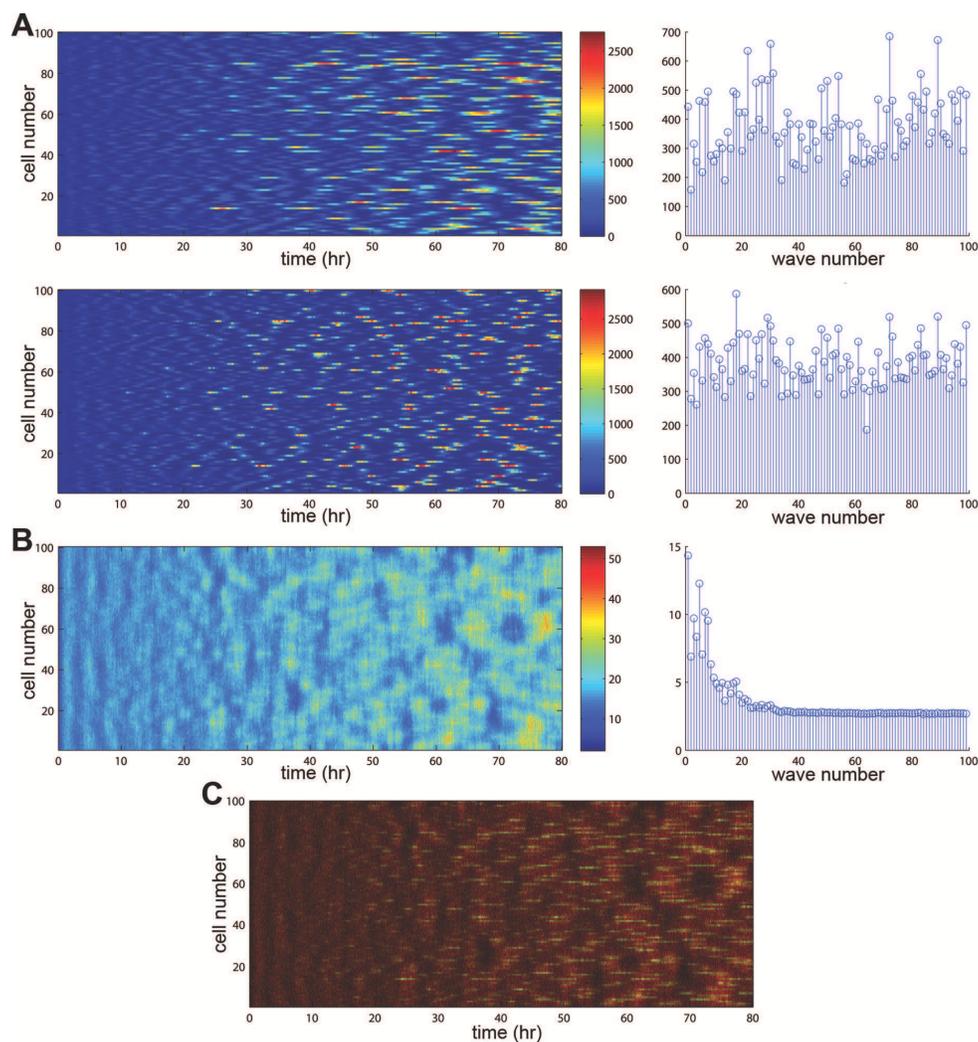


Figure 2.21: Representative sample of stochastic simulation results for a line of cells with homogeneous initial condition, including color plots (left) and DCTs averaged over hours 50 to 80 (right). The stochasticity causes oscillations to arise naturally and can be seen as early as hour 20. See Supplementary information B for full simulation results. (A) Results for p_T (top) and m_L (bottom) are indicative of the behavior for mRNA and proteins for λ cI, TetR, LacI, and LuxI. While the DCT plots vary from species to species, certain wave numbers are found to be more pronounced across all species, particularly $k = 18, 22, 30, 72, 83,$ and 89 . (B) Results for AHL produce similar behavior in all downstream species in the quenching loop. Both the color plot and DCT are markedly different due to the effects of diffusion, which causes a “spreading” of the rapid peaks seen in mRNA and protein color plots and acts like a low-pass filter in the frequency domain. (C) Overlay plot of AHL and p_I demonstrating the correspondence between the peaks in the species as well as the effect of diffusion. AHL was monochromed in red and p_I in green, leading to the appearance of yellow in areas of large overlap.

2.5.5 Discussion of Quenched Oscillator Implementation

The process of producing a set of parameters which produce pattern in the stochastic regime provided several insights which can inform implementation decisions as new promoters, proteins, and parameter manipulation techniques become available. These findings may also be of use when searching for putative natural systems which exhibit this behavior.

Two of the most restrictive parameters that we had to change significantly from our initial solution set were promoter leakage rates and dissociation constants. High amounts of leakage makes it simultaneously more difficult to make the oscillator subsystem unstable and more difficult for the quenching loop to stabilize the overall system. The dissociation constants directly affected the steady-state concentrations of the protein species in our system; the system fails to produce patterning when these values are too small. These observations were made from studying the form of the expressions for X (2.19) and F (2.21) and many other such observations and insights can be drawn from the analysis.

A few considerations only became relevant when performing stochastic simulations, the biggest of which was the bottleneck of promoter binding sites. In the PDE model, new mRNA would be produced at a rate that was a function of the amount of the appropriate activator or inhibitor in the system. By enumerating the number of promoter binding sites, we decrease the sensitivity of the system to very large concentrations of the activators and inhibitors and increase the importance of each binding and unbinding event. Analytically, we can maintain the same system behavior by holding the product $V_x N_x$ in each mRNA differential equation constant. Arbitrarily increasing the copy numbers this way has its own drawbacks. We assume the concentration of LuxR is constitutively produced and is constant. At our current value of 18 nM (12 molecules/cell), we can only bind at most six promoters with LuxR-AHL dimers, so having a large N_{TQ} will not change the amount of m_{TQ} being produced, which deviates from what our PDE model predicts.

Assuming proper parameter values can be chosen for our system, our analysis generates a testable hypothesis for a possible experimental implementation. When setting up the experiment, the following additional concerns should be taken into account:

- Beyond finding parameters that meet the Turing instability conditions, system speed is very important because it determines the visibility of changes in the system over the course of a normal experiment duration. System speed is most directly affected by the degradation rates of every species in the system. These change the period of oscillations as well as the growth and decay rates of wave modes. Very slow growth and decay would delay the emergence of visible patterns and make experimental debugging difficult because any activity would be hard to observe. Very long experiments are problematic in terms of collecting data and dealing with cell division and lifespan.
- A reporter gene was unnecessary in simulation, but one would need to be used in experiments. As seen in Figure 2.21, there are two distinct types of qualitative behaviors: the proteins cI, LacI, TetR, and LuxI exhibit brief bursts localized to single cells while AHL and subsequent quenching loop species exhibit more spread out behavior due to

diffusion. It is possible to attach a fluorescent protein to the appropriate loop to follow either type of behavior. While AHL may produce a more visually-pleasing patterning, the oscillator loop species undergo larger swings in number of molecules, which would be easier to discern in units of fluorescence.

2.6 Newer Synthetic Technologies Improve Quenched Oscillator Design

The biggest barrier to experimental plausibility of the quenched oscillator system is caused by the limited number of well-characterized synthetic components and methods for manipulating parameter values currently available to synthetic biologists. Here we explore a way to address this issue by using zinc finger proteins (ZFPs) in a new quenched oscillator implementation. This serves the dual purpose of making the quenched oscillator more experimentally plausible and creating more tunable synthetic parts for the community to use. As recently demonstrated [61], the range of DNA sequences that can be targeted by ZFPs makes it possible to create sets of orthogonal promoter-ZFP based transcriptional repressor pairs in *E. coli*. This orthogonality allows us to build larger and more complex transcriptionally regulated gene networks than was previously possible. We show here that when paired with constitutive amounts of small RNAs (sRNAs) that bind to and down regulate the translation of the mRNA of the appropriate ZFP, the ZFP inverters can be tuned to exhibit gains beyond those of common repressors such as λ *cI*, *lacI*, and *tetR*. With three of these ZFP-sRNA pairs, we propose a new ring oscillator that has a more tunable gain and consists of repressors with nearly identical characteristics. The resulting rotational symmetry in the circuit is indeed desirable, because it simplifies the analysis and yields a verifiable condition for the existence of a limit cycle. This oscillator also enables us to use a slightly modified quenching loop structure due to the presence of the sRNAs in the system.

2.6.1 Zinc Finger Protein Technology

ZFPs contain a fold coordinated by a zinc ion and commonly bind DNA, but some are able to bind other molecules such as RNA, proteins, or small molecules. Zinc fingers are frequently found in transcription factors, especially in eukaryotes. There are several classes of zinc finger proteins and the most commonly used class in engineered systems are the C₂H₂-type. This type of finger contains a Zn(II) ion coordinated by two cysteine and two histidine residues and a single C₂H₂-type zinc finger binds to 3-4 bases of double-stranded DNA. Natural C₂H₂-type zinc finger proteins generally contain three or more zinc fingers, allowing them to bind to nine or more base pairs of DNA with dissociation constants commonly in the nanomolar range. See Figure 2.22 for a general diagram of their structure.

Synthetic ZFPs can be created to target a wide variety of DNA sequences. By targeting a ZFP to bind within a promoter sequence, it is possible to repress transcription by over 250 fold due to steric hindrance of the RNA polymerase [61]. These transcriptional repressing

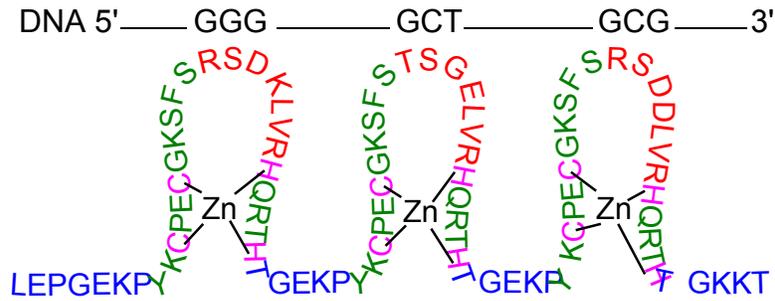


Figure 2.22: Diagram of basic structure of Cys₂His₂-type zinc finger protein. Each finger binds to 3-4 bases of double-stranded DNA and natural ZFPs generally contain three or more fingers. The ZFP is shown as a color-coded amino acid sequence and the target DNA sequence is shown above in black. The variable regions that determine the target bases are shown in red. Flanking these variable regions are repeated amino acid sequences.

ZFPs act only as DNA binding domains and have no additional activity. By combining ZFP operator sites with nominally constitutive promoters, it is possible to design new promoter-transcriptional repressor pairs. Because of the range of DNA sequences that can be targeted by ZFPs, it becomes possible to create sets of orthogonal promoter-ZFP based transcriptional repressor pairs [61].

2.6.2 Hybrid sRNA-Repressor Topology for Increased Cooperativity

ZFP-based transcriptional repressors described above are monomeric proteins and bind to DNA without any cooperativity (Hill coefficient of 1). To construct an oscillator using ZFPs, we show that the binding of sRNA to ZFP mRNA provides an ultrasensitive response

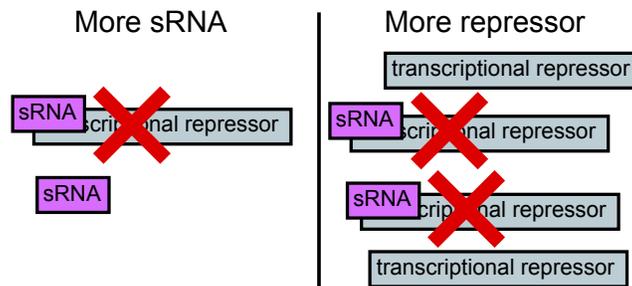


Figure 2.23: Diagram of sRNA thresholding, courtesy of William Holtz. If very little transcriptional repressor mRNA is present, then it is eliminated by the sRNA, masking its effect and effectively reducing promoter leakage. Once the sRNA threshold is exceeded, then the effect of the transcriptional repressor is seen again. At high concentrations of the transcriptional repressor, the effect of the sRNA is negligible (i.e. same maximal repression).

in a manner similar to the one produced by protein sequestration in [22] and can be tuned to generate large gains comparable to repressors with higher Hill coefficients.

Small RNAs are non-coding RNAs that bind to mRNA and post-transcriptionally regulate the translation of the mRNA [89]. Many sRNAs down regulate the translation of the mRNA they bind to. By constitutively but weakly expressing an sRNA that targets a ZFP-based transcriptional repressor, low levels of ZFP mRNA will not be translated but high levels of ZFP mRNA will result in translation and repression of the cognate promoter [61]. Figure 2.23 illustrates the effect of the interaction between sRNA and a targeted transcriptional repressor in both of these cases.

For the purposes of modeling, here we assume that the sRNA-mRNA complexes degrade away at a much faster rate than either individual molecule, eliminating the unbinding reaction from our system of equations. We also use constant terms to represent the production of sRNA from nominally constitutive promoters. Our model is then:

$$\begin{aligned}\frac{d}{dt}s_y &= V_{sy}N_{sy}C - k_{fy}s_ym_y - \gamma_s s_y \\ \frac{d}{dt}m_y &= V_xN_xC \left(\frac{1}{1 + p_x/K_x} + \ell_x \right) - k_{fy}s_ym_y - \gamma_m m_y \\ \frac{d}{dt}p_y &= \epsilon_y m_y - \gamma_p p_y\end{aligned}\tag{2.23}$$

where the subscripts x and y refer to the input and output species, respectively, and the state variables s , m , and p represent sRNA, mRNA, and protein. The parameters V_x and V_{sy} are velocity constants, N_x and N_{sy} are copy numbers, k_{fy} is the forward binding rate of sRNA and ZFP mRNA, K_x is the dissociation constant for ZFP-DNA binding, ℓ_x is the leakage rate normalized to V_x , γ_i are degradation rates, and ϵ_y is the protein translational rate. The parameters are subscripted according to their corresponding species except for velocity and leakage constants and copy numbers, which are subscripted by promoter. The parameter C is the concentration level generated by a single molecule in an *E. coli* cell.

Taking the input and output of this system to be p_x and p_y , respectively, we get the following relationship between steady-state concentrations of the input and output:

$$\bar{p}_y = \psi(\phi_1(\bar{p}_x)),\tag{2.24}$$

where the functions $\psi(\cdot)$ and $\phi_n(\cdot)$ are defined as follows:

$$\psi(z) = \frac{\epsilon_y}{2\gamma_m\gamma_p} \left[\left(z - V_{sy}N_{sy}C - \frac{\gamma_s\gamma_m}{k_{fy}} \right) + \sqrt{\left(z - V_{sy}N_{sy}C - \frac{\gamma_s\gamma_m}{k_{fy}} \right)^2 + \frac{4\gamma_s\gamma_m}{k_{fy}}z} \right],\tag{2.25}$$

$$\text{and } \phi_n(z) = V_xN_xC \left(\frac{1}{1 + (z/K_x)^n} + \ell \right).\tag{2.26}$$

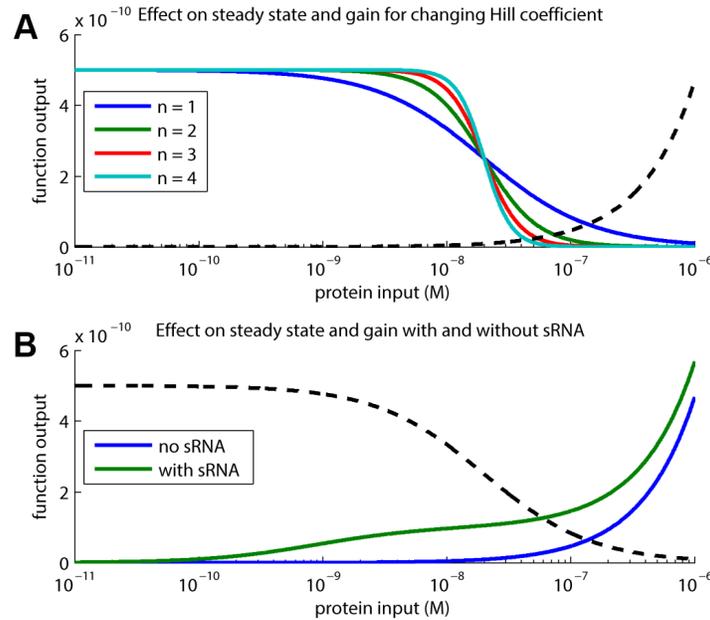


Figure 2.24: Steady-state values are found at the intersection of a solid and dotted line. The gain is the ratio of the negative to the positive slope at the intersection. The parameters used are $V_{sy} = 0.01 \text{ s}^{-1}$, $V_x = 0.05 \text{ s}^{-1}$, $N_{sy} = N_x = 10$, $C = 10^{-9} \text{ M}$, $\gamma_s = 2.3 \times 10^{-3} \text{ s}^{-1}$, $\gamma_m = 1.2 \times 10^{-2} \text{ s}^{-1}$, $\gamma_p = 3.9 \times 10^{-4} \text{ s}^{-1}$, $k_{fy} = 69 \times 10^6 \text{ M}^{-1}\text{s}^{-1}$, $K_x = 20 \times 10^{-9} \text{ M}$, $\ell_x = 0.001$, and $\epsilon_y = 0.01 \text{ s}^{-1}$. (A) For generic repressors with increasing Hill coefficients, the steady state decreases and the gain increases. (B) For monomeric repressors (such as ZFPs) with sRNA sequestration, the gain is also increased at the new steady state.

The plot of (2.24) looks similar to a standard Hill function, but the response has a sharper curve for the input range just greater than K_x . To compare the performance of this inverter against standard repressors modeled with Hill functions, we will compare inverter gains. For our application of a ring oscillator made of nearly-identical inverters, the steady state \bar{x} is given by $\bar{x} = \bar{p}_y(\bar{x})$, or equivalently, $\phi_1(\bar{x}) = \psi^{-1}(\bar{x})$, and the gain is defined as:

$$\text{gain} = \left. \frac{d\bar{p}_y(x)}{dx} \right|_{x=\bar{x}} = \frac{\phi_1'(\bar{x})}{(\psi^{-1})'(\bar{x})}, \quad (2.27)$$

where we note that:

$$\psi^{-1}(z) = z \left(\frac{k_{fy} V_{sy} N_{sy} C \gamma_p}{k_{fy} \gamma_p z + \gamma_s \epsilon} + \frac{\gamma_m \gamma_p}{\epsilon} \right). \quad (2.28)$$

Graphically, the interpretation is that \bar{x} is the intersection of the functions $\phi_1(x)$ and $\psi^{-1}(x)$ and the gain is the ratio of their slopes at the intersection. To find the gain for standard repressors, calculate the same ratio of slopes at the intersection using the appropriate $\phi_n(x)$ and $\psi^{-1}(x)$ with $k_{fy} = 0$. A comparison of inverters for the parameters $V_{sy} = 0.01 \text{ s}^{-1}$,

Table 2.6: Steady state and gain characteristics of various inverters with different Hill coefficients n and with or without sRNA. For a visual representation, see Figure 2.24.

Inverter	Steady state \bar{x}	Gain
$n = 1$	1.37×10^{-7}	0.866
$n = 2$	7.39×10^{-8}	1.837
$n = 3$	5.37×10^{-8}	2.796
$n = 1$ with sRNA	5.94×10^{-8}	3.225
$n = 4$	4.42×10^{-8}	3.746

$V_x = 0.05 \text{ s}^{-1}$, $N_{sy} = N_x = 10$, $C = 10^{-9} \text{ M}$, $\gamma_s = 2.3 \times 10^{-3} \text{ s}^{-1}$, $\gamma_m = 1.2 \times 10^{-2} \text{ s}^{-1}$, $\gamma_p = 3.9 \times 10^{-4} \text{ s}^{-1}$, $k_{fy} = 69 \times 10^6 \text{ M}^{-1}\text{s}^{-1}$, $K_x = 20 \times 10^{-9} \text{ M}$, $\ell_x = 0.001$, and $\epsilon_y = 0.01 \text{ s}^{-1}$ is shown in Figure 2.24. The calculated gains for these inverters are listed in Table 2.6 and we can see that a monomeric ZFP with sRNA has as much gain as a repressor with Hill coefficient 3.5.

2.6.3 Zinc Finger Protein Oscillator

To construct the ZFP oscillator, we put three orthogonal ZFP-sRNA pairs in a loop (Figure 2.25). The model will consist of nine differential equations: three copies of equations (2.23) with the subscripts replaced by $x \in \{3, 1, 2\}$ and $y \in \{1, 2, 3\}$.

The steady state can be found by solving the following system of equations for $\bar{\alpha}_i$:

$$\begin{aligned} \gamma_m \left(\frac{\gamma_p K_1}{\epsilon_1} \right)^2 \bar{\alpha}_1^2 + \frac{\gamma_p K_1}{\epsilon_1} (Y_1 - X_3 + \frac{\gamma_s \gamma_m}{k_{f1}}) \bar{\alpha}_1 - \frac{\gamma_s}{k_{f1}} X_3 &= 0 \\ \gamma_m \left(\frac{\gamma_p K_2}{\epsilon_2} \right)^2 \bar{\alpha}_2^2 + \frac{\gamma_p K_2}{\epsilon_2} (Y_2 - X_1 + \frac{\gamma_s \gamma_m}{k_{f2}}) \bar{\alpha}_2 - \frac{\gamma_s}{k_{f2}} X_1 &= 0 \end{aligned}$$

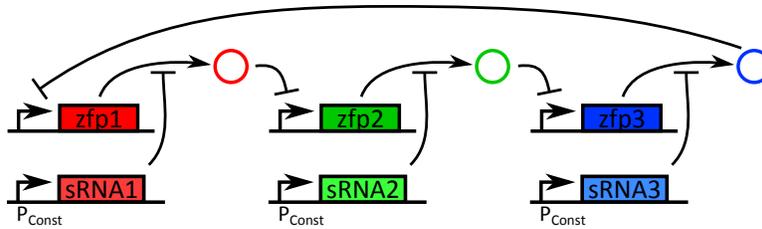


Figure 2.25: Schematic of the ZFP-sRNA oscillator. Three orthogonal ZFPs inhibit each other in a ring. Each ZFP is coupled with an orthogonal sRNA that binds to its mRNA and quickly degrades away, which leads to an ultrasensitive response and can be used to generate large effective Hill coefficients necessary to achieve oscillations. The positive interaction encompasses both transcription of mRNA and translation to ZFPs, so the sRNAs are shown to inhibit the translational portion of these interactions.

$$\gamma_m \left(\frac{\gamma_p K_3}{\epsilon_3} \right)^2 \bar{\alpha}_3^2 + \frac{\gamma_p K_3}{\epsilon_3} (Y_3 - X_2 + \frac{\gamma_s \gamma_m}{k_{f3}}) \bar{\alpha}_3 - \frac{\gamma_s}{k_{f3}} X_2 = 0 \quad (2.29)$$

where $X_i = V_i N_i C \left(\frac{1}{1 + \bar{\alpha}_i} + \ell_i \right)$ and $Y_i = V_{si} N_{si} C$. The state variables can be recovered using $\bar{p}_i = K_i \bar{\alpha}_i$, $\bar{m}_i = \gamma_p K_i \bar{\alpha}_i / \epsilon_i$, and $\bar{s}_i = \frac{V_{si} N_{si} C}{k_{fi} \gamma_p K_i \bar{\alpha}_i / \epsilon_i + \gamma_s}$.

The Jacobian linearization yields:

$$J_{osc} = \begin{bmatrix} J_1 & 0_{3 \times 3} & -B_3 \\ -B_1 & J_2 & 0_{3 \times 3} \\ 0_{3 \times 3} & -B_2 & J_3 \end{bmatrix},$$

where:

$$J_i = \begin{bmatrix} -a_{i1} & -c_{i2} & 0 \\ -c_{i1} & -a_{i2} & 0 \\ 0 & \epsilon_i & -\gamma_p \end{bmatrix} \text{ and } B_i = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & b_i \\ 0 & 0 & 0 \end{bmatrix}$$

with entries $a_{i1} = k_{fi} \bar{m}_i + \gamma_s$, $a_{i2} = k_{fi} \bar{s}_i + \gamma_m$, $c_{i1} = k_{fi} \bar{m}_i$, $c_{i2} = k_{fi} \bar{s}_i$, and $b_i = \frac{V_i N_i C}{K_i (1 + \bar{\alpha}_i)^2}$.

Assuming all species are identical ($\bar{p}_i = \bar{p}$, $\bar{m}_i = \bar{m}$, $\bar{s}_i = \bar{s}$, $a_{i1} = a_1$, etc.), then $J = I \otimes J_{eq} - L \otimes B_{eq}$, where:

$$J_{eq} = \begin{bmatrix} -a_1 & -c_2 & 0 \\ -c_1 & -a_2 & 0 \\ 0 & \epsilon & -\gamma_p \end{bmatrix}, B_{eq} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & b \\ 0 & 0 & 0 \end{bmatrix}, \text{ and } L = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

The eigenvalues of J are then identically those of $J_{eq} - \lambda_L B_{eq}$, where $\lambda_L \in \{1, -\frac{1}{2} \pm \frac{\sqrt{3}}{2}i\}$ are the three eigenvalues of L . The system will oscillate if J contains a pair of complex-conjugate eigenvalues with positive real part. It can be shown that the eigenvalues for $\lambda_L = 1$ remain stable, so we only have to check the eigenvalues of $J_{eq} - \lambda_L B_{eq}$ for $\lambda_L = -\frac{1}{2} \pm \frac{\sqrt{3}}{2}i$.

If we let all species match the parameter values used for the input-output curve in Figure 2.24, then $J_{eq} - \lambda_L B_{eq}$ contains the unstable eigenvalue $(3.141 - 9.886i) \times 10^{-4}$ for $\lambda_L = -\frac{1}{2} + \frac{\sqrt{3}}{2}i$. To verify oscillations, MATLAB and SSC simulations are shown in Figure 2.26.

2.6.4 ZFP-sRNA Quenched Oscillator

Using the ZFP-sRNA oscillator described above, we can now construct the quenched oscillator architecture shown in Figure 2.27 where species x_4 activates the production of x_1 sRNA. In a system with just mRNA and protein, the only way to inhibit x_1 is by promoting x_3 , yielding the original structure seen in Figure 2.4. First we will demonstrate that this new architecture can also produce diffusion-driven instability.

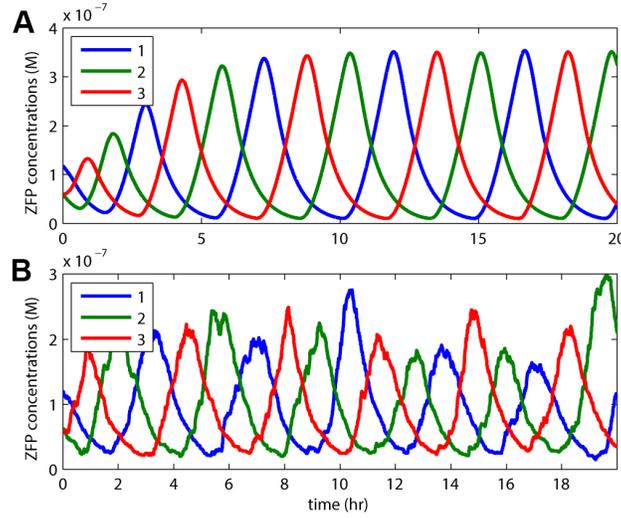


Figure 2.26: Verification of working ZFP-sRNA oscillator in simulation. Only ZFP concentrations shown here (mRNA and sRNA not shown). Parameter values for each species match those used in Figure 2.24. (A) Deterministic, continuous simulations using MATLAB. (B) Stochastic, discrete simulations using the Stochastic Simulation Compiler [85].

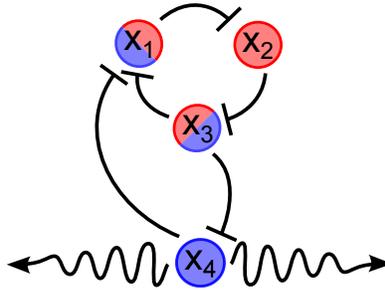


Figure 2.27: New quenching loop structure used here, where x_4 inhibits x_1 directly as opposed to 2.4, where x_4 activates production of x_3 , effectively inhibiting x_1 downstream.

We consider the following representative model of the new structure in Figure 2.27:

$$\begin{aligned}
 \frac{\partial}{\partial t} x_1 &= \frac{v_4}{1 + x_3^p + \alpha x_4^p} - x_1 \\
 \frac{\partial}{\partial t} x_2 &= \frac{v_1}{1 + x_1^p} - x_2 \\
 \frac{\partial}{\partial t} x_3 &= \frac{v_2}{1 + x_2^p} - x_3 \\
 \frac{\partial}{\partial t} x_4 &= \frac{v_3}{1 + x_3^p} - x_4 + d_4 \frac{\partial^2}{\partial \xi^2} x_4
 \end{aligned} \tag{2.30}$$

where the concentrations x_i , $i = 1, \dots, 4$, and all other variables and parameters are again

non-dimensional. In particular, the time variable t is scaled to bring the degradation constants (assumed to be identical for each species) to one, and the one-dimensional length variable ξ is scaled so that the spatial domain is $\Omega = [0, \pi]$. We assume only the fourth species is diffusible (depicted with wavy lines in Figure 2.27) and is subject to zero-flux boundary conditions. We follow the same analysis procedure as outlined in Section 2.4.1.

The Jacobian matrix about the steady state $(\bar{x}_1, \bar{x}_2, \bar{x}_3, \bar{x}_4)$ yields:

$$J = \left[\begin{array}{ccc|c} -1 & 0 & -c_3 & -c_4 \\ -b_1 & -1 & 0 & 0 \\ 0 & -b_2 & -1 & 0 \\ \hline 0 & 0 & -b_3 & -1 \end{array} \right], \quad (2.31)$$

where:

$$b_1 = \frac{pv_1\bar{x}_1^{p-1}}{(1+\bar{x}_1^p)^2}, b_2 = \frac{pv_2\bar{x}_2^{p-1}}{(1+\bar{x}_2^p)^2}, b_3 = \frac{pv_3\bar{x}_3^{p-1}}{(1+\bar{x}_3^p)^2}, c_3 = \frac{pv_4\bar{x}_3^{p-1}}{(1+\bar{x}_3^p + \alpha\bar{x}_4^p)^2}, c_4 = \frac{\alpha pv_4\bar{x}_4^{p-1}}{(1+\bar{x}_3^p + \alpha\bar{x}_4^p)^2},$$

and $D = \text{diag}\{0, 0, 0, d_4\}$. We let J_{osc} be the upper 3×3 principal submatrix of J (delineated above), corresponding to the oscillator loop.

Condition 1: J_{osc} is unstable.

For the oscillator subsystem to be unstable, we need:

$$B \triangleq b_1 b_2 c_3 > 8 \quad (2.32)$$

so that the characteristic polynomial of J_{osc} , given by $(\lambda + 1)^3 + B$, has a pair of complex conjugate roots in the right half-plane. Note that this condition, though nearly identical in form, is slightly different from (2.12) due to c_3 .

Condition 2: J is stable.

For stability of the full reaction network, we let $C \triangleq b_1 b_2 b_3 c_4$ and find that we need:

$$B < 20 \text{ and } \frac{B^2 - 64}{16} < C < B + 1 \quad (2.33)$$

so that $\det(\lambda I - J) = (\lambda + 1)[(\lambda + 1)^3 + B] - C$ has all roots in the left half-plane.

Condition 3: $J + \lambda_k D$ is unstable for some $k \geq 1$.

The k^{th} spatial mode $\cos(k\xi)$ becomes unstable when the polynomial:

$$\det(\lambda I - (J + \lambda_k D)) = (\lambda + 1 + k^2 d_4)[(\lambda + 1)^3 + B] - C \quad (2.34)$$

has right half-plane roots. Indeed, when the product $k^2 d_4$ is sufficiently large, three roots of (2.34) approach those of $(\lambda + 1)^3 + B$, which contain roots with positive real part due to

(2.32). This means that the inhomogeneous modes $\cos(k\xi)$ grow over time if k^2d_4 exceeds the threshold for instability of the polynomial (2.34) and that this architecture is able to exhibit diffusion-driven instability for a large diffusion coefficient d_4 or high wave numbers k .

The parameters $p = 3$, $v_1 = v_2 = v_3 = 6$, $v_4 = 5.65$, and $\alpha = 0.08$ satisfy conditions (2.32) and (2.33) with $B = 10.083$ and $C = 3.369$. The polynomial (2.34) becomes unstable when $k^2d_4 > 1.261$. Simulations with $d_4 = 1$ indeed exhibit growth of the spatial inhomogeneity when the steady state is perturbed by adding the second wave ($k = 2$) with a small amplitude (Figure 2.28). This Turing behavior is contrasted to the decay of the initial inhomogeneity for the first wave (Figure 2.29) or in the absence of diffusion (Figure 2.30). The bifurcation diagram for these parameters is shown in Figure 2.31.

The full proposed synthetic implementation of the ZFP-sRNA quenched oscillator system can be seen in Figure 2.32.

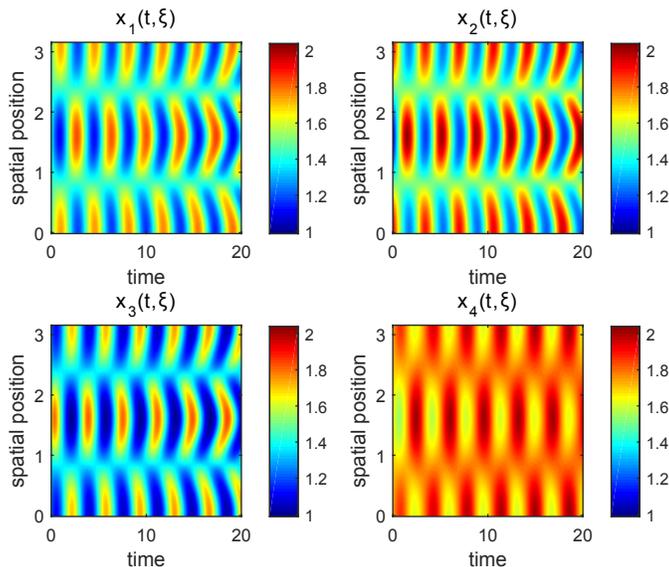


Figure 2.28: Solution of (2.30) on $\Omega = [0, \pi]$ with diffusion and growth. Using parameters $p = 3$, $v_1 = v_2 = v_3 = 6$, $v_4 = 5.65$, $\alpha = 0.08$. Here $d_4 = 1$ and $k = 2$ (wavelength π). Perturbation in x_1 of amplitude steady state $\pm 33\%$ peak-to-peak causes the inhomogeneity to grow as $k^2d_4 = 4 > d_{thresh} = 1.261$.

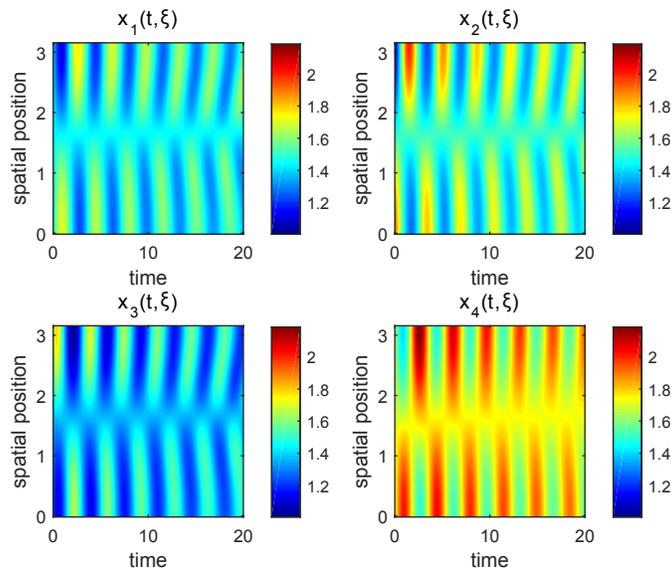


Figure 2.29: Solution of (2.30) on $\Omega = [0, \pi]$ with diffusion and decay. Using parameters $p = 3$, $v_1 = v_2 = v_3 = 6$, $v_4 = 5.65$, $\alpha = 0.08$. Here $d_4 = 1$ and $k = 1$ (wavelength 2π). Perturbation in x_1 of amplitude steady state $\pm 33\%$ peak-to-peak decays towards the steady state as $k^2 d_4 = 1 < d_{thresh} = 1.261$.

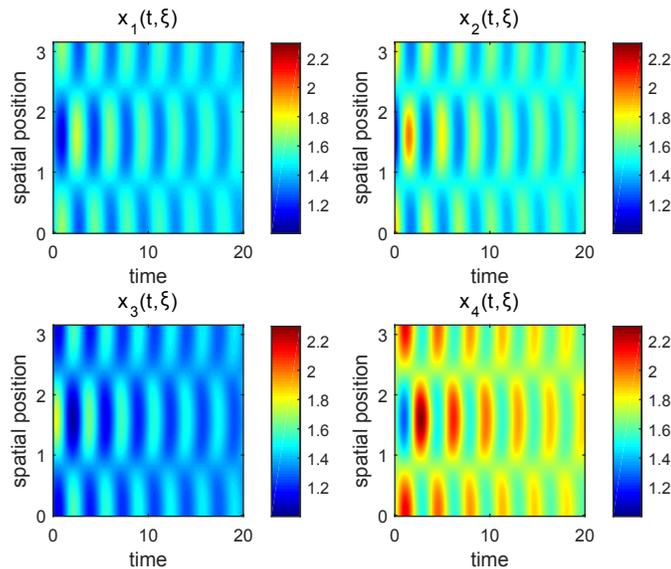


Figure 2.30: Solution of (2.30) on $\Omega = [0, \pi]$ without diffusion. Using parameters $p = 3$, $v_1 = v_2 = v_3 = 6$, $v_4 = 5.65$, $\alpha = 0.08$. Here $d_4 = 0$ and $k = 2$ (wavelength π). Perturbation in x_1 of amplitude steady state $\pm 33\%$ peak-to-peak decays towards the steady state as all cells are stable.

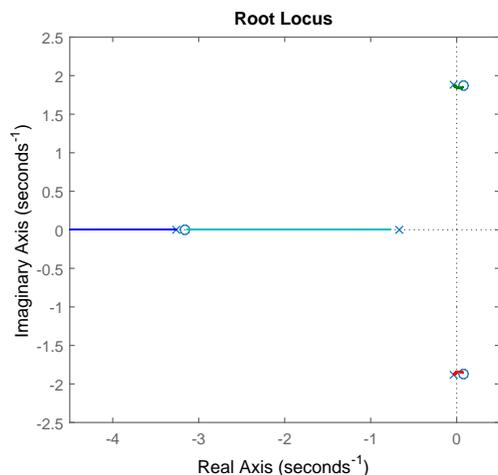


Figure 2.31: Bifurcation diagram for parameters $p = 3$, $v_1 = v_2 = v_3 = 6$, $v_4 = 5.65$, $\alpha = 0.08$. The positions of the system eigenvalues without diffusion are shown with \times 's and their limits as the gain $k^2 d_4 \rightarrow \infty$ are shown with \circ 's. For this parameter set, Turing patterning is achieved once two eigenvalues cross the imaginary axis when $k^2 d_4 > d_{thresh} = 1.261$.

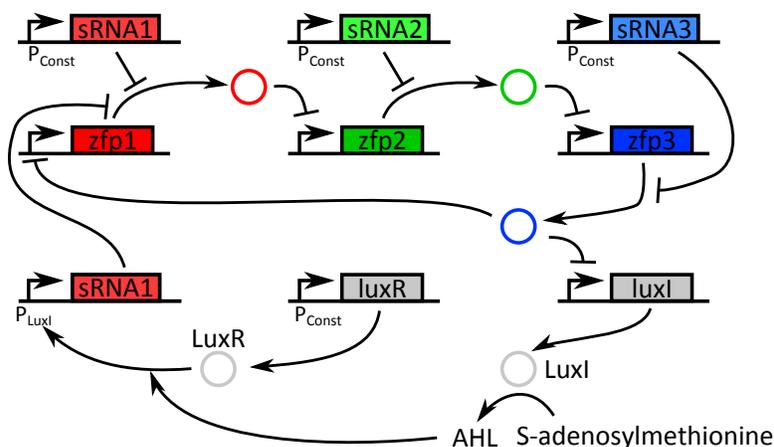


Figure 2.32: Our updated synthetic implementation of the network in Figure 2.27 using ZFPs and sRNAs. The quenching loop contains the membrane-diffusible signaling molecule 3-oxohexanoyl-homoserine lactone (3OC6HSL), which we refer to simply as AHL, and interacts with the ZFP-sRNA oscillator via production of an sRNA.

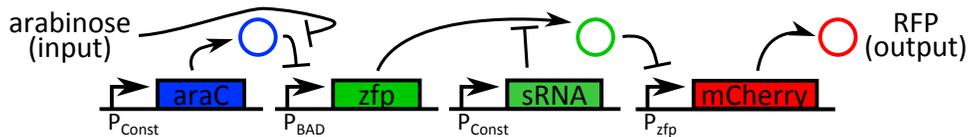


Figure 2.33: Schematic of the new ZFP-sRNA inverters. An inducible promoter is used to vary the production of ZFP, which represses the production of a measured fluorescent protein. The mRNA of the ZFP is targeted by a constitutively-produced sRNA.

2.7 ZFP-sRNA Inverter Experimental Results

As described in Section 2.6.2, we attempted to make three ZFP-sRNA inverters to put together into a ring oscillator. A system as complicated as the quenched oscillator is certainly a massive undertaking and the creation of a family of orthogonal and tunable parts would be a step towards that goal while also being a significant contribution on its own. Having no prior laboratory experience in biology, I was trained by William Holtz, whose Ph.D. dissertation was on the design and engineering of ZFPs as transcriptional repressors in *E. coli* [61]. The work performed here was built directly on top of his dissertation work.

2.7.1 Experimental Design

A schematic of our ZFP-sRNA inverters is shown in Figure 2.33. An inducible promoter is used to vary the production of a ZFP, which represses the production of a fluorescent protein. Here we examine the effect of targeting the ZFP mRNA with a constitutive amount of sRNA. We showed in Section 2.6.2 that the introduction of sRNA generates an ultrasensitive response with higher gain, but here we show the effect on our expected experimental results.

We update our model to match our inverter schematic by changing the input to a constant level of ZFP mRNA production, as would be set by our chosen inducer concentration, and our output to be the fluorescent protein concentration:

$$\begin{aligned}
 \frac{d}{dt}s &= V_s N_s C - k_f s m_z - \gamma_s s \\
 \frac{d}{dt}m_z &= V_{in} - k_f s m_z - \gamma_m m_z \\
 \frac{d}{dt}p_z &= \epsilon_z m_z - \gamma_p p_z \\
 \frac{d}{dt}m_r &= V N C \left(\frac{1}{1 + (p_z/K)^n} + \ell \right) - \gamma_m m_r \\
 \frac{d}{dt}p_r &= \epsilon_r m_r - \gamma_p p_r,
 \end{aligned} \tag{2.35}$$

where the subscripts z and r refer to the ZFP and the reporter (or RFP), respectively, and the state variables s , m , and p represent sRNA, mRNA, and protein. The parameters V

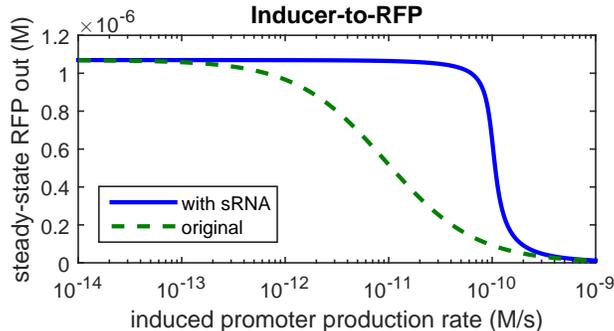


Figure 2.34: Steady-state input-output map of ZFP inverter showing the effect of adding sRNA to the system. The sRNA thresholding causes the repression to take effect at a higher inducer level and with a sharper slope.

and V_s are velocity constants, N and N_s are copy numbers, k_f is the forward binding rate of sRNA and ZFP mRNA, K is the dissociation constant for ZFP-DNA binding, ℓ is the leakage rate normalized to V , γ_i are degradation rates, and ϵ_i are the protein translational rates. The parameter C is the concentration level generated by a single molecule in an *E. coli* cell.

Using the same notation as in Section 2.6.2, we compare the steady-state input-output function $\bar{p}_r(V_{in})$ with and without sRNA.

Without sRNA:

$$\bar{p}_{r,-s} = \frac{\epsilon}{\gamma_m \gamma_p} \phi_1(V_{in}) \quad (2.36)$$

With sRNA:

$$\bar{p}_{r,+s} = \psi(\phi_1(V_{in})) \quad (2.37)$$

The comparison of these two input-output curves can be seen in Figure 2.34, where the sRNA thresholding causes the repression to take effect at a higher inducer level and with a sharper slope.

2.7.2 Parts Selection

Our main consideration in selecting parts was to construct a ring oscillator made of three ZFP-sRNA inverters, which has two major implications. First, orthogonality of parts is very important, as large amounts of crosstalk amongst oscillator species make oscillations more difficult to achieve. Secondly, we want to make our individual inverters as similar in behavior as possible. It is well-known that ring oscillators achieve their largest oscillating parameter region when using identical inverters and in practice it makes input-output matching between inverters easier.

The ZFPs were taken directly from [61] and no attempt was made to modify them. Nomenclature and numbering are preserved from William Holtz's dissertation and plas-

promoter	op	zinc finger protein					target sequence	ZFP number
		-40	-30	-18	-12	-3		
125	-40	0.14	0.95	0.97	1.00	0.93		
83	-30	1.00	0.17	0.93	0.97	0.93	TAGTGAAGGAATGGGAG	16-56
69	-18	1.00	0.95	0.21	0.94	0.96	TGGGAGATAGTGGGAGAG	16-57
125	-12	1.00	1.00	0.88	0.26	0.91		
83	-3	1.00	0.92	0.88	0.82	0.08	GAGAGGAAGGAGAGGAG	16-59

Figure 2.35: “Highly orthogonal set of 5 promoter-repressor pairs” Table 5-5 reproduced on the left from [61]. Our three chosen ZFP binding sites, op-3, op-18, and op-30, are highlighted in red and their respective nucleotide recognition sequences and targeting ZFP numbers are shown on the right. Note that op-18 is used on a different BIOFAB promoter than op-3 and op-30.

mids. We used the ZFP binding sites op-3, op-18, and op-30 and their corresponding ZFP 16-59, ZFP 16-57, and ZFP 16-56 from a set of orthogonal promoter-repressor pairs (Figure 2.35). It is important to note that this set of promoter-repressor pairs was built on top of constitutive synthetic promoters from the BIOFAB library [100] and that op-3 and op-30 are placed on BIOFAB RPL-83 while op-18 is placed on BIOFAB RPL-69. Note that these names were taken from the BIOFAB Randomized Promoter Library (RPL) version 1. The most recent BIOFAB promoter library can be found on their data access web service (<http://biofab.synberc.org/data/docs/daws?q=data/docs/daws>) under “Annotated Parts.” In this database, RPL-83 is named apFAB237, while RPL-69 is not found. For clarity, the promoter sequences used are given in Table 2.7.

We investigated the use of three separate sets of sRNAs: one based on the insertion sequence IS10 [101], one based on the plasmid pT181 mechanism [89], and one based on the use of a MicC scaffold [102]. The IS10- and pT181-based sRNAs operate based on sense region and antisense pairings, while the scaffold-based system works with a designable 24 base pair target-binding sequence. Despite designing a scaffold-based sRNA that successfully bound to the mRNA of a reporter RFP, we were unable to design one that bound to one of our ZFPs (data not shown). Additionally, the presence of repeated regions in each individual

Table 2.7: Synthetic promoters used in ZFP-sRNA inverters

Library	Name	Promoter Sequence	Rel Strength [114]
BIOFAB	RPL-69	TTGACAATTAATCATCGGCTCGTAGGGTATGTGGA	n/a
BIOFAB	RPL-83	TTGACAATTAATCATCGGCTCATAACCTTTGTGGA	n/a
Anderson	J23106	TTTACGGCTAGCTCAGTCCTAGGTATAGTGCTAGC	0.47
Anderson	J23108	CTGACAGCTAGCTCAGTCCTAGGTATAATGCTAGC	0.51
Anderson	J23118	TTGACGGCTAGCTCAGTCCTAGGTATTGTGCTAGC	0.56

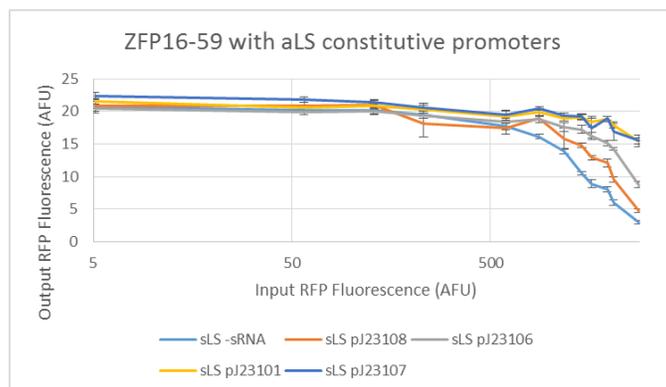


Figure 2.36: Sample inverter data showing effect of different constitutive promoters from the Anderson library. More constitutive promoters were tested than are shown here. The magnitude of effect of the sRNA did not match perfectly with the listed relative strengths of the promoters from [114] and a number of the promoters resulted in loss of inverting behavior like J23101 and J23107 shown here, likely due to excessive amounts of sRNA in the system.

finger of our ZFPs would have required the addition of a number of hidden point mutations to hopefully produce a set of three orthogonal target regions. A set of five orthogonal IS10-based sense-antisense pairs were identified in [101] and numbered 4, 5, 31, 34, and 49. In our hands, pairs 31, 34, and 49 worked much worse than pairs 4 and 5 to the degree that we were unsure if the sRNA binding was working at all. A set of three orthogonal pT181-based sRNAs were presented in [89] and labeled WT, LS, and LS2. The use of these sense regions tended to result in a lower induction fold (i.e. maximum expression level over minimum expression level) of our inverters. Needing just one more sRNA, we proceeded with the best-performing sRNA of the three, which was LS. Note that the 5'-UTR used with a pT181-based sense region includes a RepC minicistron. From here on, we will refer to our chosen set of sense regions as s04, s05, and sLS and our sRNAs (antisenses) as a04, a05, and aLS.

The sRNAs are produced at a constitutive level, so we looked for three constitutive promoters of roughly equivalent strength. Placing all of the sRNAs on the same operon is also an option, but we decided against it because of polymerase fall off (likely a negligible effect) and the high amount of similarity between sRNAs from the same set (e.g. a04 and a05). Placing them onto separate promoters allows us to space them out and place them in opposing directions as needed. We chose the promoters J23106, J23108, and J23118 from the Anderson library of synthetic promoters [114] (see Table 2.7). We tested the promoters from the library with listed relative strengths close to J23108, but found that many provided undesired behavior (see Figure 2.36). Altering the promoter strength (and/or copy number) of the sRNAs would change the degree of the effect we expect to see. We were looking for an intermediate level where the thresholding effect of the sRNAs would be clearly visible without becoming so sharp that we would miss the repression effect with our finite number of inducer levels, especially since induction is never quite linear.

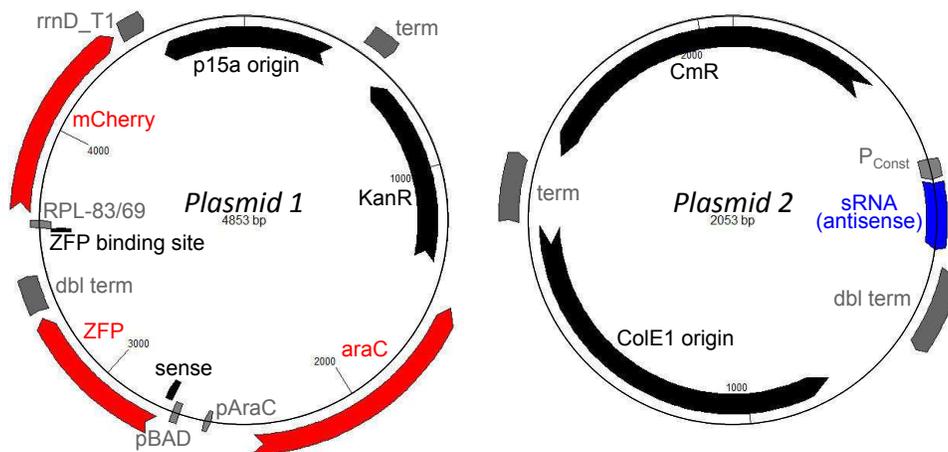


Figure 2.37: Two-plasmid design for ZFP-sRNA inverters, with a ZFP inverter on Plasmid 1 (left) and the sRNA on Plasmid 2 (right). To build different inverters, we can swap out the ZFP and sRNA, but we must make sure that we also have the proper ZFP binding site, sense region, and output promoter. Experimental strains were made by co-transforming one variant of Plasmid 1 with either a variant of Plasmid 2 or an empty CmR plasmid (see Table 2.8) into DP10 cells.

We investigated small RBS libraries to use with the different sense regions in order to try to maximize inverter induction fold without affecting sRNA binding significantly. For the IS10-based sense regions, we looked at a two-base pair library in the location of the mutated Shine-Dalgarno region identified in [101] and identified TC as the optimal RBS. For the pT181-based sense regions, we looked at a library of 256 using the nucleotide sequence “RRRRNN” starting 14 bp before the beginning of the ZFP start codon, where R stands for A or G and N is any nucleotide. The best we found was AAAGGA.

We planned from the start on using P_{BAD} as our inducible promoter with arabinose. We did test swapping in $P_{LtetO-1}$ and $P_{LlacO-1}$ as well, but did not see any significant improvements (data not shown) to justify switching over permanently. To eliminate the “all-or-none” behavior of the P_{BAD} promoter, we had to select an experimental strain of *E. coli* that expresses *araE* [72]. We used the DP10 strain from the Keasling lab [77].

For our output protein, we chose mCherry because it is monomeric, matures quickly, and is photostable [120]. mCherry was placed on the synthetic promoter-ZFP binding site pairings developed by William Holtz, so we used a very strong 5'-UTR in the Bujard RBS [90].

2.7.3 Plasmid Design

We decided on a two plasmid system in order to separate the ZFP inverter from the sRNA (see Figure 2.37). Given the combinatorial nature of the planned work, this would allow us to do cloning on the two parts of our system independently and then easily combine

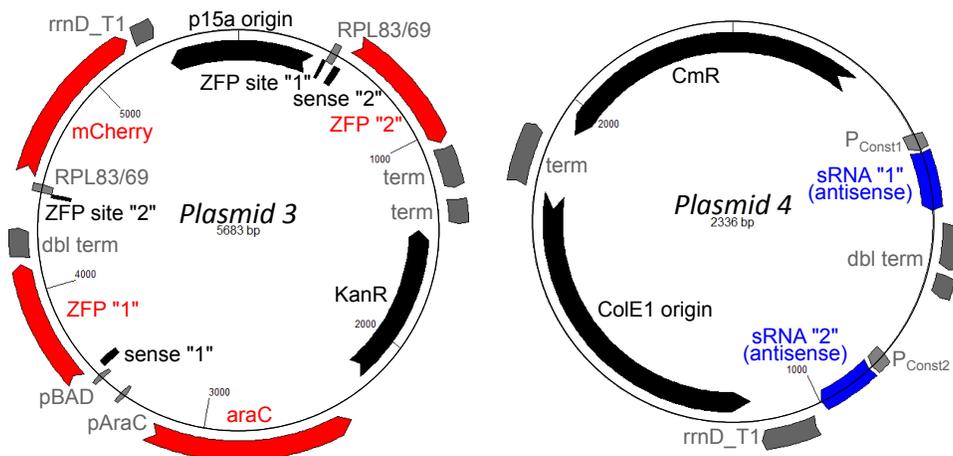


Figure 2.38: Two-plasmid design for two ZFP-sRNA inverters in series. Similar to Figure 2.37, the ZFP inverters are on Plasmid 3 (left) and the sRNAs are on Plasmid 4 (right). As shown here, the ZFP on P_{BAD} targets the ZFP between the origin of replication and resistance gene, which then targets mCherry. Other placements were not attempted. A single variant each of Plasmid 3 and Plasmid 4 were constructed (see Table 2.8) before we determined that getting two ZFP-sRNA inverters in series working was unlikely with our chosen set of parts.

them as desired. In addition, splitting the plasmids would make cloning more manageable for future experiments involving two or three inverters by reducing the individual plasmid sizes, number of operons, and number of repeated sequences.

Due to toxicity issues that arise when expressing high concentrations of ZFPs in *E. coli*, we opted for the low copy number origin of replication p15a for the inverter plasmid. We also tried the very low copy number origin of replication SC101**, but saw worse behavior (data not shown). The sRNAs were placed on a higher copy number plasmid ColE1 with the constitutive promoter used for tuning.

We used the native P_{BAD} and P_C promoters, which are found adjacent to each other in opposite directions, for the production of the ZFP and AraC, respectively. Then the remaining choice is the placement of the reporter operon. While putting it on the higher copy origin might result in higher output strength that is more distinguishable from noise, keeping it with the rest of the inverter operons is simpler and allows us to run inverter experiments without sRNA in single antibiotic conditions and saves us a co-transformation step. We place the mCherry operon in the same direction as the ZFP operon to avoid any possible mRNA hybridization issues, even though that is not as much of a concern in prokaryotes.

When putting two inverters in series, we decided to keep the same two plasmid split with the inverters on one plasmid and the sRNA on the other plasmid (see Figure 2.38). Given that each ZFP is six fingers long and contain repeated sequences between every finger, we decided to separate the two ZFP operons as much as possible to avoid replication slippage.

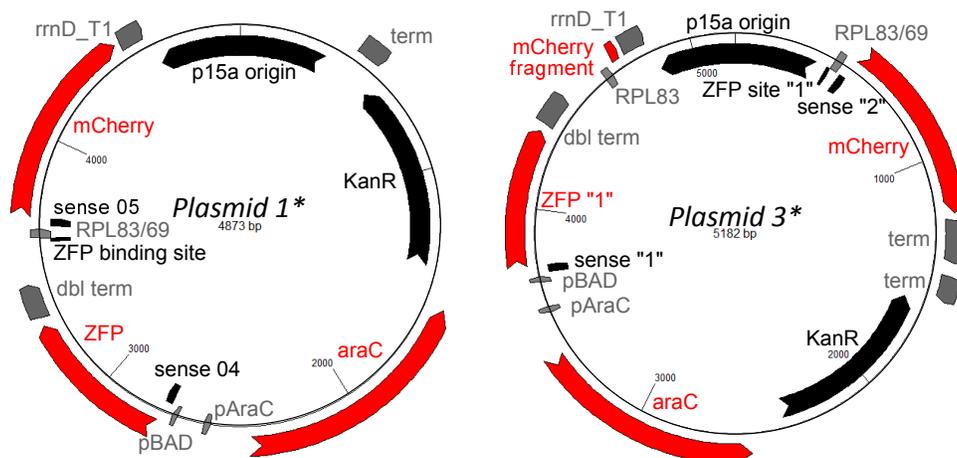


Figure 2.39: Modified versions of inverter plasmids to test the viability of two inverters in series. Left: modified version of Plasmid 1 to test the effect of the 5'-UTR that includes the sense region to our reporter. Right: modified version of Plasmid 3 to test how well the intermediate operon works in the location specified between the origin of replication and the resistance gene.

By placing the second operon in-between the origin of replication and resistance gene we naturally select against instances of slippage between the ZFPs in either direction. For the two sRNA plasmid, direction and placement was not as important. While sRNAs of the same type (IS10- or pT181-based) have similar sequences, they would constitute a single repeated sequence. Issues of context effects and terminator read-through could be counteracted by changing the constitutive promoters as needed, so we simply added the second sRNA operon downstream with a different terminator.

By choosing to keep the reporter operon in the same location, we are introducing an intermediate operon in a new location with unknown context effects. So we designed two test plasmids to help gauge the viability of our two inverters in series design (see Figure 2.39). We first constructed Plasmid 3* to see how well the new intermediate operon performed. To accomplish this, we replaced the second ZFP with our reporter mCherry and removed most of the original reporter, leaving just a small fragment so it would not fluoresce. Later on, we constructed Plasmid 1* to isolate the effect of the 5'-UTR containing a sense region on our output.

For a full list of the plasmids used in these experiments, see Table 2.8 in the Materials and Methods section below.

2.7.4 Inverter Results

Inverter input-output functions (confusingly called “transfer functions” in the biology community) are shown in Figure 2.40. Data for nine different inverters is shown, each inverter with one of three ZFPs paired with one of three sRNA sense regions, as well as the

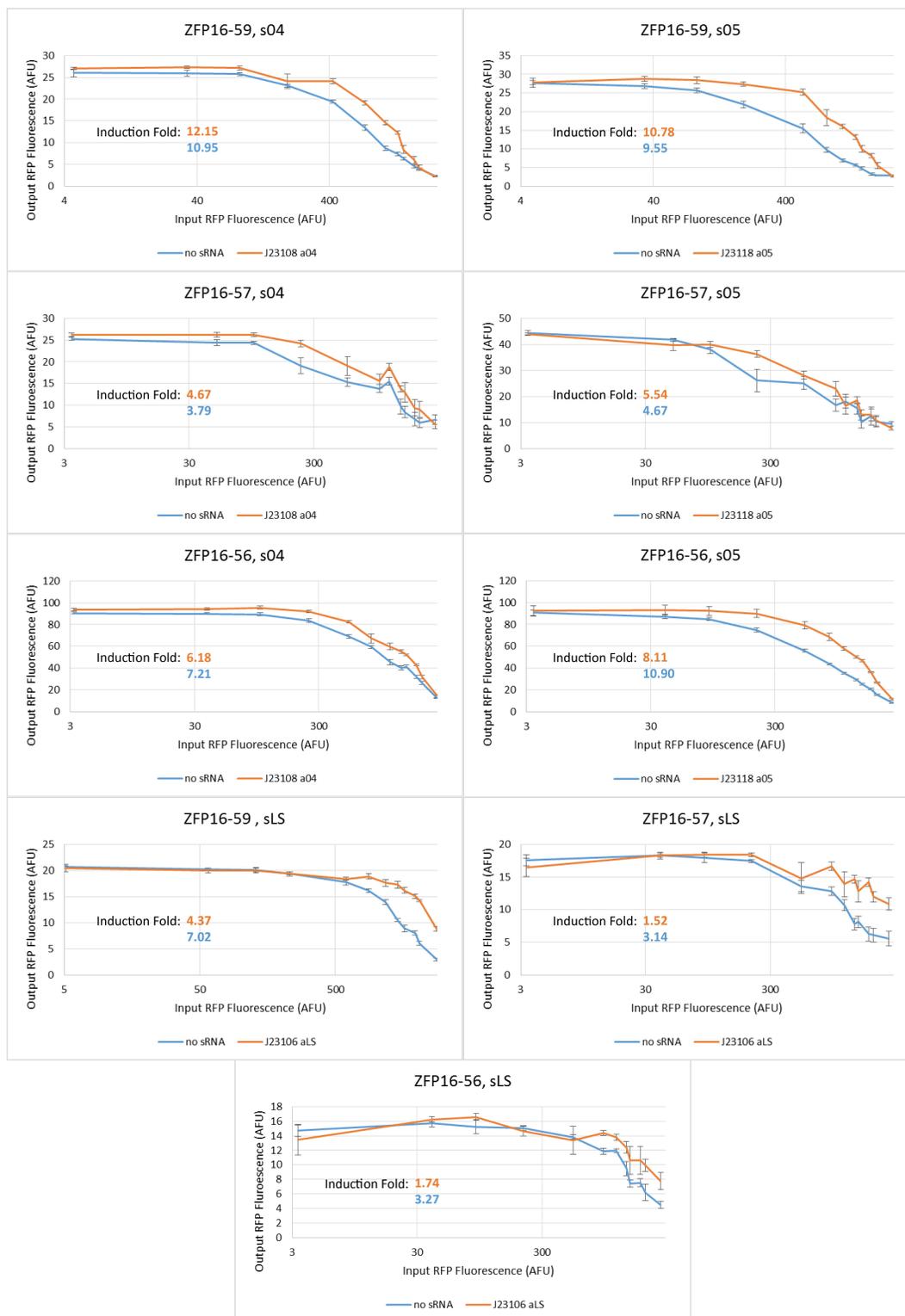


Figure 2.40: Inverter data for 3 ZFPs and 3 sRNAs in combination with induction folds.

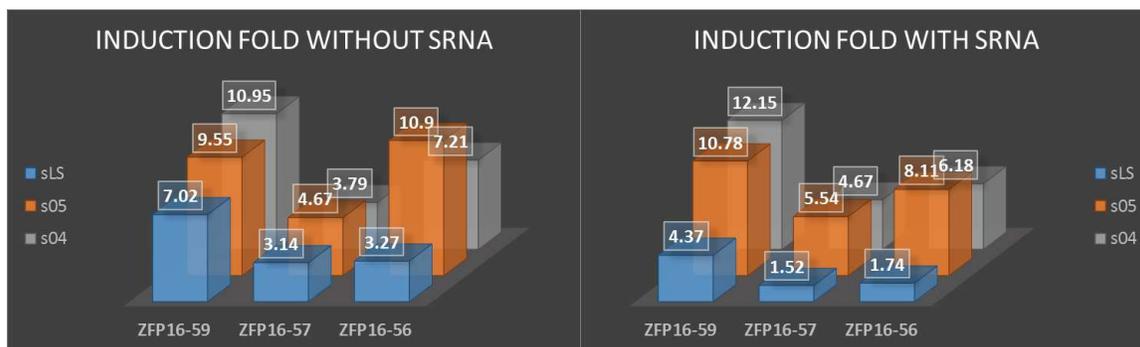


Figure 2.41: Grid visualization of induction folds of our inverter data from Figure 2.40 to help make observations in Section 2.8.2. Induction fold is one indicator of how well our inverter is working since it lets us know how significant the difference between “on” and “off” states are.

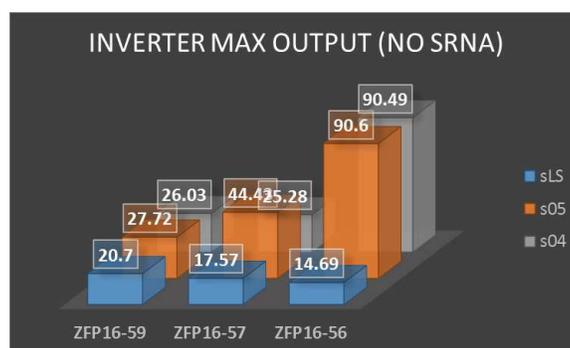


Figure 2.42: Grid visualization of maximum output of our inverter data from Figure 2.40 to help make observations in Section 2.8.2. The maximum output with sRNA is nearly identical, so is not shown. Maximum output is not always indicative of how well an inverter is working, but can be used as a selection criterion and in this case allows us to examine context effects of ZFP binding sites on our reporter promoter.

induction fold (maximum output/minimum output). Each plot corresponds to one of these inverters in the presence and absence of a constitutive amount of the corresponding sRNA. In addition, we’ve visualized some of the numbers in Figures 2.41 and 2.42 to help us make observations.

Inverter orthogonality tests were not run because the orthogonality of the individual components were demonstrated in previous work [61, 101, 89]. Using multiple inverters together would also demonstrate viable levels of orthogonality.

For two inverters in series, we expect the steady-state output to act as a delay buffer where low induction levels lead to low output fluorescence and high induction levels lead to high output fluorescence. Initial experiments with two inverters in series (no sRNA) showed no change in output across the induction range (data not shown).

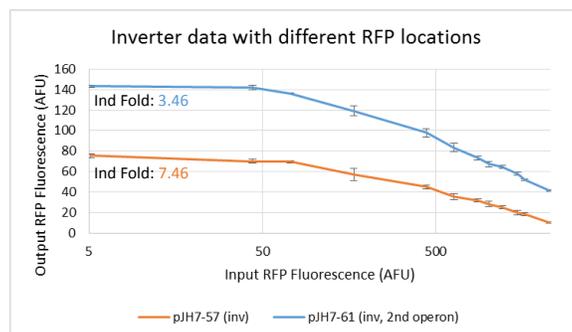


Figure 2.43: Inverter data showing effect of 2nd operon location between the origin of replication and resistance gene. The induction fold dropped from 7.46 to 3.46. There were two major differences: (1) The location of the reporter operon and (2) the 5'-UTR on mCherry. We ran further tests to help separate the two effects.

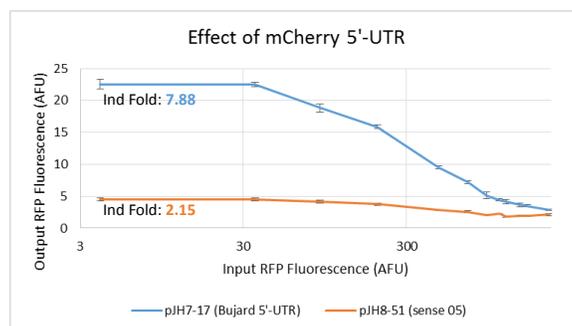


Figure 2.44: Inverter data showing effect of sense region 5'-UTR. Swapping the Bujard 5'-UTR on the output RFP with one including an IS10-based sense region drastically reduces the induction fold from 7.88 to 2.15.

In the process of debugging, we ran experiments with our test plasmids. While testing our new intermediate operon location (Plasmid 3*), we saw a reduction in induction fold, but could not determine if this was caused by the new operon location, the change in 5'-UTR, or both (Figure 2.43). To help clarify, we tested just a change in 5'-UTR on our reporter (Plasmid 1*) and saw that even just the introduction of the sense region greatly reduced our induction fold (Figure 2.44). Based on this evidence, we concluded that with our current set of parts two inverters in series was unlikely to work due to inverter input-output mismatch.

2.8 Turing Patterning Discussion

The motivation behind this project was to develop a synthetic gene network to exhibit Turing patterning. While we were able to devise a new quenched oscillator system, our final design (Figure 2.32) involved the production of nine (including AHL) different molecules

within each cell, so we knew that the likelihood of getting a full experimental implementation to work given the current state of synthetic biology were slim. Still, the hope was that we could begin to put parts of our design together experimentally while also developing new parts for the synthetic biology community to use.

2.8.1 Implications of the Novel Architecture

The engineering of cooperative ensembles of cells, whether in the context of designer microbial communities or other synthetic multicellular systems will require tractable model systems which exhibit spontaneous symmetry breaking and pattern formation, both fundamental prerequisites for any kind of replicating or “programmed” heterogeneity of form or function. Attempts to produce spontaneous pattern formation using Turing’s canonical system have proven difficult. We alleviated some of the difficulties arising in the activator-inhibitor model by using oscillator subsystems instead. To our knowledge, this is the first attempt of this kind and significant effort was devoted to providing researchers with an experimentally tractable road map towards implementation.

Our modified quenched oscillator system was made possible by recent work with ZFPs [61] that allow for the construction of nearly identical, orthogonal transcriptional repressors. Taking advantage of the ability of sRNAs to bind with ZFPs to produce high inverter gains, we attempted to construct a new ring oscillator. Next we plugged this ZFP-sRNA oscillator into a quenched oscillator system using a new quenching loop and demonstrated this system’s ability to produce Turing patterning.

This work also implicitly suggests that natural systems may have arisen where oscillating subsystems, initially evolved for other purposes, provide the backbone not just for coordinated oscillation (as in the diffusively coupled systems demonstrated by others [33, 98, 47]) but for robust Turing-type pattern formation phenomena. It is not difficult to find examples in the literature of naturally-occurring coupled negative feedback oscillators, both in prokaryotes [17] and eukaryotes [87, 32]. A function as fundamental as cell cycle oscillation appears to be maintained in yeast and other eukaryotes by coupled oscillators (a negative feedback oscillator coupled to a relaxation oscillator) [32]. Going further, these motifs are also present in protein-protein systems [73]; while outside the scope of the present work, the general results presented (i.e. coupled multi-step negative feedback oscillators with one diffusible component can exhibit Turing instability) would likely apply to kinase loops [73]. Lastly, in our model the relative phase lag between the oscillator loop and the quenching loop affect both the emergence and wave numbers of pattern; these, in turn, depend on the relative number of “steps” around the loops. It is tempting to suggest that the alteration of the number of steps, or the total delay around the loop, could provide a mechanism by which adaptation and evolution could generate systems (and variants) capable of pattern formation.

2.8.2 Inverter Performance

In order to put a three-component ring oscillator together, we selected three ZFPs and three sRNAs and demonstrated their modularity by running steady-state fluorescence experiments on all nine different combinations. Individually, every ZFP-sRNA inverter did what it was supposed to do: produce “high” output at “low” input levels and “low” output at “high” input levels. Additionally, we were able to demonstrate the expected qualitative change in behavior with the addition of a constitutive amount of sRNA in the system (compare Figure 2.34 to Figure 2.40).

Beyond the qualitative effects, how useful might these inverters be to the synthetic biology community? The highest induction fold achieved was around 12 (91% repression), which is enough to be useful depending on the application and compares favorably to the ZFP repressors in isolation (see Figure 2.35 and [61]). However, there certainly are better-performing repressors available such as $P_{LtetO-1}$, $P_{LlacO-1}$, and $P_{lac/ara-1}$, which have induction folds >600 [90]. But right now the synthetic biology community is limited to a small number of known promoter-repressor pairs, and ZFPs add to this library. In addition, the modularity of ZFPs allow us to target promoters that do not have known repressors, such as the growing library of synthetic constitutive promoters.

The goals of building ZFP-sRNA inverters were to develop a “large” library of orthogonal parts with nearly-identical parameters, as these would aid in the construction of a new synthetic ring oscillator. While neither goal was quite achieved and the ring oscillator has yet to come together, there is much to be learned from the effort. In an attempt to avoid revisiting the design work on the ZFPs, we chose to investigate the minimal number of parts needed for our ring oscillator. Even then, the parts chosen displayed an unexpected amount of variation in behavior. We make the following observations:

Induction Fold:

In Figure 2.41, we use induction fold as a loose measure of how well our inverters are working as it is an indicator of how different our “on” and “off” states are. Another important metric would be the slope of the linear region of the inverter input-output map, but given the limited number of induction levels used, this value is difficult to determine from our data.

In terms of effect on induction fold, the ZFPs were generally ordered with ZFP16-59 performing the best and ZFP16-57 performing the worst. ZFP16-56 did seem to work surprisingly well with s05. s04 and s05 performed comparably in regards to induction fold and sLS performed significantly worse. Additionally, the data with sLS and aLS suffered in induction fold because the output was unexpectedly high at maximal induction (whereas s04 and s05 achieved nearly the same minimum value with and without sRNA present). Based on the data of different constitutive promoters (Figure 2.36), it is more likely that this was caused by the choice of J23106. As mentioned previously, the Anderson library of constitutive promoters did not exhibit the expected ordering of relative promoter strengths as provided [114]. These trends did seem pretty consistent and the worst induction fold was achieved with the combination of ZFP16-57 and aLS, as expected.

Attempting to comment on the possible sources of these differences is difficult, since the induction fold is affected by nearly every interaction in our inverter. The most consistent trend that we can point out is the ordering of the ZFPs in that with one exception, using ZFP16-59 outperformed ZFP16-56, which in turn outperformed ZFP16-57. This would suggest a general ordering of the binding affinity and repressive strengths of the ZFPs to their targeted promoters, but we can't draw any strong conclusions due to our small data set and our inability to differentiate between the two effects.

Maximum Output:

Figure 2.42 shows the maximum output level of our inverters without sRNA. There was very little difference in maximum output with and without sRNA, which is expected for low induction levels, so we didn't show the other set of data. Maximum output is not a good measure of inverter performance but may allow us to infer some context effects between our inverter parts. In particular, maximum output occurs in a region with low induction, meaning the effects of the ZFP and sRNA should be minimal. Then the data here should reflect only the effects of the ZFP binding sequence on the target promoter and the choice of promoter (RPL-69/83).

Unfortunately because we could not find RPL-69 in the online BIOFAB data set, we can't determine the expected relative strength of those two promoters. The output expression level was significantly higher when using ZFP16-56 compared to ZFP16-59 and ZFP16-57, except oddly when paired with sLS. This makes even less sense when we remember that the sRNA and its binding site should not affect maximal expression. Given that both the maximal and basal levels were elevated (i.e. the induction fold did not change as significantly), this behavior is likely caused by context effects of having the ZFP op-30 binding site upstream of the target constitutive promoter as opposed to op-3 or op-18. There certainly could be uses for boosting the output levels this way while maintaining the inverter behavior.

We believe that there are two main reasons that putting two inverters in series was unsuccessful. The first is the poor induction fold each individual inverter and the second is the input-output mismatching between inverters. One thing we had not considered was the effect of the sense regions on their target promoter behavior. Each inverter was run with RFP having the Bujard 5'-UTR, which is known to be very strong. However, once we assemble multiple inverters in series, the intermediate operons must have sense regions instead. When we finally measured the effect (Figure 2.44), there was a drastic reduction in promoter output range, which would affect the intermediate ZFP variations in concentration.

Secondly, the inverting behavior is seen in the very top portion of the induction range. In order for the ring oscillator to work, we need the steady-state solution to lie in the linear regime of the input-output map, so that changes in the input level around the steady state cause significant changes in the output as well. Since the linear regime is in the upper input range, we would need each inverter to be able to produce a very large maximal amount of ZFP, which is unlikely given the effect of the sense region on the promoter. The sense region further compounds the problem by reducing the inverter induction fold, meaning that the

input changes passed to the next inverter are of small magnitude.

Even though the ring oscillator doesn't work with the set of parts we investigated, the problem lies with the parameters of our chosen parts and not our design as all parts exhibit the desired behavior, but not well enough. With improved parts there's no reason to believe this work can't be completed.

2.8.3 Turing Patterning Future Work

Immediate future work will focus on getting the ring oscillator to work by either selecting better parts or improving the current ones. The chief components in question are the promoter-repressor pairs. If we decide to continue working with ZFPs in *E. coli*, a few of the different improvements to investigate include:

- Changing the amino acid sequences between fingers to not be exact nucleotide repeats will aid in construction of the ZFPs and future cloning involving them.
- Reducing the number of fingers used to five may help with cloning without reducing binding specificity too much.
- William Holtz did investigate optimal placement of the ZFP binding site relative to the promoter transcriptional start site [61], but did not investigate using multiple binding sites.
- It is clear that we need to use promoters that can provide us with greater induction fold. Although we would eventually want three promoters of roughly equivalent strength, we could try using known strong promoters (e.g. $P_{LtetO-1}$) or some of the many contained in the BIOFAB promoter library.
- Look for a better-performing set of orthogonal ZFPs than the ones found by William Holtz that we used here.

Another avenue would be to have the ZFPs constructed for us to save time. Active research on ZFP and zinc finger nuclease (ZFN) use in eukaryotes is carried out by the company Sangamo Biosciences [78]. They own a proprietary dataset of characterized finger pairs and can provide custom ZFPs for a fee, though the costs are significant.

ZFPs are not the only option for designable repressors. We simply need something that will bind to the DNA next to a promoter to block transcription. The main alternatives are CRISPRi [110] from recent work on clustered regularly interspaced short palindromic repeats (CRISPR) [69] and transcription activator-like effectors (TALEs) [18], both of which allow binding to an arbitrarily specified nucleotide sequence. TALEs are similar to ZFPs in that they are constructed from a sequence of repeats where two amino acids called the repeat-variable diresidues (RVDs) specify a single base that will be bound. So for binding the same sequence, a TALE will contain about three times as many repeated sequences as

a ZFP. In CRISPRi, we coexpress a catalytically dead CRISPR-associated system protein (dCAS9) with designable guide RNAs to form DNA recognition complexes. The general consensus is that ZFPs are the most difficult to design and construct properly but bind the most specifically. CRISPR systems are the easiest to design (only need to change the guide RNA), but suffer from off-target effects, which are mitigated in CRISPRi by the lack of endonuclease cutting activity. TALEs fall in the middle in terms of ease of design and construction and binding specificity.

Assuming that we can create a working ring oscillator, the next steps would be to add the quenching loop and then investigate different options for the patterning assay. As will be discussed more in-depth in Chapter 3, there are only a handful of well-studied AHLs that have been tried in *E. coli*. Thankfully this project only requires one and the *lux* system works well in our hands. As for the experimental assay, the two biggest concerns will be creating a fixed geometry and dealing with cell growth. In our models, we assumed we would be dealing with a one-dimensional line of cells that are fixed in space and communicating with each other. To achieve this effect, we would likely need to design a microfluidic device with a linear channel carved into it and attach bacteria to the surface of the channel. There are many ways to attach bacteria to a desired surface [130], such as sequence-specific DNA hybridization [131], and this would enable us to fix the positions of our cells of interest while allowing daughter cells to float or be flowed away. A clever method would need to be devised to help remove extraneous cells without adversely affecting AHL diffusion.

2.9 Turing Patterning Materials and Methods

2.9.1 Computational

Analytical models were investigated primarily in MATLAB Version 8.5.0 (R2015a) and some representative scripts are provided in the Appendix. All non-experimental data plots found in this Chapter were generated in MATLAB, even if the data came from another program.

Steady-state values for different sets of parameters for all models were numerically calculated using the function `fzero`. For systems with multiple steady states, it was important to make sure that this function returned a realistic (i.e. non-negative) set of steady-state values. Root locus plots were generated using the function `rlocus` and matrix eigenvalues were checked using the function `eig`.

Data from experiments were measured using the instruments specified in Section 2.9.6 and output to files. These files were opened in Microsoft Excel 2013 and the data was analyzed and plotted using standard Excel functions along with a few custom macros.

2.9.2 PDE Simulations

Continuous, deterministic models are useful because of the wide variety of analysis tools we can apply to them to generate predictions of system behavior and workable parameter spaces, which we cannot do for stochastic models. These models are accurate when the number of molecules for all species in the system are very large, but generally need to be supplemented with stochastic simulations for systems with small numbers of molecules. PDE simulations were run with the function `ode15s`, which is a multi-step, variable order solver based on numerical differentiation formulas. PDE simulations for the toy models were run in MATLAB Version 8.5.0 (R2015a) while PDE simulations for the quenched oscillator implementation were carried over from [62] and run in MATLAB Version 7.10.0 (R2010a). For line of cell simulations, diffusion was handled using a finite difference approximation with 101 evenly-spaced grid points and zero-flux boundary conditions. For single cell simulations, the long empty volume was represented using a finite difference approximation with Dirichlet boundary conditions of zero AHL concentration.

2.9.3 Stochastic Simulations

Stochastic simulations of the network were performed using the Stochastic Simulator Compiler (SSC) v0.6 [85]. The output from SSC was reformatted with custom Perl scripts and then plotted in MATLAB. SSC handles concentrations in units of molecules, so all parameter values were scaled appropriately, but the output values were converted to units of molarity in the figures given in this paper for ease of comparison. Reported values for protein concentrations are the totals of all forms of the protein: monomer, dimer, and bound to promoter. We represented cells with cubes of edge length $1 \mu\text{m}$. For single cell simulations, the cell was located at the center of a volume of $100 \times 1 \times 1 \mu\text{m}$. All multi-cell simulations consisted of a line containing 100 directly adjacent cells.

2.9.4 Discrete Cosine Transforms

A discrete cosine transform (DCT) expresses a finite sequence of data as a sum of cosine functions of different frequencies [103]. The eigenfunctions of the Laplacian operator on a one-dimensional spatial domain with zero-flux boundary conditions are cosine functions [55], which are represented more accurately by the DCT than by the discrete Fourier transform, which is appropriate for periodic boundary conditions. The DCT is useful for our analysis because it allows us to examine the presence of certain spatial wave numbers in a line of cells simulation relative to the other wave numbers and how these relations change over time. Because the amplitudes of a DCT are changing in time and can be both positive and negative, we take the average of the absolute values of spatial DCTs over an interval of time. This was handled in MATLAB using the function `dct`. Because concentrations are non-negative, there is always a significant offset component $k = 0$, which we omit from our figures for better scaling of the remaining wave numbers.

2.9.5 Construction of Plasmids

Plasmid construction was done via circular polymerase extension cloning (CPEC) [111] and 'Round-the-horn site-directed mutagenesis [58]. CPEC designs were started by using the j5 DNA assembly design automation software [60] to generate an initial set of oligonucleotides, which were then checked manually and tweaked based on the online Thermo Fisher Scientific Tm Calculator. All polymerase chain reactions (PCRs) were performed using the Phusion Hot Start II High-Fidelity DNA Polymerase (Thermo Scientific F-549).

To build a ZFP inverter (Plasmid 1 in Figure 2.37), we started initially with pWH29-77, which already had AraC on P_C and ZFP16-59 on P_{BAD} , and added the reporter operon from pWH32-32, which had mCherry on the constitutive promoter RPL-83 and the ZFP binding site op-3. During this CPEC construction step, we simultaneously added sLS, taken from pWH18-29, to the 5'-UTR of ZFP16-59 and changed the terminator on mCherry, since the same double terminator was used on pWH29-77 and pWH32-32. These were all placed on a p15a/KanR backbone taken from the BglBrick vector pBbA8k-RFP [83]. This initial inverter (pJH4-21) was for ZFP16-59 and sLS and was the basis for all of the other inverters. In the interest of time and space, the Genbank file for pJH4-21 is provided in the Appendix. It is a small enough plasmid that synthesis or construction from the other plasmids given here is feasible.

The other ZFP inverters were constructed using combinations of the following cloning steps, the oligonucleotides for which are given in Table 2.9:

- (1) Swap ZFP16-59 for ZFP16-57 or ZFP16-56 and swap op-3 on RPL-83 to op-18 or op-30 (o4-63 to o4-69).
- (2) Change RPL-83 to RPL-69 (o9-24, o9-25 - must happen after Step 1 to op-18).
- (3) Swap sLS for s04 or s05 (o6-23, o6-24, o6-37).
- (4) Perform RBS library search on either sLS (o3-42, o3-43, o9-23) or s04 (o2-14, o7-45, o7-74).

There was very little construction needed for the sRNA plasmid (Plasmid 2 in Figure 2.37) as William Holtz had previously constructed plasmids that had aLS constitutively produced by the entire Anderson library of synthetic promoters [114] (20 promoters: pWH39-21 through pWH39-40). These constructs were based on the ColE1/CmR backbone taken from a standard BglBrick vector such as pBbE2c-RFP [83]. Because the difference between a04 and a05 is so small, cloning for Plasmid 2 was carried out by swapping sLS for s04 and then doing mutagenesis to change s04 into s05.

Final experimental plasmids were obtained by co-transforming a Plasmid 1 variant with either an empty ColE1/CmR plasmid or the corresponding Plasmid 2 variant into DP10 cells.

Table 2.8: Plasmids used in ZFP-sRNA inverters, with intermediate cloning plasmids first, followed by the final experimental plasmids. All plasmids were transformed into the strain DP10 [77].

Name	Resistance	Origin	Description
pWH29-77	CmR	SC101**	AraC on P_C and ZFP16-59 on P_{BAD} .
pWH32-32	KanR	ColE1	mCherry on RPL-83 with op-3.
pWH18-29	CmR	p15a	GFP on pJ23119 with sLS. Used to get sLS.
pJH4-21	KanR	p15a	Operons from pWH29-77 and pWH32-32 moved onto KanR/p15a vector with sLS added to ZFP16-59 and mCherry terminator changed. The basis for Plasmid 1 (Figure 2.37).
pJH5-57	KanR	p15a	pJH4-21 with sLS changed to s04.
pWH16-57	AmpR	ColE1	Plasmid from William Holtz with ZFP16-57.
pWH16-56	AmpR	ColE1	Plasmid from William Holtz with ZFP16-56.
pWH39-21 pWH39-40	CmR	ColE1	Twenty plasmids with aLS on numerically-ordered promoters J23100 to J23119. The bases for Plasmid 2 (Figure 2.37).
VKM40	AmpR	ColE1	Plasmid from Vivek Mutalik with a04 on $P_{LlacO-1}$ and LacI. Used to get a04; a05 made from a04 via cloning.
pJH1-81	KanR	p15a	Empty plasmid.
pWH17-39	CmR	ColE1	Empty plasmid.
pJH4-37	KanR	p15a	AraC on P_C and mRFP1 on P_{BAD} for input measurement. Originally pBbA8k-RFP [83].
pJH7-17	KanR	p15a	Plasmid 1 with ZFP16-59, op-3, RPL-83, s04.
pJH7-59	KanR	p15a	Plasmid 1 with ZFP16-59, op-3, RPL-83, s05.
pJH4-40	KanR	p15a	Plasmid 1 with ZFP16-59, op-3, RPL-83, sLS.
pJH9-55	KanR	p15a	Plasmid 1 with ZFP16-57, op-18, RPL-69, s04.
pJH9-56	KanR	p15a	Plasmid 1 with ZFP16-57, op-18, RPL-69, s05.
pJH9-52	KanR	p15a	Plasmid 1 with ZFP16-57, op-18, RPL-69, sLS.
pJH7-57	KanR	p15a	Plasmid 1 with ZFP16-56, op-30, RPL-83, s04.
pJH7-60	KanR	p15a	Plasmid 1 with ZFP16-56, op-30, RPL-83, s05.
pJH4-64	KanR	p15a	Plasmid 1 with ZFP16-56, op-30, RPL-83, sLS.
pJH5-56	CmR	ColE1	Plasmid 2 with J23108 and a04.
pJH6-26	CmR	ColE1	Plasmid 2 with J23118 and a05.
pWH39-27	CmR	ColE1	Plasmid 2 with J23106 and aLS.
pJH6-63	KanR	p15a	Plasmid 3 with ZFP16-56, op-30, RPL-83, s04 targeting ZFP16-59, op-3, RPL-83, s05.
pJH7-42	CmR	ColE1	Plasmid 4 with a04 on J23108 and a05 on J23118.
pJH8-51	KanR	p15a	Plasmid 1* with ZFP16-59, op-3, RPL-83, s04/s05.
pJH7-61	KanR	p15a	Plasmid 3* with ZFP16-56, op-30, RPL-83, s04 targeting mCherry, s05.

Table 2.9: Oligonucleotides used in the construction of the ZFP-sRNA inverter plasmids. Here “for” denotes a forward primer, “rev” denotes a reverse primer, and “RTH” indicates use in ‘Round-the-horn site-directed mutagenesis.

Name	Description	Sequence
o4-63	ZFP swap pJH4-40vec for (to op-18)	cctatcagctgcgtgctttctat tgaggagatagtgaggag ttgacaattaatcatcgg
o4-69	ZFP swap pJH4-40vec for (to op-30)	cctatcagctgcgtgctttctat tagtggaaaggaatgggag ttgacaattaatcatcgg
o4-64	ZFP swap pJH4-40vec rev (RBS AAAGGA)	cctggttcagcatagatcctatcctttagatc
o4-65	ZFP pWH16-56/57 for	tagccggttgtaaggatctaaaggataggatctatgctggaaccaggatc
o4-66	ZFP pWH16-56/57 rev	gcctggagatccttactcgagtttgatccttattaagagggttttagatc
o4-67	ZFP swap pJH4-40 term spacer for	ggatccaaactcgagtaaggatctccaggc
o4-68	ZFP swap pJH4-40 term spacer rev	atagaaagcacgcagctgataggggtcga
o7-45	RTH s04/s05 RBS for	. nn agacaacaagatgtgcgaactcgatgctggaac
o2-14	RTH s04 RBS rev	ttattgatttttggcatggagaaacagtagag
o7-74	RTH s05 RBS rev	ttattgattttacgcatggagaaacagtagag
o3-42	vec sLS RBS lib rev	agatccttacaaccggctattagagtagc
o3-43	vec sLS RBS lib for	aggatctatgctggaaccaggatc
o9-23	sLS RBS lib	gatcctggttcagcatagatcct annyyyy agatccttacaaccggctattagagtagc
o7-24	RTH op rev	atagaaagcacgcagctgataggggtcga
o7-25	RTH op-3 for (RPL-83)	. gagagggaaggagagg ttgacaattaatcatcggctcataacc
o7-81	RTH op-18 for (RPL-83)	. tgaggagatagtgaggag ttgacaattaatcatcggctcataacc
o7-80	RTH op-30 for (RPL-83)	. tagtggaaaggaatgggag ttgacaattaatcatcggctcataacc
o9-24	RTH RPL-83 to 69 for	tgtggaacaattcattaaagaggagaaaggtac
o9-25	RTH RPL-83 to 69 rev (op-18)	taccctacgagccgatgattaattgtcaactctcc
o6-19	vec pWH39-29 for	ggatcctaactcgagtaaggatctccaggca
o6-20	vec pWH39-29 rev (pJ23108)	gctagcattatactaggactgagctagctgtcaga
o6-21	a04 from VKM40 for	agctcagtcctaggtataatgctagctcgacatcttgtgtctgatta
o6-22	a04 from VKM40 rev	gagatccttactcgagttaggatcctgatgaatccctaattgatttg
o6-34	RTH a04/a05 rev	aaatcaataatcagacaacaagatgtgcga
o6-35	RTH a05 for	. tacg cgaaaccatttgatcatatgacaagatgtg
o1-6	RTH to pJ23118 for	cggtagctcagtcctaggtattgtgctagctcgacatcttgtgtctg
o6-49	RTH to pJ23118 rev	tcaaacgtgccagatctttagaattcgatatctg
o6-23	RTH to s04/s05 for	caacaagatgtgcgaactcgatgctggaaccaggatc
o6-24	RTH to s04 rev (RBS TC)	tctgattattgatttt tg catggagaaacagtagagagttgc
o6-37	RTH to s05 rev (RBS TC)	tctgattattgatttt ac gcatggagaaacagtagagagttgc

2.9.6 Experimental Conditions and Procedure

Experiments were run in liquid media using EZ Rich defined medium (Teknova M2105) in 96-well deep well plates (DWPs) with 1.7 mL round wells using an AeraSeal breathable sealing film (Sigma Aldrich A9224) to cover. All liquid culture growth was performed in an INFORS HT Multitron Standard shaker with 25 mm throw at 37°C and 900 rpm. Antibiotic concentrations of 25 $\mu\text{g}/\text{mL}$ for Chloramphenicol (Cm) and 50 $\mu\text{g}/\text{mL}$ for Kanamycin (Kan) were used.

Each bacterial strain was streaked out from glycerol freezer stock onto double antibiotic LB CmKan plates and colonies were grown overnight in a 37°C warm room. Six colonies of each strain were transferred to 400 μL EZ Rich + CmKan wells of a DWP. These cell growth plates were placed on the shaker to grow for about 8 hours.

During growth, experimental DWPs were filled with 392 μL EZ Rich with different induction levels in each of the 12 columns. The chosen induction levels were: 40 mM, 10 mM, 5 mM, 1 mM, 500 μM , 200 μM , 100 μM , 50 μM , 20 μM , 10 μM , 5 μM , and no arabinose. This setup was chosen so that the data for each replicate (all induction levels) came from the same plate and plate-to-plate variation would get averaged out across the replicates. Previous data taken with a single induction level (and all replicates) on each plate produced data with plate-to-plate variability showing up in the induction levels (data not shown).

After removal from the shaker, each column of the growth plates were subcultured at 50X (8 μL) into every column of one of the experimental plates. The experimental plates were then placed on the shaker and grown for about 13 hours.

Upon removal from the shaker, the experimental plates were sampled using a Beckman Coulter Biomek FX laboratory automation workstation and 150 μL from each well was transferred into 96-well black plates with clear flat bottoms (Corning #3631). Bulk fluorescence measurements were taken using a Molecular Devices SpectraMax M2 microplate reader. Settings used were OD measured at 600 nm and RFP measured with 565/620 excitation and emissions wavelengths taken with 30 reads and medium sensitivity flash mode.

In each experimental plate, three sets of controls were used: a positive induction plasmid that has RFP on P_{BAD} used to measure the input induction level, an “empty” plasmid without RFP, and a “blank” row with just media (no cells). When analyzing the data, background OD levels were calculated by averaging the OD measurement of the “blank” wells for each plate and then subtracted from the OD measurements of the other wells. Next the background fluorescence was calculated by dividing the fluorescence measurements of the “empty” wells by their adjusted OD measurements and averaging for each plate. Final fluorescence values for the wells of interest were calculated by dividing their fluorescence measurements by their adjusted OD measurements and then subtracting off the background fluorescence. Each experiment was run with 6 replicates, average and standard error values were calculated and plotted against the positive induction data in Microsoft Excel.

Chapter 3

Lateral Inhibition

3.1 Lateral Inhibition and Contact-Dependent Inhibition

A number of recent efforts have focused on engineering synthetic multi-cellular behavior within ensembles of bacterial cells. To date, these efforts have largely focused on the use of diffusible signals (e.g. N-acyl homoserine lactones) to trigger transcription factors (and thus, up- or down-regulate gene expression). Cell signaling networks based on diffusible molecules tend to act only over long distances (i.e. large ensembles of cells), degrade slowly (half-life > 2 hours) and severely restrict the type of cell-cell cooperation possible. It is not surprising that while some early events in canonical developmental biology models employ diffusible (or diffusion-like) signals (e.g. bicoid, Nanos), most of the patterning that leads to segmentation and fate-specification appears to employ contact-mediated signals (either between cells or cells and extracellular matrix). Unfortunately, no synthetic contact-mediated system for affecting gene expression exists.

Lateral inhibition is a mechanism where cell-to-cell signaling induces neighboring cells to compete and diverge into sharply contrasting fates, enabling developmental processes such as segmentation or boundary formation [94]. The best-known example of lateral inhibition is the Notch pathway in Metazoans where membrane-bound Delta ligands bind to the Notch receptors on the neighboring cells. This binding releases the Notch intracellular domain in the neighbors, which then inhibits their Delta ligand production [28, 99, 124, 9]. Recent discoveries have shown that lateral inhibition is not limited to complex organisms: a contact-dependent inhibition (CDI) system has been identified in *E. coli* where delivery via membrane-bound proteins of the C-terminus of the gene *cdiA* causes down regulation of metabolism [7, 6, 134]. Despite the vigorous research on elucidating natural pathways such as Notch and CDI, a synthetic lateral inhibition system for pattern formation has not been developed.

The general diagram of lateral inhibition is shown in Figure 3.1.

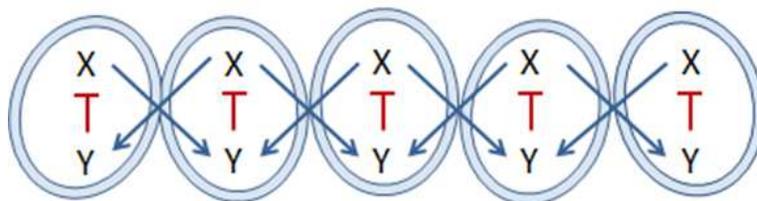


Figure 3.1: A two-component network in which the production of molecule Y is activated by high concentrations of X in neighboring cells and the production of X is inhibited by Y internally in each cell. This diagram was drawn to imply the use of a contact-mediated mechanism in a line of cells, but lateral inhibition is not restricted to just this type of cell-to-cell communication nor this particular geometry.

3.1.1 CDI as a Transport System

The CDI system in *E. coli* is particularly exciting because it presents a currently untapped method of synthetic cell-to-cell communication and in a less complex organism. In the natural circuit, an inhibitor cell contains a three-gene operon containing *cdiBAI*. CdiB shares close homology to two-partner secretion systems and transports CdiA protein out of the cell [8]. CdiA becomes cleaved in three different locations, and the C-terminus is transported into the neighboring cell. The inhibitor cell also expresses *cdiI*, which binds to the C-terminus and prevents inhibition. Figure 3.2 shows a schematic of the inhibition process. If the system could be modified to deliver a chosen payload instead of the *cdiA* C-terminus, it would open up a whole new set of possibilities for designing complex synthetic cellular behaviors.

There is evidence to suggest that this CDI system might work as a transport system for modified or swapped out C-terminus. It has been shown that homologous systems exist in other bacteria and the C-terminus of the *cdiA* genes show variability and inhibit growth in different ways [6]. The authors swapped the C-terminus of the *cdiA* genes and showed that they could prevent growth inhibition only if they expressed the cognate *cdiI* from the same organism. The evidence suggests that the homologous CDI systems utilize the same mechanism for exiting the inhibitor cell since the CdiB protein was not changed and most of the *cdiA* gene is fairly well-conserved. It is noted that CDI sensitivity is dependent upon having the cognate *bamA* species in the target cells [118]. Figure 3.3 shows where the Low group swapped the downstream C-terminus of the *cdiA* genes; the VENN motif and upstream sequence are fairly well-conserved.

A graduate student from the Arkin lab, David Chen, attempted to modify the CDI system to achieve a measurable target cell response to a new CdiA payload. Initial attempts focused on fusing or linking a small passenger at different locations in the CdiA C-terminus. Later attempts focused on using tRNase co-localization (“scaffolding”) of CdiI and CysK to deliver our transcription factor via TEV cleavage in a linker with CdiI [26]. Unfortunately all of these attempts failed to produce a measurable response in receiver cells. The hypothesis we were left with is that the robust metabolic down regulation response is initiated with

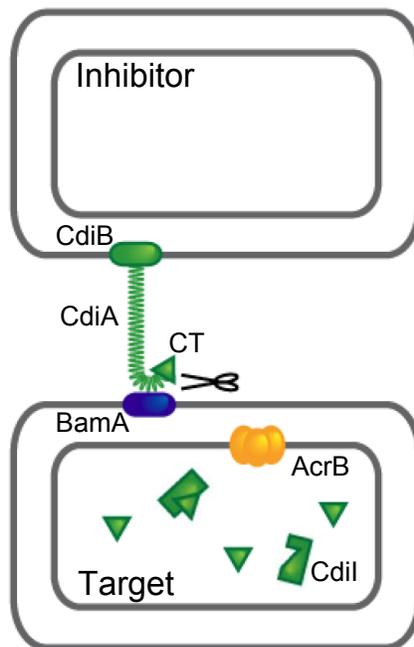


Figure 3.2: Schematic of the CdiABI system, adapted from [6] Supplementary Figure 1. CdiB transports CdiA to the outside of the cell. When the inhibitor cell comes in contact with a neighboring cell, CdiA is cleaved and the C-terminus (CT) is transported inside the target cell via the BamA and AcrB proteins. The C-terminus then causes growth inhibition via an unknown downstream pathway. If CdiI is expressed, it binds and inactivates the C-terminus, preventing or reversing the metabolic down regulation [7].

only a very small number of native CdiA C-termini.

This certainly remains a ripe area for research, as the creation of a contact-based transcription factor delivery system would be a huge contribution to synthetic biology. However, we were left with lots of ideas for patterning based on CDI, but no experimental system with which to test them.

3.2 Lateral Inhibition System Analysis

The theoretical work for this project was done in collaboration with Murat Arcaç and Ana S. Rufino Ferreira [9, 116]. Only the main results will be reproduced here. See the cited papers for proofs of the theorems and lemmas.

We analyze lateral inhibition networks by treating them as interconnections of input-output models for each cell, where the inputs represent the influence of adjacent cells and the outputs correspond to the concentrations of the species that interact with adjacent cells. We represent these interconnections using an undirected graph where the cells are the vertices

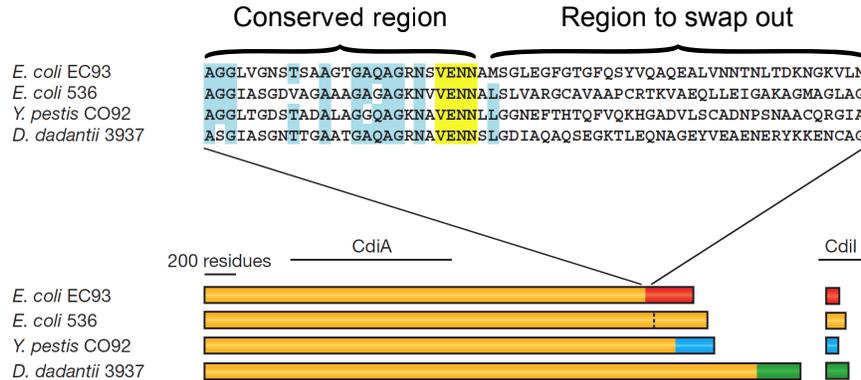


Figure 3.3: Similarities between homologous CDI systems, reproduced from [6] Figure 1b. The authors of that paper were able to swap C-termini and show continued metabolic downregulation, though CdiI-mediated immunity was still restricted to the cognate CdiI. Because these systems appear to use the same transport mechanism, it may be possible to modify CdiA after the conserved VENN peptide motif to deliver a specified payload of our choosing.

and available communication channels are the edges, weighted by communication strength. In the example of Notch or CDI, the presence of an edge between two vertices means that the corresponding cells are in contact and the edge weight will depend on how many Notch receptors are bound or CdiA C-termini are transported into the cytoplasm.

We implicitly assume that all of the cells are identical, and thus there exists a homogeneous steady state. With a few assumptions on the input-output map, we can use algebraic properties of the graph and tools from monotone systems theory [121] to prove existence and stability (or instability) of the homogeneous steady state or other non-homogeneous steady states (patterns) based on graph partitioning. From these results we can predict whether an experimental system will exhibit patterning and within what parameter ranges [9, 116].

3.2.1 Lateral Inhibition Model

We represent the cell network by an undirected and connected graph $\mathcal{G} = \mathcal{G}(V, E)$, where the set of vertices V represents a group of cells and each edge $e \in E$ represents a contact between two cells. The connectivity between cells i and j is represented by the non-negative constant $w_{i,j} = w_{j,i} \in \mathbb{R}_{\geq 0}$. We let $w_{i,j} = 0$ when i and j are not in contact, and $w_{i,j} > 0$ when they are in contact.

Let N be the number of cells and define the *scaled adjacency matrix* $P \in \mathbb{R}^{N \times N}$ of \mathcal{G} as

$$p_{ij} = d_i^{-1} w_{i,j}, \quad (3.1)$$

where the scaling factor is the vertex degree $d_i = \sum_j w_{i,j}$. The structure of P is identical to the transposed probability transition matrix of a reversible Markov chain, therefore P

is non-negative and row-stochastic (i.e. $P\mathbf{1}_N = \mathbf{1}_N$, where $\mathbf{1}_N \in \mathbb{R}^N$ denotes the vector of ones) with real-valued eigenvalues and eigenvectors.

The identical cells are numbered $i = 1, \dots, N$ and are each described by the dynamical model:

$$\begin{cases} \dot{x}_i = f(x_i, u_i), \\ y_i = h(x_i), \end{cases} \quad (3.2)$$

where $x_i \in \mathcal{X} \subset \mathbb{R}^n$ describes the state of the n species concentrations in cell i , $u_i \in \mathcal{U} \subset \mathbb{R}$ is an aggregate input from neighboring cells, and $y_i \in \mathcal{Y} \subset \mathbb{R}$ represents the output of each cell that contributes to the input to its neighbors.

The cell-to-cell interaction is then described by:

$$u = Py, \quad (3.3)$$

where P is the scaled adjacency matrix of the contact graph as in (3.1), $u \triangleq [u_1, \dots, u_N]^T$, and $y \triangleq [y_1, \dots, y_N]^T$. This means that the input to each cell is a weighted average of the outputs for neighboring cells.

Standing assumptions. We assume that $f(\cdot, \cdot)$ and $h(\cdot)$ are continuously differentiable, and that for each constant input $u^* \in \mathcal{U}$, the system (3.2) has a globally asymptotically stable steady state

$$x^* \triangleq S(u^*), \quad (3.4)$$

which is also a hyperbolic equilibrium (i.e. $\frac{\partial f}{\partial x}|_{(x^*, u^*)}$ has no eigenvalues on the imaginary axis). Furthermore, we assume that the map $S : \mathcal{U} \rightarrow \mathcal{X}$ and the map $T : \mathcal{U} \rightarrow \mathcal{Y}$, defined as

$$T(\cdot) \triangleq h(S(\cdot)), \quad (3.5)$$

are continuously differentiable, and that $T(\cdot)$ is a positive, bounded, and decreasing function. The decreasing property of $T(\cdot)$ is consistent with the lateral inhibition feature, since higher outputs in one cell lead to lower values in neighboring cells.

Note that the steady states of the system (3.2)-(3.3) are given by $x_i = S(u_i)$, in which u_1, \dots, u_N are solutions of the equation

$$u = P\mathbf{T}_N(u), \quad (3.6)$$

where $\mathbf{T}_N(u) = [T(u_1), \dots, T(u_N)]^T$. Because P is row-stochastic, (3.6) admits a solution that is homogeneous across all cells, that is $u_i = u^* \forall i = 1, \dots, N$, where u^* is the unique fixed point of $T(\cdot)$ (i.e. $T(u^*) = u^*$). We refer to the corresponding steady state $x_i^* = S(u^*)$ for all i as the homogeneous steady state of the interconnection.

3.2.2 Lateral Inhibition Patterns

The main theoretical results will only be paraphrased here in a way that is most relevant to this project. Full descriptions and derivations can be found in [9, 116] and these concepts will be covered in further detail in Section 3.3.2.

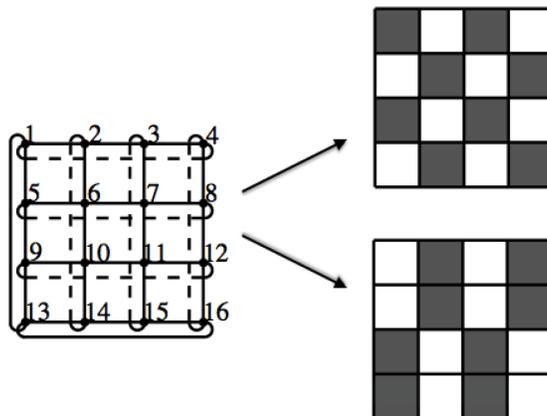


Figure 3.4: Example of steady state patterns on a 2-D mesh. Left: Contact graph structure with vertices numbered and represented as dots with edges represented as lines connecting the dots. Right: Two different bipartite equitable partitions are shown. Here the steady state of each vertex is represented as either a white or grey colored square. The existence criterion from [116] is an algebraic condition based on the graph structure and the dynamical model. Note that both steady states may co-exist in certain conditions.

Because we are expecting “on-off” patterns, we examine bipartite equitable graph partitions as potential steady states of the system, where one orbit contains all the vertices in the “on” state and the other orbit contains all the vertices in the “off” state. We can determine the existence of these steady state patterns using an algebraic condition that is a function of both the graph structure and the dynamical model [116]. See Figure 3.4 for an example on a 2-D mesh where two bipartite equitable partitions are shown. Under certain conditions multiple non-homogeneous steady state patterns may exist. We can further provide a small-gain-type sufficient condition [35] to show asymptotic stability of these non-homogeneous steady states [116]. Based on this theoretical framework, it should be possible to experimentally demonstrate multiple types of patterns with the same graph structure by changing either dynamical parameters or initial conditions.

3.3 Compartmental Lateral Inhibition System

Given that we were not able to achieve a synthetic CDI system, we set about trying to design a lateral inhibition system with similar properties to CDI but not based on contact. We will use diffusible molecules for cell-to-cell communication due to their availability and our familiarity with them. The two main properties that distinguish contact-based communication from diffusion are:

- (1) The prevention of auto-inhibition due to the outward-facing nature of the membrane-bound effectors.

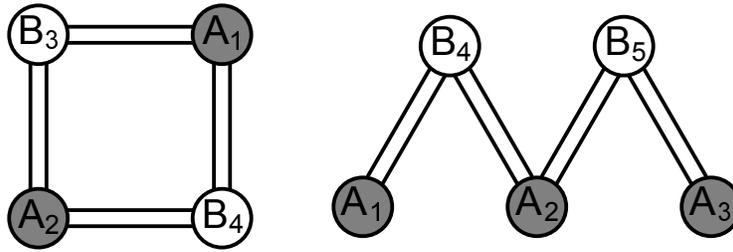


Figure 3.5: Example of network geometries for compartmental lateral inhibition system. There are two cell types A and B contained in the compartments (circles) that produce different AHLs and sense the other AHL, preventing auto-inhibition. The AHLs diffuse along engineered communication channels to neighboring compartments, mimicking cells in contact.

- (2) A range of effect limited to just neighbors in contact.

In order to mimic these behaviors with diffusion, we design our system as follows:

- (1) We use two orthogonal AHLs and create two distinct cell types, each producing one AHL and sensing the other.
- (2) We run experiments in devices that create geometries of our choosing with communication channels between compartments that house colonies of one of the cell types (Figure 3.5). In practice, our devices will be machined geometries filled with agar to allow diffusion of AHL between growing colonies. The agar will fix the center of the colonies in space.

We will show that this system is able to spontaneously generate contrasting patterns between neighboring compartments, much like the patterns predicted in CDI networks.

Our experimental compartmental lateral inhibition system is shown in Figure 3.6. The two orthogonal diffusible signals chosen are 3OC6HSL, produced by LuxI from *V. fischeri*, and 3OC12HSL, produced by LasI from *P. aeruginosa*. These signals are paired with a constitutive amount of the non-cognate receptor proteins LasR and LuxR, respectively, so that each cell type cannot detect the AHL that it produces. Sensing of the orthogonal AHL triggers the promoter P_{LuxI} , which produces a red fluorescent protein (RFP) and TetR, which represses the further production of autoinducer synthase from the promoter $P_{LtetO-1}$. For further discussion of the parts selection, see Section 3.4.2.

We model the dynamics of cell type A with the following set of equations:

$$\begin{aligned} \frac{\partial}{\partial t} m_T &= V_{P_{LuxI}} N_{P_{LuxI}} C \left(\frac{1}{1 + (K_{R-Y}/p_{R-Y})^{n_{R-Y}}} + \ell_{P_{LuxI}} \right) - \gamma_{mT} m_T \\ \frac{\partial}{\partial t} p_T &= \epsilon_T m_T - \gamma_T p_T \end{aligned}$$

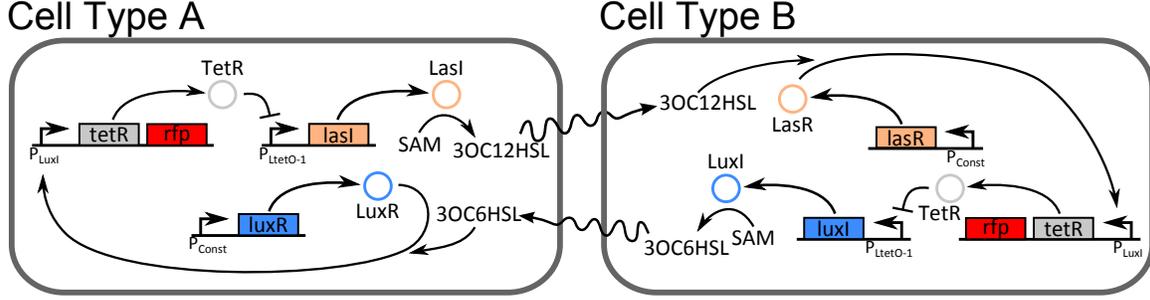


Figure 3.6: Diagram of the components in the compartmental lateral inhibition system. Two distinct cell types and orthogonal diffusible communication channels are used to prevent auto-inhibition. The quorum sensing components used are LuxI/LuxR from *V. fischeri* and LasI/LasR from *P. aeruginosa*. This diagram shows a single cell of type *A* communicating with a single cell of type *B*. In practice, we will have colonies of each cell type separated spatially in a variety of network geometries.

$$\begin{aligned}
 \frac{\partial}{\partial t} m_{I,X} &= V_{P_{LtetO-1}} N_{P_{LtetO-1}} C \left(\frac{1}{1 + (p_T/K_T)^{n_T}} + \ell_{P_{LtetO-1}} \right) - \gamma_{m_{I,X}} m_{I,X} \\
 \frac{\partial}{\partial t} p_{I,X} &= \epsilon_{I,X} m_{I,X} - \gamma_{I,X} p_{I,X} \\
 \frac{\partial}{\partial t} X &= \nu_X p_{I,X} - \gamma_X X + d_X \nabla^2 X \\
 \frac{\partial}{\partial t} Y &= -k_{f,Y} Y (p_{R,Y} - p_{R-Y}) + k_{r,Y} p_{R-Y} - \gamma_Y Y + d_Y \nabla^2 Y \\
 \frac{\partial}{\partial t} p_{R-Y} &= k_{f,Y} Y (p_{R,Y} - p_{R-Y}) - k_{r,Y} p_{R-Y}
 \end{aligned} \tag{3.7}$$

where m_i are mRNA concentrations, p_i are protein concentrations, V_i are velocity constants, N_i are copy numbers, K_i are dissociation constants, n_i are Hill coefficients, ℓ_i are leakage rates normalized to V_i , γ_i are degradation rates, and ϵ_i are protein translational rates. The velocity and leakage constants and copy numbers are subscripted by promoter. All other parameters are subscripted according to their corresponding species. X is the diffusible molecule 3OC12HSL and Y is the diffusible molecule 3OC6HSL. We make note in the subscripts that the autoinducer synthase and receptor proteins are different for species X and Y by using a comma notation (e.g. $p_{I,X}$ is the concentration of the autoinducer synthase for X). $p_{R,Y}$ is the total amount of receptor protein in the cell, which is assumed constant. ν is the generation rate of AHL from autoinducer synthase and k_f and k_r are the forward and reverse binding rates, respectively, of AHL to its cognate receptor protein. C is the concentration level generated by a single molecule in an *E. coli* cell and d_X and d_Y are the diffusion coefficients of the AHLs. The dynamics for cell type *B* are similar and can be obtained by swapping the X and Y differential equations and then swapping all of the X and Y indices.

3.3.1 Compartmental Lateral Inhibition Analysis

Our model of lateral inhibition needs to be tweaked to accommodate our compartmental system. In particular, we must now account for having two cells types instead of identical cells and compartments filled with colonies instead of individual cells.

Our network has N_A compartments of type A and N_B compartments of type B that communicate through diffusible molecules. Cells of type A produce the diffusible species X (3OC12HSL), which is bound by the receptor species found only in cells of type B . Similarly, cells of type B produce the diffusible species Y (3OC6HSL), which is only detected by cells of type A . Due to the nature of the system, we separate the dynamics into three modules:

- (1) **Transmitter** module where species X (or Y) is produced and released,
- (2) **Receiver** module where Y (or X) is detected (i.e. bound by its cognate receptor species to form a receiver complex), and
- (3) **Inhibitory** module which inhibits transmitter activity in the presence of the receiver complex.

To facilitate the analysis, we combine the transmitter modules of A , the diffusion of X , and the receiver modules of B into a new “**transceiver**” block for X . Similarly, the transceiver block for Y is composed of the transmitter modules of B , the diffusion of Y , and the receiver modules of A . The cell network is represented in Figure 3.7A. Each compartment is represented with a sub-block labeled H_A or H_B , corresponding to the inhibitory circuit of types A and B , respectively. The concentration of the autoinducer synthase for the production of X (respectively, Y) is denoted y_A (y_B), and R_A (R_B) is the concentration of the receiver complex, resulting from the binding of Y (X) to its cognate receptor protein.

The transceiver blocks incorporate diffusion in an ordinary differential equation compartmental model that describes the concentrations of the diffusible species in each compartment. We modify our undirected graph $\mathcal{G}(V, E)$ so that each vertex is a compartment and each edge represents a channel between compartments. Each edge $(i, j) \in E$ has an edge weight $d_{ij} = d_{ji}$ ($d_{ij} = 0$ if compartments i and j are not connected), which is proportional to the diffusivity of the species and inversely proportional to the square of the distance between compartments i and j . We define the weighted Laplacian of the graph to be:

$$\{L\}_{ij} = \begin{cases} -\sum_{j=1}^N d_{ij} & \text{if } i = j \\ d_{ij} & \text{if } i \neq j \end{cases} \quad (3.8)$$

The dynamical model of the transceiver for X (tx/rx $B \leftarrow A$ in Figure 3.7A) is represented by:

$$\text{tx/rx}_{A \rightarrow B} : \begin{cases} \begin{bmatrix} \dot{X}_A \\ \dot{X}_B \end{bmatrix} & = \begin{bmatrix} \Gamma_X(X_A, y_A) \\ \Phi_X(X_B, R_B) \end{bmatrix} + L \begin{bmatrix} X_A \\ X_B \end{bmatrix} \\ \dot{R}_B & = \Psi_X(X_B, R_B), \end{cases} \quad (3.9)$$

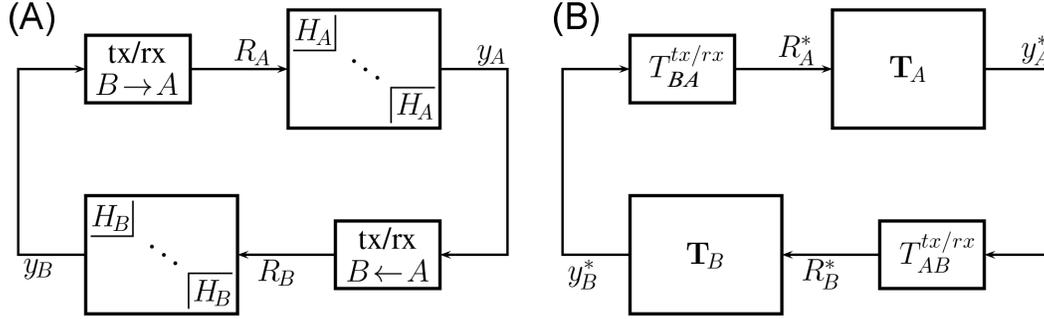


Figure 3.7: Network diagram of compartmental lateral inhibition system model with compartments of type A and B communicating via diffusion. (A) For each type of diffusible species, the transceiver (tx/rx) block includes the dynamics of the transmitter modules of the sender compartments, the diffusion process, and the receiver modules on the detecting compartments. The compartments (H_A and H_B) contain the inhibition modules. (B) The same network but with the associated steady-state maps overlaid on the blocks.

where $X_A \in \mathbb{R}_{\geq 0}^{N_A}$ represents the concentration of species X in compartments A due to production, $X_B \in \mathbb{R}_{\geq 0}^{N_B}$ represents the concentration of species X in compartments B due to diffusion, and $R_B \in \mathbb{R}_{\geq 0}^{N_B}$ represents the concentration of receiver complexes in compartments B . The functions $\Gamma_X(\cdot, \cdot) \in \mathbb{R}_{\geq 0}^{N_A}$, $\Phi_X(\cdot, \cdot) \in \mathbb{R}_{\geq 0}^{N_B}$, and $\Psi_X(\cdot, \cdot) \in \mathbb{R}_{\geq 0}^{N_B}$ are concatenations of the decoupled elements $\gamma_X^i(X_A^i, u^i) \in \mathbb{R}_{\geq 0}$, $\phi_X^j(X_B^j, R_B^j) \in \mathbb{R}_{\geq 0}$, and $\psi_X^j(X_B^j, R_B^j) \in \mathbb{R}_{\geq 0}$ for $i = 1, \dots, N_A$ and $j = 1, \dots, N_B$, and assumed to be continuously differentiable. $\gamma_X^i(\cdot, \cdot)$ models the production and degradation of X in compartment i of type A , $\phi_X^j(\cdot, \cdot)$ models the binding and unbinding of X to its cognate receptor protein and its degradation in compartment j of type B , and $\psi_X^j(\cdot, \cdot)$ models the binding and unbinding of the receiver complex in compartment j of type B . The transceiver $\text{tx/rx}_{B \rightarrow A}$ for Y is defined similarly by changing X to Y and switching indices A with B in (3.9).

Assumption 3.1. For each constant input $y_A^* \in \mathbb{R}_{\geq 0}^{N_A}$ (and $y_B^* \in \mathbb{R}_{\geq 0}^{N_B}$), the subsystem (3.9) has a globally asymptotically stable steady state (X_A^*, X_B^*, R_B^*) , which is a hyperbolic equilibrium (i.e. the Jacobian has no eigenvalues on the imaginary axis). Furthermore, there exist positive and increasing functions $T_{AB}^{\text{tx/rx}} : \mathbb{R}_{\geq 0}^{N_A} \rightarrow \mathbb{R}_{\geq 0}^{N_B}$ and $T_{BA}^{\text{tx/rx}} : \mathbb{R}_{\geq 0}^{N_B} \rightarrow \mathbb{R}_{\geq 0}^{N_A}$ such that

$$R_B^* \triangleq T_{AB}^{\text{tx/rx}}(y_A^*) \quad \text{and} \quad R_A^* \triangleq T_{BA}^{\text{tx/rx}}(y_B^*). \quad (3.10)$$

The increasing property of these maps is meaningful, since a higher autoinducer synthase input leads to more production, and thus, through diffusion, more detection on the receiver side.

Next, we represent the blocks H_k^i , $i = 1, \dots, N_k$ of type $k \in \{A, B\}$ with models of the form:

$$H_k^i = \begin{cases} \dot{x}_i &= f_k(x_i, u_i) \\ y_i &= h_k(x_i), \end{cases} \quad (3.11)$$

where $x_i \in \mathbb{R}_{\geq 0}^n$ is the vector of species combinations in compartment i , $u_i \in \mathbb{R}_{\geq 0}$ is the input of i (concentration of the receiver complex), and $y_i \in \mathbb{R}_{\geq 0}$ is the output of i (concentration of autoinducer synthase). We denote $x_k = [x_1^T, \dots, x_{N_k}^T]^T \in \mathbb{R}_{\geq 0}^{nN_k}$, $u_k = [u_1, \dots, u_{N_k}]^T \in \mathbb{R}_{\geq 0}^{N_k}$, and $y_k = [y_1, \dots, y_{N_k}]^T \in \mathbb{R}_{\geq 0}^{N_k}$ for $k \in \{A, B\}$.

Assumption 3.2. For $k \in \{A, B\}$, $f_k(\cdot, \cdot)$ and $h_k(\cdot)$ in the subsystem (3.11) are continuously differentiable.

Assumption 3.3. For $k \in \{A, B\}$ and each constant input $u^* \in \mathbb{R}_{\geq 0}$, the subsystem (3.11) has a globally asymptotically stable steady state $x^* \triangleq S_k(u^*)$, which is a hyperbolic equilibrium. Furthermore, the maps $S_k : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}^n$ and $T_k : \mathbb{R}_{\geq 0}^n \rightarrow \mathbb{R}_{\geq 0}$, defined as:

$$T_k(\cdot) \triangleq h_k(S_k(\cdot)), \quad (3.12)$$

are continuously differentiable, and $T_k(\cdot)$ is a positive, bounded, and decreasing function.

The decreasing property of $T_k(\cdot)$ is consistent with lateral inhibition, since higher input in one cell leads to lower output in its neighbors.

Given Assumptions 3.1-3.3, the full system given by (3.9) and (3.11) has a steady state if there exist variables $z_A \in \mathbb{R}_{\geq 0}^{N_A}$ and $z_B \in \mathbb{R}_{\geq 0}^{N_B}$ such that:

$$\begin{cases} z_A &= \mathbf{T}_A(T_{BA}^{\text{tx/rx}}(\mathbf{T}_B(T_{AB}^{\text{tx/rx}}(z_A)))) \\ z_B &= \mathbf{T}_B(T_{AB}^{\text{tx/rx}}(\mathbf{T}_A(T_{BA}^{\text{tx/rx}}(z_B)))) \end{cases} \quad (3.13)$$

with

$$\begin{aligned} \mathbf{T}_A(u_A) &= [T_A(u_A^1), \dots, T_A(u_A^{N_A})]^T : \mathbb{R}_{\geq 0}^{N_A} \rightarrow \mathbb{R}_{\geq 0}^{N_A}, \\ \mathbf{T}_B(u_B) &= [T_B(u_B^1), \dots, T_B(u_B^{N_B})]^T : \mathbb{R}_{\geq 0}^{N_B} \rightarrow \mathbb{R}_{\geq 0}^{N_B}. \end{aligned}$$

Figure 3.7B shows the steady-state maps overlaid on each of the corresponding network blocks.

3.3.2 Existence and Proof of Contrasting Patterns

We will now determine when z_A and z_B exhibit sharply contrasting values, indicating an “on/off” pattern. Here we use the notion of *equitable partition* from graph theory [50] to reduce the dimension of the maps in (3.13).

Definition 3.1. For a weighted and undirected graph $\mathcal{G}(V, E)$ with a Laplacian matrix L as defined in (3.8), a partition of the vertex set V into classes O_1, \dots, O_r is said to be equitable if there exists \bar{d}_{ij} for $i, j = 1, \dots, r$ such that

$$\sum_{v \in O_j} d_{uv} = \bar{d}_{ij}, \quad \forall u \in O_i, i \neq j. \quad (3.14)$$

This means that the sum of the edge weights from a vertex in a class O_i into all the vertices in a class O_j ($i \neq j$) is invariant of the choice of the vertex in class O_i . We let the *quotient Laplacian* $\bar{L} \in \mathbb{R}^{r \times r}$ be formed by the off-diagonal entries \bar{d}_{ij} , and $\{\bar{L}\}_{ii} = \{L\}_{ii} = -\sum_{j=1, j \neq i}^r \bar{d}_{ij}$.

Assumption 3.4. The partition of V into the classes O_A of type A and O_B of type B is equitable.

This implies that the total incoming edge weight of the species X (and Y) is the same for all the compartments of type B (A). For example, the network shown on the left in Figure 3.5 is equitable with respect to the classes O_A and O_B if $d_{13} + d_{14} = d_{23} + d_{24}$ and $d_{13} + d_{23} = d_{14} + d_{24}$, which implies $d_{13} = d_{24}$ and $d_{23} = d_{14}$. Since the edge weights d_{ij} are inversely proportional to the square of the distance, this means that opposite channels must have the same length, so any parallelogram geometry would be equitable.

Assumption 3.4 allows us to search for solutions to (3.13) where the compartments of the same type have the same steady-state value, that is:

$$z = [\bar{z}_A, \dots, \bar{z}_A, \bar{z}_B, \dots, \bar{z}_B]^T = [\bar{z}_A \mathbf{1}_{N_A}^T, \bar{z}_B \mathbf{1}_{N_B}^T]^T, \quad (3.15)$$

where $\bar{z}_A \in \mathbb{R}_{\geq 0}$ and $\bar{z}_B \in \mathbb{R}_{\geq 0}$. This means that the transceiver input-output maps become decoupled and $T_{AB}^{\text{tx/rx}}(\bar{z}_A \mathbf{1}_{N_A}) = T_{AB}(\bar{z}_A) \mathbf{1}_{N_B}$, with $T_{AB} : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$. A similar relation holds for $T_{BA}^{\text{tx/rx}}(\cdot)$ with $T_{BA} : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$. Furthermore, \bar{z}_A and \bar{z}_B satisfy the following reduced system of equations:

$$\begin{cases} \bar{z}_A = T_A(T_{BA}(T_B(T_{AB}(\bar{z}_A)))) \triangleq \bar{T}_A(\bar{z}_A) \\ \bar{z}_B = T_B(T_{AB}(T_A(T_{BA}(\bar{z}_B)))) \triangleq \bar{T}_B(\bar{z}_B), \end{cases} \quad (3.16)$$

where $\bar{T}_A : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ and $\bar{T}_B : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ are compositions of scalar maps. For any solution \tilde{z}_A to the top equation in (3.16), $\tilde{z}_B = T_B(T_{AB}(\tilde{z}_A))$ must be a solution to the bottom equation.

From Assumptions 3.1-3.3, the input-output maps $\bar{T}_A(\cdot)$ and $\bar{T}_B(\cdot)$ are positive, increasing, and bounded functions. Figure 3.8 illustrates typical shapes of $\bar{T}_A(\cdot)$ and $\bar{T}_B(\cdot)$. In Figure 3.8A there exists only one solution (orange circles), which is a *near-homogeneous* steady state. The discrepancy between \tilde{z}_A and \tilde{z}_B is due only to nonidentical $\bar{T}_A(\cdot)$ and $\bar{T}_B(\cdot)$. The map $\bar{T}_A(\cdot)$ in Figure 3.8B has three fixed points: a middle solution (near-homogeneous), a high fixed point (blue triangle), and a low fixed point (green square). The latter two have a

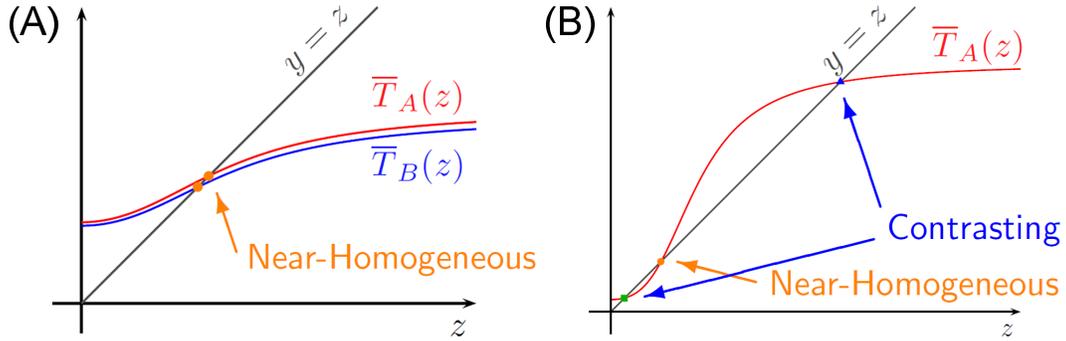


Figure 3.8: Typical shapes of input-output steady-state maps $\bar{T}_A(\cdot)$ and $\bar{T}_B(\cdot)$. (A) The unique pair of fixed points (orange circles) is near-homogeneous and no contrasting patterns emerge. (B) There exist three pairs of fixed points (orange circle, green square, and blue triangle) and the two additional solutions represent contrasting steady-state patterns. The curve for $\bar{T}_B(z)$ was omitted for clarity, but looks very similar to $\bar{T}_A(z)$.

corresponding opposite (low/high) fixed point in $\bar{T}_B(\cdot)$ and therefore represent contrasting steady-state patterns.

Note that a contrasting pattern emerges when the near-homogeneous steady state has a slope greater than 1, that is:

$$\left. \frac{d\bar{T}_A}{dz_A} \right|_{\tilde{z}_A} = T'_{AB}(\tilde{z}_A)T'_B(T_{AB}(\tilde{z}_A))T'_{BA}(\tilde{z}_B)T'_A(T_{BA}(\tilde{z}_B)) > 1. \quad (3.17)$$

Indeed, due to the boundedness and strictly increasing properties of the map $\bar{T}_A(\cdot)$, there must exist at least two other fixed point pairs of (3.16):

$$\begin{aligned} & (z_A^*, z_B^* \triangleq T_B(T_{AB}(z_A^*))) \\ & (z_A^{**}, z_B^{**} \triangleq T_B(T_{AB}(z_B^{**}))), \end{aligned}$$

for which $z_A^* > \tilde{z}_A$, $z_B^* < \tilde{z}_B$, $z_A^{**} < \tilde{z}_A$, and $z_B^{**} > \tilde{z}_B$.

To analyze convergence to the steady-state pattern in (3.16), we employ monotonicity assumptions.

Definition 3.2. A monotone system is one that preserves a partial ordering of the initial conditions as the solutions evolve over time.

A partial ordering is defined with respect to a *positivity cone* in the Euclidean space that is closed, convex, pointed ($K \cap -K = \{0\}$), and has nonempty interior. In such a cone, $x \preceq \hat{x}$ means $\hat{x} - x \in K$. Given the positivity cones K^U, K^Y, K^X for the input, output, and state space, the system $\dot{x} = f(x, u), y = h(x)$ is said to be *monotone* if $x(0) \preceq \hat{x}(0)$ and $u(t) \preceq \hat{u}(t) \forall t \geq 0$ imply that the resulting solutions satisfy $x(t) \preceq \hat{x}(t) \forall t \geq 0$, and the output map is such that $x \preceq \hat{x}$ implies $h(x) \preceq h(\hat{x})$ [5].

Assumption 3.5. *The system tx/rx_{A→B} in (3.9) is monotone with respect to $K^U = \mathbb{R}_{\geq 0}^{N_A}$, $K^Y = \mathbb{R}_{\geq 0}^{N_B}$, and $K^X = \mathbb{R}_{\geq 0}^{N_A+2N_B}$. Similarly, tx/rx_{B→A} is monotone with respect to $K^U = \mathbb{R}_{\geq 0}^{N_B}$, $K^Y = \mathbb{R}_{\geq 0}^{N_A}$, and $K^X = \mathbb{R}_{\geq 0}^{2N_A+N_B}$.*

Assumption 3.6. *The systems H_A and H_B in (3.11) are monotone with respect to $K^U = -K^Y = \mathbb{R}_{\geq 0}$ and $K^X = K$, where K is some positivity cone in \mathbb{R} .*

These monotonicity assumptions are consistent with Assumptions 3.1 and 3.3, as they imply the increasing property of the input-output maps $T_{BA}^{\text{tx/rx}}(\cdot)$ and $T_{AB}^{\text{tx/rx}}(\cdot)$ and the decreasing behavior of $T_A(\cdot)$ and $T_B(\cdot)$.

Theorem 3.1. *Consider the network (3.9) and (3.11) and suppose Assumptions 3.1, 3.3, 3.5, and 3.6 hold. Let the partition of the compartments into the classes O_A and O_B be equitable. The steady state described by (3.16) is asymptotically stable if:*

$$T'_{AB}(\tilde{z}_A)T'_B(T_{AB}(\tilde{z}_A))T'_{BA}(\tilde{z}_B)T'_A(T_{BA}(\tilde{z}_B)) < 1, \quad (3.18)$$

and unstable if (3.17) holds.

For the proof of Theorem 3.1, see [117].

3.3.3 Patterning Region

Now we revisit our actual compartmental lateral inhibition system (Figure 3.6) and fit it to our graph theory framework and use Theorem 3.1 to examine the parameter space for when contrasting patterns are stable. We re-organize the equations given in (3.7) and split them into the appropriate inhibition and transceiver blocks. This splitting of the model species is shown in Figure 3.9.

The inhibitory sub-blocks for the $i = 1, \dots, N_A$ cells of type A are given by:

$$H_A^i = \begin{cases} \frac{d}{dt}m_T^i &= V_{P_{LuxI}}N_{P_{LuxI}}C\left(\frac{1}{1+(K_{R-Y}/R_A^i)^{n_{R-Y}}} + \ell_{P_{LuxI}}\right) - \gamma_{mT}m_T^i \\ \frac{d}{dt}p_T^i &= \epsilon_T m_T^i - \gamma_T p_T^i \\ \frac{d}{dt}m_{I,X}^i &= V_{P_{LtetO-1}}N_{P_{LtetO-1}}C\left(\frac{1}{1+(p_T^i/K_T)^{n_T}} + \ell_{P_{LtetO-1}}\right) - \gamma_{mI,X}m_{I,X}^i \\ \frac{d}{dt}p_{I,X}^i &= \epsilon_{I,X}m_{I,X}^i - \gamma_{I,X}p_{I,X}^i, \end{cases} \quad (3.19)$$

where $R_A^i = p_{R-Y}^i$ (receiver complexes in compartments A).

For the transceiver dynamics, recall that we split the concentration of the AHL by compartment type, so we define the state of tx/rx_{A→B} to be $[X^T, R_B^T]^T \in \mathbb{R}_{\geq 0}^{N_A+2N_B}$, where $X = [X_A^T, X_B^T]^T = [X_A^1, \dots, X_A^{N_A}, X_B^1, \dots, X_B^{N_B}]^T$ and $R_B^j = p_{R-X}^j$ (receiver complexes in compartments B). Thus we have:

$$\text{tx/rx}_{A \rightarrow B} = \begin{cases} \frac{d}{dt}X_A^i &= \nu_X p_{I,X}^i - \gamma_X X_A^i + L_i X \\ \frac{d}{dt}X_B^j &= -k_{f,X} X_B^j (p_{R,X} - R_B^j) + k_{r,X} R_B^j - \gamma_X X_B^j + L_{j+N_A} X \\ \frac{d}{dt}R_B^j &= k_{f,X} X_B^j (p_{R,X} - R_B^j) - k_{r,X} R_B^j, \end{cases} \quad (3.20)$$

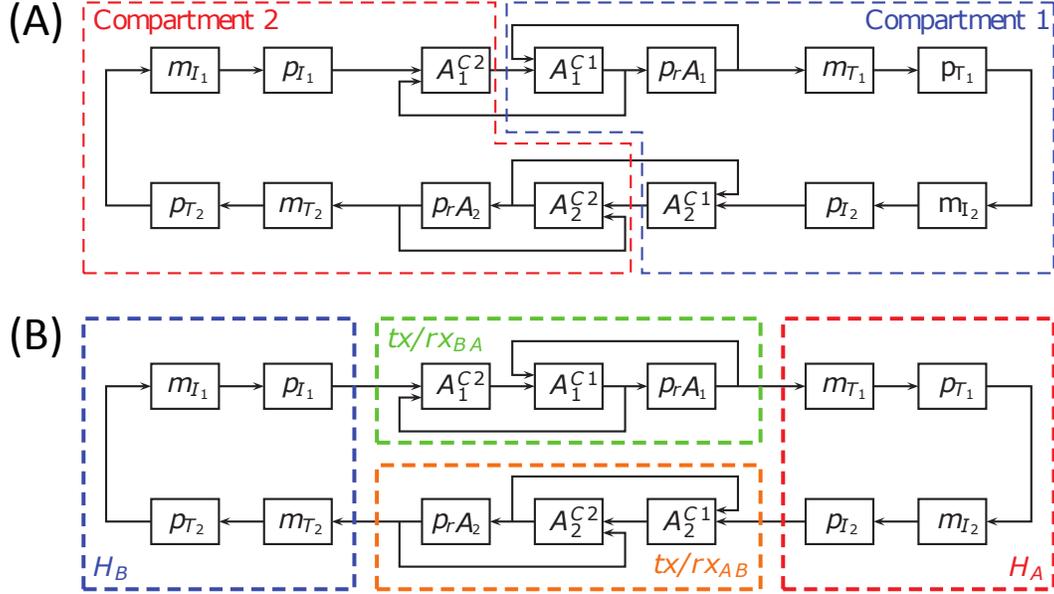


Figure 3.9: Compartmental lateral inhibition system model split into network blocks. For simplicity here we show a single compartment of type A communicating with a single compartment of type B. (A) The species of the entire network split by compartment. This is the most natural division but does not lend itself well to analysis as both compartments are two input-two output systems. (B) The same network now split into the transceiver and inhibitory modules. Note that if the network had n compartments of type A and B each, there would be n inhibitory modules of each type, but still just a single transceiver module of each type.

where L_i corresponds to the i^{th} row of the Laplacian matrix (3.8). The dynamics for H_B^i and $tx/rx_{B \rightarrow A}$ are obtained similarly by changing the indices appropriately.

Note that H_A and H_B satisfy Assumptions 3.2, 3.3, and 3.6 and $tx/rx_{A \rightarrow B}$ and $tx/rx_{B \rightarrow A}$ satisfy Assumptions 3.1 and 3.5. For H_A in (3.19), for each constant input R_A^{i*} , there is only one steady-state solution $(m_T^{i*}, p_T^{i*}, m_{I,X}^{i*}, p_{I,X}^{i*})$, which is a globally asymptotically stable hyperbolic equilibrium, due to the lower triangular structure of (3.19) with bounded nonlinearities. Furthermore, the static input-output map is decreasing:

$$T_A^i(R_A^{i*}) = K_1 \left(\frac{1}{1 + \left(\frac{K_2}{K_T} \left(\frac{1}{1 + (K_{R-Y}/R_A^{i*})^{n_{R-Y}}} + \ell_{P_{LuxI}} \right) \right)^{n_T}} + \ell_{P_{LtetO-1}} \right), \quad (3.21)$$

where $K_1 = \frac{\epsilon_{I,X}}{\gamma_{I,X}} \cdot \frac{V_{P_{LtetO-1}} N_{P_{LtetO-1}}^C}{\gamma_{m_{I,X}}}$ and $K_2 = \frac{\epsilon_T}{\gamma_T} \cdot \frac{V_{P_{LuxI}} N_{P_{LuxI}}^C}{\gamma_{m_T}}$ and the subsystem is monotone with respect to $K^U = -K^Y = \mathbb{R}_{\geq 0}$, $K = \{x \in \mathbb{R}^4 | x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0\}$ [5].

For $tx/rx_{A \rightarrow B}$ in (3.20), we see that in steady state, for a constant input $p_{I,X}^* \in \mathbb{R}^{N_A}$, the dynamic equations for R_B become zero, which implies that the first terms in the dynamical

equations for X_B are also zero. Therefore, due to the linearity of the remainder terms, there exists a unique solution for $[X_A^{*T}, X_B^{*T}]^T$:

$$\begin{bmatrix} X_A^* \\ X_B^* \end{bmatrix} = (-L + \gamma_X I_N)^{-1} \begin{bmatrix} \nu p_{I,X}^* \\ 0_{N_B} \end{bmatrix}. \quad (3.22)$$

The inverse of $-L + \gamma_X I_N$ exists since $-L$ is a positive semidefinite matrix (property of Laplacian matrices). The single solution for the steady state of R_B^i is given by

$$R_B^{i*} = \frac{p_{R,X}}{1 + \frac{k_{r,X}}{k_{f,X}} \cdot \frac{1}{X_B^{i*}}}, \quad (3.23)$$

where X_B^{i*} is as in (3.22). Note that the static input-output map $T_{AB}^{\text{tx/rx}}(p_{I,X}^{i*})$ is positive and increasing, because $-L + \gamma_X I_N$ is a positive definite matrix with nonpositive off-diagonal elements, and thus its inverse is a positive matrix [15]. Finally, to conclude that these steady states are asymptotically stable and hyperbolic, we write the Jacobian of the transceiver as:

$$J = \left[\begin{array}{cc|c} L - \gamma_X I_N & & 0 \\ & & 0 \\ \hline 0 & 0 & 0 \end{array} \right] + \left[\begin{array}{c|cc} 0 & 0 & 0 \\ \hline 0 & -D_{R_B} & D_{X_B} \\ 0 & D_{R_B} & -D_{X_B} \end{array} \right], \quad (3.24)$$

where D_{R_B} and D_{X_B} are diagonal matrices with elements $\{D_{R_B}\}_{ii} = k_{f,X}(p_{R,X} - R_B^{i*})$ and $\{D_{X_B}\}_{ii} = k_{f,X}X_B^{i*} + k_{r,X}$ for $i = 1, \dots, N_B$. The matrix J has negative diagonal terms and nonnegative off-diagonal terms, and there exists a $D = \text{diag}\{\mathbf{1}_N^T, k * \mathbf{1}_{N_B}^T\}$ with $1 < k < 1 + \frac{\gamma_X}{k_{f,X}p_{R,X}}$ such that the column sum of DJD^{-1} are all negative for all states in the nonnegative orthant. Note that this implies that the *matrix measure* of DJD^{-1} with respect to the induced one-norm is negative [35], and $\mu_D(J) = \mu_1(DJD^{-1}) < 0$. Therefore, for each constant input, the steady state is globally asymptotically stable [123]. Moreover, it is a hyperbolic equilibrium since $\text{Re}\{\lambda_k(J)\} \leq \mu(J) < 0$ [35]. The transceiver is monotone with respect to the cones in Assumption 3.5 since the Jacobian off-diagonal terms are all positive and the dependence on the input variable $p_{I,X}$ is positive [5].

To find stable steady-state patterns where all the compartments of the same type have the same final value, let the network be an equitable graph \mathcal{G} with respect to the compartment types. The transceiver input-output map decouples into the scalar map

$$T_{AB}(\tilde{z}_A) = \frac{1}{1 + \frac{k_{r,X}}{k_{f,X}} \cdot \frac{\gamma_X + \bar{d}_{AB} + \bar{d}_{BA}}{\bar{d}_{BA}\nu} \cdot \frac{\gamma_X}{\tilde{z}_A}}, \quad (3.25)$$

where \bar{d}_{AB} and \bar{d}_{BA} are as in (3.14). As discussed in Section 3.3.2, we look for steady states that are fixed points of $\bar{T}_A(\cdot)$ and $\bar{T}_B(\cdot)$ and apply the patterning condition (3.17).

As an illustration, we consider the simplest case where one compartment of each cell type communicate via a single channel. Then we generated the patterning region seen in Figure 3.10. The parameters used are given in Table 3.1 and bear a heavy resemblance to those in Parameter Set 1 of Table 2.2. It is important to note that we used identical parameters for

Table 3.1: Parameter values for compartmental lateral inhibition system simulation

Variable	Description	Units	Parameter Value
γ_X γ_Y	Degradation rates of AHLs	s^{-1}	7.70×10^{-4} [33]
γ_T	Degradation rate of TetR	s^{-1}	2.89×10^{-4} [84]
$\gamma_{I,X}$ $\gamma_{I,Y}$	Degradation rate of autoinducer synthases	s^{-1}	1.16×10^{-3} [84]
γ_{mT} $\gamma_{mI,X}$ $\gamma_{mI,Y}$	Degradation rate of mRNA	s^{-1}	5.78×10^{-3} [112]
$V_{P_{LuxI}}$	mRNA production velocity rate for P_{LuxI}	s^{-1}	0.26 [24]
$V_{P_{LtetO-1}}$	mRNA production velocity rate for $P_{LtetO-1}$	s^{-1}	0.3 [90]
$N_{P_{LuxI}}$	Copy number for P_{LuxI}		5
$N_{P_{LtetO-1}}$	Copy number for $P_{LtetO-1}$		5
C	Concentration of a single protein/mRNA in a typical bacterium	M	1.5×10^{-9} [81]
K_{R-X} K_{R-Y}	Dissociation constant of receiver complexes to P_{LuxI}	M	1.5×10^{-9} [24]
K_T	Dissociation constant of TetR to $P_{LtetO-1}$	M	1.786×10^{-10} [92]
n_{R-X} n_{R-Y}	Hill coefficients for P_{LuxI}		2 [24]
n_T	Hill coefficient for $P_{LtetO-1}$		2 [16]
$\ell_{P_{LuxI}}$	mRNA leakage of P_{LuxI} normalized to $V_{P_{LuxI}}$		1/167 [24]
$\ell_{P_{LtetO-1}}$	mRNA leakage of $P_{LtetO-1}$ normalized to $V_{P_{LtetO-1}}$		1/5050 [90]
ϵ_T	Translation rate for TetR	s^{-1}	6.224×10^{-6}
$\epsilon_{I,X}$ $\epsilon_{I,Y}$	Translation rates for autoinducer synthases	s^{-1}	2.655×10^{-5}
ν_X ν_Y	Catalytic rates of autoinducer synthases to AHL	s^{-1}	0.01335 [119]
$k_{f,X}$ $k_{f,Y}$	Binding rates of receiver complexes	$s^{-1}M^{-1}$	1×10^9 [137]
$k_{r,X}$ $k_{r,Y}$	Dissociation rates of receiver complexes	s^{-1}	50 [1, 24, 137]
$p_{R,X}$ $p_{R,Y}$	Constitutive levels of total receptor proteins	M	variable
d_{12}	Diffusion rate of AHLs across channel length ℓ_{12}	s^{-1}	variable

both diffusible molecules X and Y even though this is likely not the case in reality. As will be discussed further in Section 3.4.2, the currently well-studied AHLs come from a variety of different organisms with different native systems, so their parameters and even their mode of transport will likely not match. As more information becomes available, we can easily adjust the parameters in our model as necessary. Given the size of the patterning region shown in yellow, we are hopeful that this system will remain experimentally viable even with discrepancies between AHL parameters.

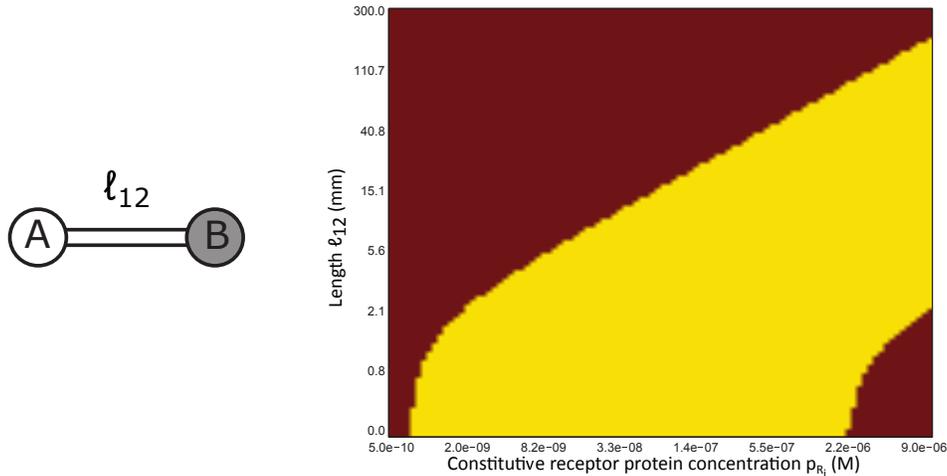


Figure 3.10: Patterning region for compartmental lateral inhibition. Left: We are examining the simplest geometry of one compartment of each cell type connected by a single channel of length ℓ_{12} . Right: By varying the channel length (y-axis) and receptor protein concentrations (x-axis), we analytically determine whether the system given by the parameters in Table 3.1 will exhibit contrasting patterns. The yellow region is where one compartment is “on” and the other is “off” and the red regions are where both compartments are “on” or both compartments are “off.”

Figure 3.10 maps the regions over the pairs $(p_{R,i}, \ell_{12})$ where contrasting patterns emerge. This does imply that we alter the constitutive amount of receptor protein in both cell types simultaneously. The reason we chose these two parameters is that they should be the easiest to manipulate experimentally. To change $p_{R,i}$, we can swap out the constitutive promoter. When each compartment is a square of side w and the channel is of length ℓ_{ij} and width w , then by [38] the edge weight is:

$$d_{ij} = \frac{D_{\text{AHL}}}{\ell_{ij}w} = k \frac{D_{\text{AHL}}}{\ell_{ij}^2}, \quad (3.26)$$

where we let the width be a factor k of the length (i.e. $w = \ell/k$). This means that we can change the diffusivity by changing the channel length, which can be accomplished much faster than manipulating our plasmids via cloning. We consider the diffusivity coefficient for AHL in water at 25°C [125]: $D_{\text{AHL},25^\circ\text{C}} = 4.9 \times 10^{-10} \text{ m}^2/\text{s}$.

We obtain patterning within a wide range of realistic parameters. We can intuitively understand the graph if we consider the constitutive receptor concentrations as a measure of how “sensitive” our cell types are to sensing incoming AHL of the correct type and the channel length as an inverse measure of how strongly our cell types can produce and send AHL. So the non-patterning red region in the upper-left of Figure 3.10 is where either the channel length is too long or there is not enough receptor protein. These situations cause P_{LuxI} to remain inactive in both compartments. The non-patterning region in the lower-right

is where either the channel length is too short or there is too much receptor protein. These situations cause P_{LuxI} to be activated in both compartments either via leakage or even when only small amounts of AHL are being produced.

3.4 Lateral Inhibition Experimental Results

As of the writing of this dissertation, the compartmental lateral inhibition experiments are ongoing. The following write-up will cover everything that has been accomplished thus far and outline the ongoing and future work.

3.4.1 Experimental Design

(1) Choosing parts

Because this project was started from scratch and not built on top of existing work in our lab, a big part of the experimental design centered on obtaining the chosen parts and making sure that they worked in our hands. The exact rationale behind our parts selection is covered in detail below in Section 3.4.2.

(2) Testing individual parts

To help test individual parts, we constructed the following test plasmids:

- **Senders** produce AHL by expressing an autoinducer synthase from a strong promoter (Figure 3.11A).
- **Receivers** express a receptor protein from a strong promoter and also contain a reporter operon that produces RFP in response to receiver complexes (Figure 3.11B).
- **“Matching pairs”** are similar to our cell types, except that they contain the cognate autoinducer synthase and receptor protein pair. Because of this, they contain an internal feedback loop (Figure 3.11C).

Senders are used to test whether or not the autoinducer synthase is functioning properly as well as to examine the diffusion of AHL in different media. Receivers are used to test the binding of AHL to receptor proteins (crosstalk, too) and the activation of the P_{LuxI} promoter. “Matching pairs” are used to test the reporter operon and the TetR- $P_{LtetO-1}$ interaction.

Hoping to avoid proteomics and protein purification, we designed simple tests that would verify the behavior of small subsets of our chosen parts. Given how well-studied the *lux* system is, we felt that it was reasonable to expect at least that system to work without much trouble.

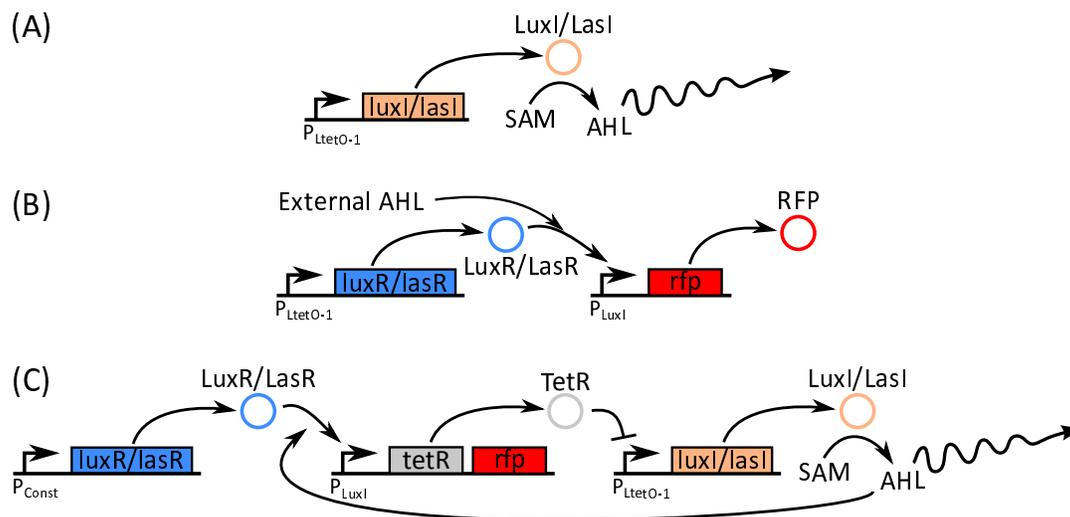


Figure 3.11: Testing plasmids used to verify working behavior of the different components of our compartmental lateral inhibition system. (A) Sender plasmids produce large amounts of AHL. (B) Receiver plasmids produce large amounts of receptor protein and should produce RFP in response to external AHL of the matching type. (C) “Matching pair” plasmids are similar to cell types A and B but produce the cognate autoinducer synthase and receptor protein pairs so that there is an internal feedback loop that causes P_{LuxI} to be activated.

The first test was to grow up receiver plasmids in liquid culture with and without AHL induction. We were expecting to see an increase in RFP fluorescence with the proper AHL induction compared to both no induction and the wrong AHL induction. This test would show that P_{LuxI} , our receptor proteins, and RFP are all working correctly.

Assuming our first test was successful, the next test was to allow senders to communicate with receivers. This could be tested in a number of different contexts, including co-cultured, co-transformed and grown in liquid culture, or separated spatially on agar. We expect the receivers to fluoresce in response to the AHL received from the senders when matched properly. This test would show that the autoinducer synthase was correctly producing AHL. The agar tests, in particular, would also verify that our AHLs could travel in our desired agar medium and give a sense of relative diffusion rates.

Finally, we grow up “matching pairs” in liquid culture with and without anhydrotetracycline (ATc) [82]. Because the “matching pairs” form receiver complexes, they should fluoresce more than a receiver without AHL or a plasmid with just the output operon. But because of the presence of TetR, in the absence of ATc the plasmid should fluoresce at an intermediate level because the production of the autoinducer synthase is being repressed. In the presence of saturating levels of ATc, the TetR repression is removed, allowing the system to fluoresce at a high level. This test would show that TetR and $P_{LtetO-1}$ are working and would finish covering all of the individual parts of our system.

(3) Test cell types with context

Assuming our individual parts tests succeed, the next step is to determine how well they are working in the context of the cell types for the compartmental lateral inhibition system. The sender and receiver test plasmids were designed for verification purposes only and thus were constructed in a nearly-identical manner to send strong signals and respond in a sensitive manner, respectively, which is also useful in making observations about the relative strengths of the two communication channels. These tests are then designed to allow us to evaluate how our cell type designs perform so we can determine any necessary tuning. Behavior may be different between our testing plasmids and cell types due to plasmid design, context effects, or other unknown factors.

First, we will run experiments with our cell types and our senders to gauge the receiving sensitivity of the cell types. Then we will run experiments with our cell types and our receivers to gauge the sending strength of the cell types. These tests can also be used to test our experimental assay (see the next step). These tests can be thought of as the sender-receiver tests but with our cell types acting as the receiver in the first test and then acting as the sender in the second test. If we are satisfied with the sending strengths and receiving sensitivities, then we can also run experiments with cell type A and cell type B where we seed the compartments at different times. The compartment type that is seeded first will have a growth and AHL production advantage, so the compartment type seeded second should light up and have its AHL production inhibited. In this way we can bias the competitive inhibition either way and check how close our tuning has made the two communication channel strengths.

(4) Design channels and experimental assay

This was done in parallel with the first three steps above. We need to restrict the communication between compartments to engineered channels and this will be done by creating cavities and filling them with agar. For initial experiments, we will start with the simplest case: one compartment of cell type A communicating with one compartment of cell type B across a single channel (as shown on the left of Figure 3.10). We seed our compartments with a small amount of dilute culture placed on top of the agar and allow the colonies to grow. Details can be found in Section 3.6.4, but during the design phase we try to keep the following in mind:

- Given that cell fate should be determined by noise, we will need to run large numbers of experiments simultaneously.
- What are good dimensions of the channel (length, width, depth) to use?
- For small volumes of agar, dehydration becomes important.
- As agar solidifies, a thin layer of water forms on the surface. Given the difference in diffusion through water and agar, will the orientation of the channels during growth be important?

- How much will growth temperature affect AHL diffusion? How long do our devices need to last for the experiments to run to completion?
- It is important to find a way to both fill channels with agar and to place dilute culture consistently.
- What is the best way to measure RFP fluorescence on agar?

(5) **Tune the strengths of cell types A and B**

This step will be iterative and depend on the results of the experiments from Steps 2 and 3 above. Lateral inhibition relies on the inhibition to be competitive between cells. In the case of CDI, there is only one cell type so this isn't an issue. But here we have two cell types using different communication molecules, so it is important that we adjust the signaling strengths properly using the tools available to us so that one cell type doesn't dominate the other.

For changing receiving sensitivity, we can change the constitutive promoter, the RBS, or the plasmid copy number for the receptor protein. For changing sending strength, we can change the RBS or the plasmid copy number for the autoinducer synthase. Much of the rest is dictated by our parts choices, such as the promoters that our receiver complexes interact with and the autoinducer synthase promoter, which is set by our choice of repressor. The actual plasmid designs are covered in Section 3.4.3.

(6) **Test different geometries**

We need our two cell types to be properly tuned to exhibit competitive inhibition in our simplest geometry where the compartment fates are determined by noise or initial conditions. The two main directions will be to change the dimensions of our simplest geometry and to test new geometries.

Sticking with the simplest geometry, we can verify our analysis by altering the channel lengths to show that at very long lengths both compartments remain "off," at very short lengths both compartments turn "on," and at intermediate lengths we get contrasting patterns. We may also be able to make some crude observations about a few of our system parameters based on the limits of our patterning region or the length of time for the system to hit steady state.

Changing geometries, we will examine geometries with equitable partitions of our two cell types in order to follow the analysis and we may end up being limited by what we can engineer. In particular, any graph that contains a cycle means that there must be at least one completely separated piece in the middle, so we may need to reconsider how we machine out our channels.

3.4.2 **Parts Selection**

The biggest goal of this project was simplicity. Starting from scratch with new parts was expected to be a challenge, so we wanted to use as many well-studied parts as possible,

Table 3.2: Candidate AHLs for Lateral Inhibition

Name	Abbr	Autoinducer Synthase	Native Species
N-(3-oxohexanoyl)-L-homoserine lactone	3OC6HSL	LuxI [44]	<i>V. fischeri</i>
N-(3-oxododecanoyl)-L-homoserine lactone	3OC12HSL	LasI [108]	<i>P. aeruginosa</i>
N-decanoyl-DL-homoserine lactone	C10HSL	BviI [31]	<i>B. vietnamiensis</i>
N-(3-oxooctanoyl)-L-homoserine lactone	3OC8HSL	TraI [65]	<i>A. tumefaciens</i>

especially ones that have worked well previously in our lab. To simplify cloning, we wanted to make both cell types as similar as possible.

By far the most important selections for this project are the AHLs because we would have to design around any differences or shortcomings of the two orthogonal communication channels and not too many AHLs have been characterized, much less tested in *E. coli*. A short list of candidate AHLs are given in Table 3.2. Of particular importance, note that because of the confinement of the receptor proteins within each cell type, we only need orthogonality of the AHL to receptor binding, not necessarily orthogonality of the receiver complex to promoter binding. The two options are to use two AHLs that are naturally orthogonal to each other or to alter LuxR ligand specificity to match our chosen AHLs [29, 30].

We planned to use LuxI (3OC6HSL) from the start due to its widespread use and characterization [46]. Initial work was planned with TraI (3OC8HSL), but consultation with other groups revealed that the LuxR-3OC8HSL crosstalk was “strong” and 3OC8HSL was not one of the chosen AHLs tested with the LuxR mutants in [29, 30], so it was abandoned as a candidate. Looking more closely at [30], BviI (C10HSL) was chosen as a candidate for use with LuxR-G2E-R67M*. William Holtz manually constructed *luxR-G2E-R67M* from *luxR* and we synthesized *bviI* based on a nucleotide sequence found online at UniProt. Unfortunately we were unable to get the parts tests to work for BviI and LuxR-G2E-R67M. Finally we decided on LasI (3OC12HSL). Not only is 3OC12HSL-LasR known to activate P_{LuxI} [53], but there have been synthetic biology papers published with LasI being used in *E. coli* [128], in particular one on a synthetic predator-prey system that used 3OC6HSL and 3OC12HSL as orthogonal communication channels [11]. Even though it is known that 3OC12HSL-LasR does not activate P_{LuxI} as strongly as 3OC6HSL-LuxR [53], the advantages were that we could use P_{LuxI} in both cell types and we were able to receive *lasR* and *lasI* on plasmids from Lingchong You.

We are aware that the *lux* box can be moved onto different promoters to make receiver complexes act as repressors instead of as activators [40]. This would have removed the need to have a separate repressor in our system. We decided against this because the promoter induction fold was stronger with LuxR as an activator [53] than a repressor [40] and we decided to have our output RFP reflect when a cell was receiving AHL signal (on the same

*Note that, confusingly, R67M follows the standard amino acid substitution nomenclature while G2E is a second generation LuxR variant from [29] that encompasses three amino acid substitutions.

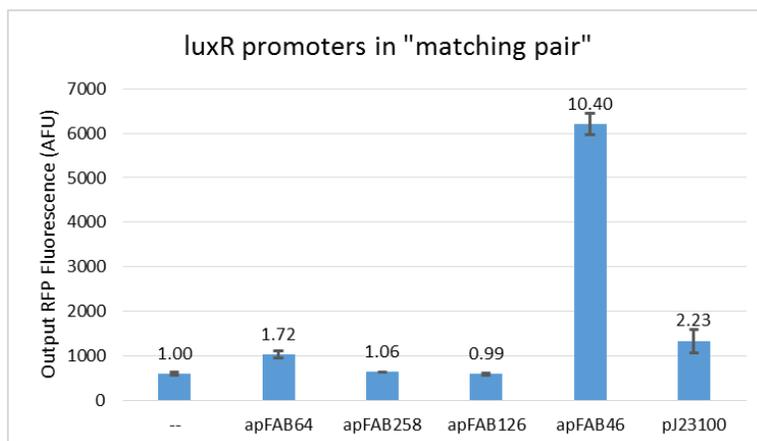


Figure 3.12: Early test to pick an initial constitutive promoter for the receptor protein. Test was only run with *luxR* because this was before we received working *las* sequences from the You lab. Here a reporter plasmid (i.e. receiver without the receptor protein) is co-transformed with various *lux* “matching pair” plasmids with different constitutive promoters on *luxR*. The numbers listed on each bar is how many times higher than the negative control (far left - no *luxR*) that output was. The promoter apFAB46 worked by far the best and was used with our *lasR* plasmids as well.

operon as our repressor).

For the rest of the parts we picked parts well-studied parts that we knew worked in our hands. For the repressor we chose TetR and $P_{LtetO-1}$ due to the ability to induce with ATc [82] and the low leakage of $P_{LtetO-1}$ [90]. For the reporter, we knew that red is more visible to the naked eye on agar than green, so we chose between mCherry [120] and mRFP1 [23]. Despite the literature stating otherwise, we found that mRFP1 in our hands tended to be slower-degrading and thus provided a brighter-looking red color at maximal expression. For this reason we chose to proceed with mRFP1.

At present, tuning of our cell types is not complete, so the final constitutive promoters for the receptor proteins is unknown. We are choosing from the Anderson [114] and BIO-FAB [100] constitutive promoter libraries. Based on an early test (see Figure 3.12), we are currently using apFAB46 for both cell types.

There are not any known genotype requirements for our chosen parts. The predator-prey paper [11] used Top10F' cells, which are closely related to DH10B and have the addition of the F' episome. We ran some initial tests in DH10B, Top10, and Top10F' cells and found that the behavior was not noticeably different (data not shown), so we proceeded with DH10B since it was the most readily available in our lab.

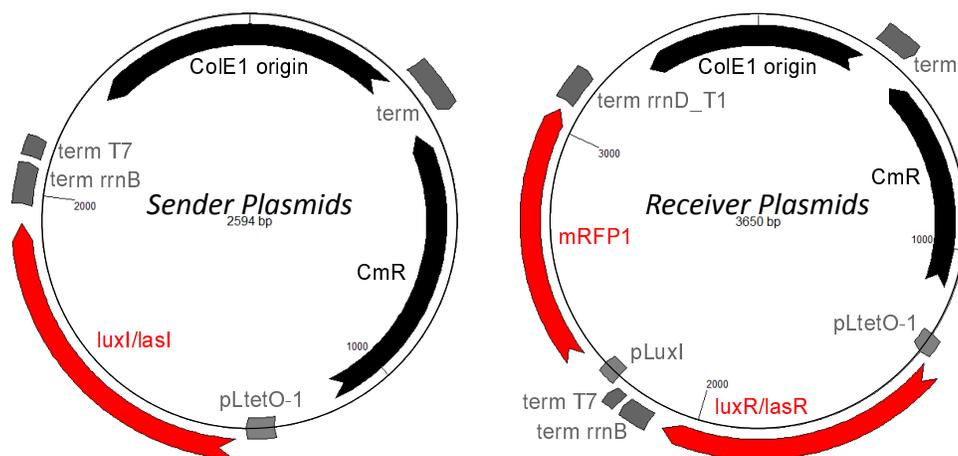


Figure 3.13: Plasmid designs for senders (left) and receivers (right) for compartmental lateral inhibition project. These are based on existing plasmids BBa_F1610 and BBa_F2620, which already contained *luxI* and *luxR*, respectively. These were later swapped out for *lasI* and *lasR* from plasmids from [11]. For certain experiments we used both the CmR versions shown here as well as AmpR versions. See Table 3.3 for a full plasmid list and descriptions.

3.4.3 Plasmid Design

We started with known quorum sensing plasmids BBa_F1610 and BBa_F2620 [114]. Plasmid BBa_F1610 is a *lux* sender device with polymerase per second (PoPS) input, meaning that it is lacking the promoter to produce LuxI (3OC6HSL). We choose to use $P_{LtetO-1}$ because it is a very strong constitutive promoter in the absence of TetR. Plasmid BBa_F2620 is a *lux* receiver device with PoPS output, meaning that LuxR is being constitutively produced and in the presence of 3OC6HSL will activate P_{LuxI} , which currently lacks an output gene. We place our chosen reporter *mRFP1* on P_{LuxI} . We then swapped in *lasI* and *lasR* from the plasmids ptetLuxRLasI-*luxCcdA* and pLasRLuxI-*luxCcdBs* from [11]. These plasmids all have ampicillin resistance with the senders on the pMB1 origin and the receivers on the ColE1 origin. We then moved these onto one of our preferred vector backbones, a common CmR/ColE1 backbone found in many BioBrick plasmids (see Figure 3.13). Note that *luxI* has the LVA degradation tag [3] and both *luxI* and *luxR* have genetic barcodes, all of which are artifacts carried over from BBa_F1610 and BBa_F2620. *LasI* and *lasR* have neither barcodes nor degradation tags. All subsequent uses of the *lux* and *las* genes are the same.

For Plasmid 1 we start with the BioBrick plasmid pBbe2c-RFP [83], which already has our desired CmR/ColE1 vector and *mRFP1* on P_{tetR} . We replace P_{tetR} with P_{LuxI} and move *tetR* between P_{LuxI} and *mRFP1*. Because we want our system to be able to switch between “on” and “off” states relatively quickly, we also add the LVA degradation tag to *tetR*. Plasmid 2 was started with the KanR/p15a vector backbone from pBbA8k-RFP. The four variants of Plasmid 2 (type A, type B, *lux* “matching pair,” *las* “matching pair”) can be constructed in any order based on availability of parts as they are formed by exchanging *luxR*

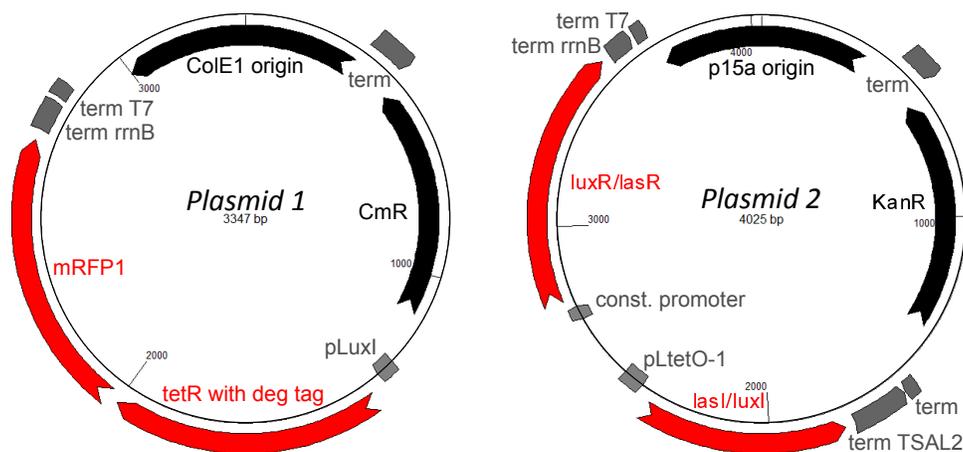


Figure 3.14: Two-plasmid design for compartmental lateral inhibition system. The reporter and repressor operon was placed on a higher copy number Plasmid 1 (left) for increased output visibility while the autoinducer synthase and receptor operons were placed on Plasmid 2 (right). This split design means that we can reuse Plasmid 1 in our experimental strains and swap out the parts on Plasmid 2 as necessary. These designs cover cell type A (*lasI/luxR*), cell type B (*luxI/lasR*), and the “matching pairs” (*luxI/luxR* and *lasI/lasR*). Experimental strains were made by co-transforming Plasmid 1 with either a variant of Plasmid 2 or an empty KanR plasmid (see Table 3.3) into DHB10 cells.

and *lasR* or *luxI* and *luxR*. The autoinducer synthase goes on P_{LuxI} and the receptor protein goes on a constitutive promoter. We place the two operons facing in opposite directions so that they do not interfere with each other (see Figure 3.14).

We are in the process of moving from the two-plasmid design shown in Figure 3.14 to the single plasmid design shown in Figure 3.15. The reasons for this change will be discussed in Section 3.5.2, but the goal is to move the operons from Plasmid 2 onto the higher copy number of Plasmid 1. We separate the operons and place the receptor protein operon downstream of our reporter operon because we do not care if the constitutive level of the receptor protein goes up slightly due to things like terminator read-through. We place the autoinducer synthase operon behind and in the opposite direction as the reporter operon.

3.4.4 Parts Testing Results

Receivers with AHL Induction:

For this test, the main goal was to show that our receiver parts were working. AHLs are commercially available in solid form (e.g. Sigma-Aldrich K3007 and O9139), but we had issues of 3OC12HSL being on back order indefinitely. So we decided to circumvent both the wait and possible issues of getting the AHLs into solution due to hydrophobicity and looked for another way to test AHL induction. What we settled on doing was to grow up a liquid culture of each sender plasmid to saturating density and then filter-sterilize the culture to

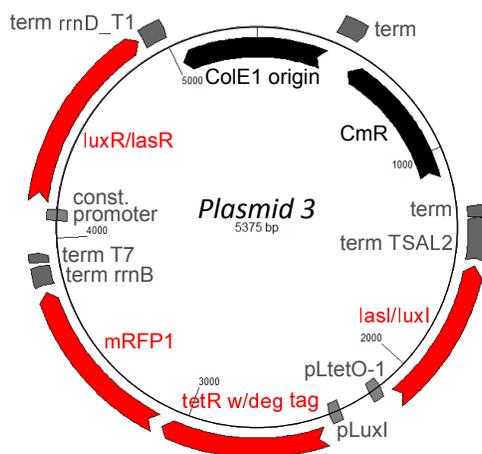


Figure 3.15: Combined single plasmid design for compartmental lateral inhibition system with the operons from Plasmid 2 added to Plasmid 1 (Figure 3.14). The goal is to increase both the sending strength and receiving sensitivity from the two-plasmid design. This design still covers cell type A (*lasI/luxR*), cell type B (*luxI/lasR*), and the “matching pairs” (*luxI/luxR* and *lasI/lasR*). In DH10B cells.

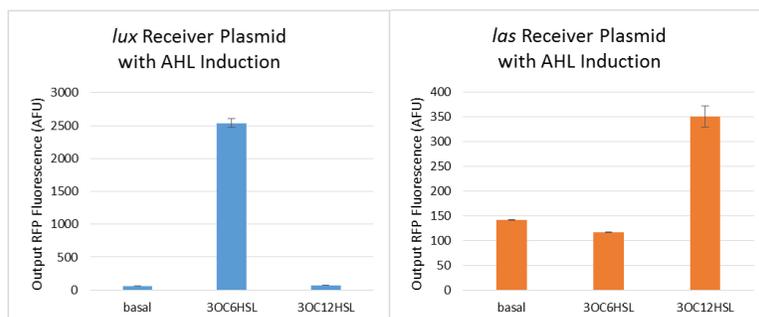


Figure 3.16: Data for parts testing showing AHL induction of *lux* receiver plasmid (left) and *las* receiver plasmid. Using unknown levels of AHL induction in liquid culture, but we can see that the RFP output does increase significantly with the proper AHL induction for both the *lux* and *las* systems with insignificant amounts of crosstalk.

remove the cells. What remained should be spent media plus the AHL of interest, meaning we can use this spent media to induce our receiver experiments. This assumes that our sender plasmids are working properly and producing AHL, which was not an original part of this test. Thankfully they did work and we show our results in Figure 3.16, but it does mean that we can only conclude whether or not the parts are working, not how well, since we don't know the actual induction strength. We also did check to see if there were significant amounts of crosstalk between the receptor proteins and non-cognate AHLs and we found that there were not.

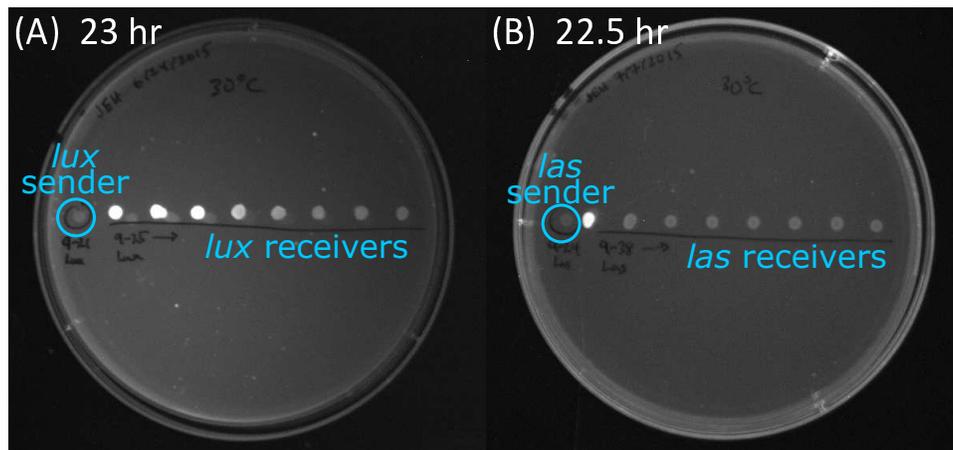


Figure 3.17: Ultraviolet (UV) imaging of sender and receiver experiments on plates of agar (no channels). The sender colony is seeded on the far left and a line of receiver colonies are seeded across the length of the plate. Over time we expect to see the receiver colonies light up radially outward from the sender colony. Images were taken after about 23 hr of growth in 30°C. (A) The *lux* system clearly works even without the channels to direct diffusion. (B) The *las* system does work, but much more weakly than the *lux* system. By 40 hr (not shown) the second *las* receiver colony from the left is also fully lit. The difference in diffusion strengths will need to be addressed via tuning.

Sender and Receiver Tests:

For this test we want to make sure that the senders and receivers work together like they are supposed to. We ran these tests both in liquid culture and on agar, but we only show the agar results here because they are closer to our experimental plasmids and assay. Here we show what we call a “sender and receiver” assay, where we use a standard 100 mm petri dish filled with agar and then seed one colony near the edge of the plate with a sender plasmid and seed many colonies of a receiver plasmid roughly evenly-spaced across the length of the petri dish. The sender colony will grow and continue to produce AHL, which will diffuse radially outward since our plate has no channels. What we see is that the receiver colonies light up over time with those closest to the sender colony fluorescing first (Figure 3.17).

“Matching Pair” Tests:

For this test we grow up our “matching pair” plasmids in liquid culture with and without ATc induction and compare against a receiver plasmid. As explained in Section 3.4.1, the culture without ATc induction should produce an intermediate fluorescence level, the culture with ATc induction should produce a higher fluorescence level, and the reporter plasmid gives us a background basal leakage level (see Figure 3.18). We were quite confident that TetR and $P_{LtetO-1}$ would work, but in particular wanted to make sure we didn’t make a mistake when placing both *tetR* and *mRFP1* on the same operon.

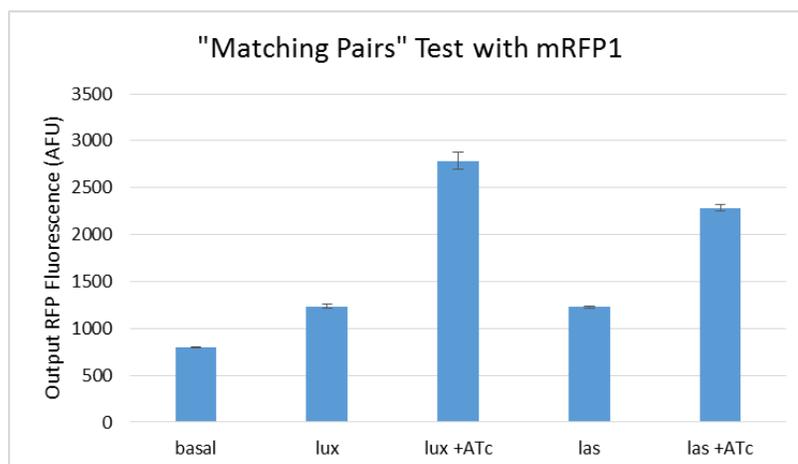


Figure 3.18: “Matching pair” experiment data verifying TetR behavior. Reporter plasmid gives us a basal fluorescence level (P_{LuxI} leakage), while a “matching pair” without ATc induction gives an intermediate fluorescence level and a “matching pair” with ATc induction gives a high fluorescence level.

3.4.5 Compartmental Lateral Inhibition Partial Results

The images shown in this section are from an early version of our assay based on the standard 100 mm circular petri dishes. For a full discussion of the development of the assay and further improvements in the works, see Section 3.6.4.

We are currently at the iterative stage between testing cell types with context and tuning. We want to determine if the receiving sensitivities and sending strengths of our two cell types are sufficient and sufficiently balanced. To test the receiving sensitivity, we ran experiments with senders and our cell types in channels (Figure 3.19) and observed that both cell types were receiving the AHL signal and fluorescing brightly on channel lengths at least up to 14 mm in reasonable amounts of time. We did not try longer lengths due to concerns over the ability of 3OC12HSL to diffuse over long distances (see Figure 3.17). We deemed it would be easier to weaken the *lux* system in the future.

Encouraged by this result, we ran biased cell type experiments in channels. Here we had cell type A communicating with cell type B in channels, but we delayed the seeding of one of the colonies by about 2.5 hours in order to bias the inhibition. Given the additional growth time, we expected the early colony to have more AHL in the channel and cause the late colony to fluoresce. However, what we actually saw was that, regardless of biasing, all of the colonies ended up in a state of slight fluorescence (Figure 3.20). Given that we know the cell types will fluoresce brightly in the presence of enough AHL, our conclusion is that the cell types are not producing AHL at a high enough level. “Sender and receiver” assays with the cell types replacing the senders confirmed that the cell types could not make the receivers fluoresce brightly (data not shown).

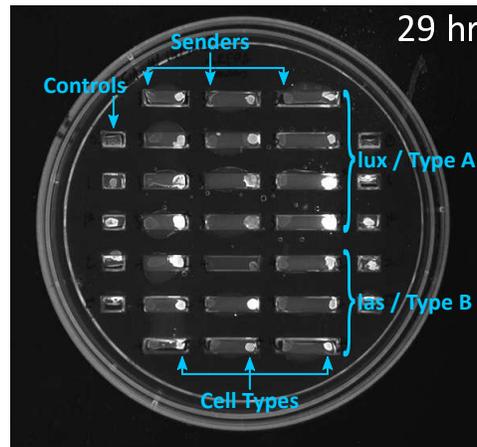


Figure 3.19: Receiving sensitivity experiment data in channels using early assay in circular inserts in petri dish. Channels are of increasing length from left to right. On the left side of each channel is a sender colony that does not fluoresce (minus a few where seeding failed). On the right side of the channel is a cell type colony of the appropriate type to sense the AHL. The top four rows are *lux* senders paired with cell type A and the bottom three rows are *las* senders paired with cell type B. The controls used (from top to bottom) are empty plasmid, cell type A ($\times 2$), cell type B ($\times 2$). This data highlighted a few issues with this experimental assay that we address in Section 3.6.4.

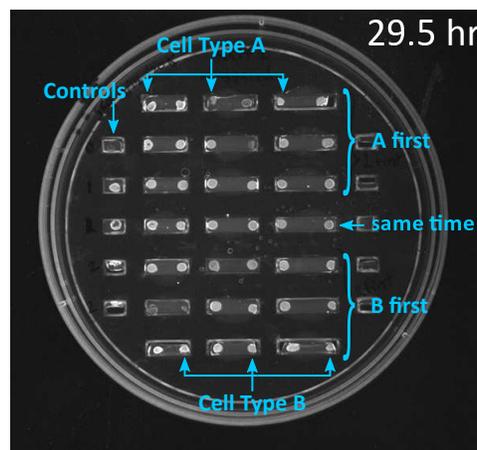


Figure 3.20: Cell type biased experiment data in channels using early assay in circular inserts in petri dish. Channels are of increasing length from left to right. On the left side of each channel is a type A colony and on the right side of the channel is a type B colony. The top three rows are had the type A colony seeded first, the bottom three rows had the type B colony seeded first, and the middle row both colonies were seeded simultaneously. All delayed seedings were done 2.5 hr after the initial seedings. The controls used (from top to bottom) are empty plasmid, cell type A ($\times 2$), cell type B ($\times 2$). Despite being given ample growth time, colonies appear all be fluorescing faintly with none brightly.

3.5 Lateral Inhibition Discussion

3.5.1 Implications of Compartmental Lateral Inhibition

The main contribution of the compartmental lateral inhibition system is to serve as an experimental demonstration of CDI-like patterning. It is intended to be a simple, experimentally feasible system to verify our theoretical framework and was born from the lack of a synthetic contact-mediated transport system. Because of this we had to engineer a lateral inhibition system using diffusion, causing us to modify our analysis to fit the new system. But while its future uses are unknown, the experimental setup contributes in other areas that may prove useful to future multicellular applications. The use of two orthogonal diffusible molecules in a single system has been achieved previously [11], but not in a patterning system. The approach to spatially separating colonies of different strains has been done before with logic gates on an agar surface [128], but we added engineered channels to allow for more directed communication and larger, more diverse geometries. Additionally, the UV imaging setup should prove to be a faster, less expensive way of assaying potentially large geometries as opposed to the use of flow cytometry on every colony seen in [128].

Finding or building a synthetic CDI system that can transport a transcription factor of our choosing between neighboring cells remains hugely important. In order to engineer any complex multicellular behavior, especially those related to spatial organization or shape, a large number of different communication channels will be necessary. Thus far very few tractable synthetic communication channels have been found, highlighted by the difficulties we have had getting just two diffusible molecules to work together in this project. We will continue to work towards getting this compartmental lateral inhibition system to work in order to confirm the veracity of our analysis.

Our graph theoretic approach to analyzing CDI systems fits intuitively with the actual interactions of a network of cells communicating via contact-based means. Our analysis is able to predict whether or not a potentially large network of cells can or will display equitable patterns based only on the graph structure and the steady-state input-output maps of the cell types involved. This can greatly reduce the amount of simulation work necessary and can help inform network design and parameter choices once a synthetic CDI system is finally engineered.

3.5.2 Lateral Inhibition Future Work

Immediate work is focused on the conclusion drawn from Figure 3.20 that the sending strength of the cell types is too weak. The initial cloning plan is to move from the two-plasmid design of Figure 3.14 (Plasmids 1 and 2) to the one-plasmid design of Figure 3.15 (Plasmid 3). Most importantly we are moving the autoinducer synthase operon to a higher copy number origin of replication, but moving everything to a single plasmid allows us to run experiments in a single antibiotic and saves us a co-transformation step, though every future tuning step will have to be done sequentially. Currently we are working to overcome

some replication slippage between two of the terminators. If this proves problematic, we can leave the receptor protein operon on the p15a vector.

Another option would be to move the autoinducer synthase operon onto an even higher copy number plasmid such as pMB1. We are choosing not to pursue this at the moment because the readily-available origins in our lab are SC101**, p15a, pBBR1, and ColE1 from the BioBrick collection [83]. ColE1 has the highest copy number amongst those and our cell types did respond to our senders (Figure 3.19), which had our autoinducer synthase genes on $P_{LtetO-1}$ and ColE1, which is the same as Plasmid 3.

Once the cell types send AHL strongly enough, we will turn our attention to balancing the strengths of the communication channels. The strength of each channel is a combination of many parts: the rate of production of the autoinducer synthase, the catalytic conversion rate to AHL, the diffusivity of the AHL through our medium, amount of receptor protein present, the binding affinity of AHL to receptor protein, the dissociation constant of the receiver complex to P_{LuxI} , and the activation strength of the receiver complex on P_{LuxI} . Of these, the catalytic conversion rate, the diffusivity, binding affinity, dissociation constant, and activation strength are essentially set by our choice of AHL and are difficult, if not impossible, to change at present. This leaves the rate of production of the autoinducer synthase and the amount of receptor protein present as tunable parameters. Because we have chosen TetR to be our repressor, it is easiest to change the 5'-UTRs of $P_{LtetO-1}$ instead of changing the promoter itself to alter the rate of production. The amount of receptor protein present has always been identified as a tunable parameter by swapping out the constitutive promoter in use with any of those from the BIOFAB or Anderson libraries or an inducible promoter without its inducer (e.g. $P_{LlacO-1}$).

All evidence points to the *lux* signaling channel being much stronger than the *las* signaling channel. A number of factors contribute to this, including LasR-3OC12HSL not activating P_{LuxI} as strongly as LuxR-3OC6HSL [53] and 3OC12HSL being a larger molecule (molecular weight 297.39 versus 213.23 g/mol) that is not freely diffusible. 3OC12HSL is transported through the membrane via an active-efflux pump in *P. aeruginosa* [107], meaning that there may be an additional internal/external concentration barrier to overcome that wouldn't be present in a freely diffusible compound such as 3OC6HSL. Given the large discrepancy, it is possible that we may deem the difference in signaling strengths too large to overcome in a manner that preserves the viability of our experimental assay (i.e. experiments take too long or don't ever fluoresce brightly enough). There are other candidate AHLs we could use, certainly more than were shown in Section 3.4.2, but most would require significant work including changing P_{LuxI} to another promoter in cell type B, introducing another possible source of strength discrepancy.

More work is also being devoted to improving the experimental assay with the use of larger, rectangular inserts and the pouring of agar and the seeding of colonies in a more consistent manner using robots. See Section 3.6.4 for details.

Table 3.3: Plasmids used in compartmental lateral inhibition system. All plasmids were transformed into the strain DH10B [52].

Name	Resistance	Origin	Description
pJH1-81	KanR	p15a	Empty plasmid.
pWH17-39	CmR	ColE1	Empty plasmid.
BBa_F1610	AmpR	pMB1	<i>lux</i> (3OC6HSL) sender device with PoPS input [114]. The basis for the sender plasmids (Figure 3.13).
BBa_F2620	AmpR	ColE1	<i>lux</i> (3OC6HSL) receiver device with PoPS output [114]. The basis for the receiver plasmids (Figure 3.13).
pJH4-81	AmpR	pMB1	<i>lux</i> (3OC6HSL) sender on backbone of BBa_F1610.
pJH9-21	CmR	ColE1	<i>lux</i> (3OC6HSL) sender.
pJH9-22	CmR	ColE1	<i>las</i> (3OC12HSL) sender.
pJH4-65	KanR	p15a	Random inverter from Turing project (pJH4-40 with <i>mRFP1</i>). Used to get <i>mRFP1</i> and terminator <i>rrnD_T1</i> .
pJH5-1	AmpR	ColE1	<i>lux</i> (3OC6HSL) receiver on backbone of BBa_F2620.
pJH5-4	AmpR	ColE1	pJH5-4 with 5'-UTR of <i>mRFP1</i> changed.
pJH5-29	AmpR	ColE1	P_{LuxI} reporter with no receptor protein (pJH5-4 minus <i>luxR</i>).
pJH9-35	CmR	ColE1	<i>lux</i> (3OC6HSL) receiver.
pJH9-36	CmR	ColE1	<i>las</i> (3OC12HSL) receiver.
pJH4-22	CmR	ColE1	Plasmid 1 from Figure 3.14. Based on pBbE2c-RFP [83].
pJH5-39	KanR	p15a	Plasmid 2 from Figure 3.14 with <i>luxR/luxI</i> (<i>lux</i> “matching pair”).
pJH5-64	KanR	p15a	Plasmid 2 with <i>luxR/lasI</i> (cell type A).
pJH5-66	KanR	p15a	Plasmid 2 with <i>lasR/luxI</i> (cell type B).
pJH5-68	KanR	p15a	Plasmid 2 with <i>lasR/lasI</i> (<i>las</i> “matching pair”).
pJH9-43	CmR	ColE1	Plasmid 3 from Figure 3.15 with <i>luxR/lasI</i> (cell type A).
pJH9-44	CmR	ColE1	Plasmid 3 with <i>lasR/luxI</i> (cell type B).

3.6 Lateral Inhibition Materials and Methods

3.6.1 Computational

Analytical models were investigated in MATLAB Version 8.5.0 (R2015a). Figure 3.10 was generated by varying the specified parameters in a grid and analytically solving for the stability of the contrasting pattern steady state. No simulation was necessary as the model was already given and our existence and stability criterion was a function of the model and the steady state values, which we can solve for.

Data from experiments were measured using the instruments specified in Section 3.6.3 and output to files. These files were opened in Microsoft Excel 2013 and the data was analyzed and plotted using standard Excel functions.

3.6.2 Construction of Plasmids

Plasmid construction was done via circular polymerase extension cloning (CPEC) [111] and 'Round-the-horn site-directed mutagenesis [58]. CPEC designs were started by using the j5 DNA assembly design automation software [60] to generate an initial set of oligonucleotides, which were then checked manually and tweaked based on the online Thermo Fisher Scientific Tm Calculator. All polymerase chain reactions (PCRs) were performed using the Phusion Hot Start II High-Fidelity DNA Polymerase (Thermo Scientific F-549).

We constructed our senders starting with BBa_F1610 and adding the promoter $P_{LtetO-1}$ using o5-33/o5-34 to create pJH4-81. This sender was not on our desired vector, so we moved the sender onto the Cm/ColE1 backbone of pBbE2c-RFP using o8-66 to o8-69 to create pJH9-21. Then we replaced *luxI* in pJH9-21 with *lasI* from ptetLuxRLasI-*luxCcdA* [11] using o7-10 to o7-13 to create pJH9-22.

We constructed our receivers starting with BBa_F2620 and adding $mRFP1$ to P_{LuxI} by pulling the gene and terminator from pJH4-65, which was an inverter with $mRFP1$ we had constructed for the Turing project, using o5-35 to o5-38 to create pJH5-1. We found that we got better behavior when we changed the 5'-UTR on $mRFP1$ using o5-43/o5-44 to create pJH5-4. From pJH5-4, we created a reporter test plasmid pJH5-29 by removing the *luxR* operon using o6-8/o6-9. Then we moved the receiver onto the Cm/ColE1 backbone of pBbE2c-RFP using o8-67/o9-7 and o8-68/o9-8 to create pJH9-35. Finally we replaced *luxR* in pJH9-35 with *lasR* from pLasRLuxI-*luxCcdBs* [11] using o7-26 to o7-29 to create pJH9-36.

To build our reporter plasmid pJH4-22 (Plasmid 1 in Figure 3.14), we started with the BioBrick plasmid pBbE2c-RFP [83] and flipped *tetR* to be with $mRFP1$ and placed them on P_{LuxI} using the primers o4-17 to o4-20. We ended up not needed to modify this plasmid any further (until construction of Plasmid 3, which is not covered here).

Constructing Plasmid 2 was a roundabout process that involved failed attempts to use *bviI* and *luxR-G2E-R67M*. In the interest of time and space, the Genbank file for pJH5-39 is provided in the Appendix. It is a small enough plasmid that synthesis or construction from the other plasmids given here is feasible. From pJH5-39, we can swap out *luxI* for *lasI* from ptetLuxRLasI-*luxCcdA* using o7-6 to o7-9 to create pJH5-64 (cell type A) or we can swap out *luxR* for *lasR* from pLasRLuxI-*luxCcdBs* using o5-78/o6-25 and o5-80/o5-81 to create pJH5-66 (cell type B). Then we perform the opposite swap (either *lasI* into pJH5-66 or *lasR* into pJH5-64) to create pJH5-68 (*las* "matching pair").

Table 3.4: Oligonucleotides used in the construction of the compartmental lateral inhibition plasmids. Here “for” denotes a forward primer, “rev” denotes a reverse primer, “rm” is short for remove/deletion, and “RTH” indicates use in ‘Round-the-horn site-directed mutagenesis.

Name	Description	Sequence
o4-17	get pBbE2c-RFP vec for	agtagcttaataagatcttttaagaaggagatatacatatggcg
o4-18	get pBbE2c-RFP vec rev	cttttctagatctttattcgactataacaaaccattttcttgcgtaaacctgtacgatccta caggtgacgtcgatatctggcga
o4-19	get pBbE2c-RFP tetR for	tggtttgatatagtcgaataaagatctaggaaaaagctcatataactagagtaagaggtca atgatgtctagattagataaaagt
o4-20	get pBbE2c-RFP tetR rev	tctcttcttaaaagatcttattaagctactaaagcgtagtttctgctgttgagcagaccc actttcacatttaagt
o5-33	RTH add pLtetO-1 for	tcagtgatagagatactgagcactactagagaaaggagaaatactagatgactataat
o5-34	RTH add pLtetO-1 rev	tagggatgtcaatctctatcactgatagggactctagaagcggccgcaattccagaaat
o8-66	get CmR/ColE1 vec for	tggcctttctgctttatacctagggcgttcggct
o8-67	get CmR/ColE1 vec rev	tctctatcactgatagggagacgtcgatatctggcgaaaatgagacgt
o8-68	get sender for	tcgcagatatcgacgtctcctatcagtgatagagattgacatccc
o8-69	get sender rev	ccgcagccgaacgccttaggtataaacgcagaaaggccaccgg
o7-10	sender rm luxI for	gcgactggcggtttcatgataactagagccaggcatca
o7-11	sender rm luxI rev	cgcgccgaccaattgtacgatcatctagtatttctcctctttctct
o7-12	get lasI for	agaggagaaatactagatgatcgtaaaaattggcggcgcaaga
o7-13	get lasI rev	tgcttggtctagtattatcatgaaaccgcagtcgctgt
o5-35	get BBa_F2620 vec for	cactctccggggcgtactagtagcggccgctgca
o5-36	get BBa_F2620 vec rev	acctttctcctttaatgaattgttttattcgactataacaaacca
o5-37	get mRFP1 for	acaattcattaaaggagagaaaggtaccatggcg
o5-38	get mRFP1 rev	gccgctactagtagcggcggagagtggtcacc
o5-43	RTH change 5UTR for	aaaagatcttttaagaaggagatatacatatggcgagtagcgaagacgttatcaagagt
o5-44	RTH change 5UTR rev	gaattcccaaaaaaacgggtatggagaatttattcgactataacaaaccattttcttgcg
o6-8	RTH rm luxR for	tactagagtcacactggctcacctcgggt
o6-9	RTH rm luxR rev	ctctagaagcggccgcaattc
o9-7	get CmR/ColE1 vec for (for use with o8-67)	ctcccggggcctagggcgttcggctg
o9-8	get receiver rev (for use with o8-68)	acgccttaggcggccgggagagtggtcac
o7-26	receiver rm luxR for	ttgggtcttattactctctaataactagagccaggcatca
o7-27	receiver rm luxR rev	cgtcaaccaaggccatctagtatttctcctctttctct
o7-28	get lasR rec for	agaggagaaatactagatggccttgggtgacgg
o7-29	get lasR rec rev	tgcttggtctagtattagagagtaataagaccctaaatcaacggcca
o5-78	type rm luxR for	gggtcttattactctctaataaggatccaaactcgagtaaggatctcca
o6-25	type rm luxR rev	aaccaaggccatctagtatttctcctctttctctagtaatga
o5-80	get lasR type for	gaggagaaatactagatggccttgggtgacg
o5-81	get lasR type rev	gagttggatccttattagagagtaataagaccctaaatcaacg
o7-6	type rm luxI for	cgcgccgaccaattgtacgatcatttttttctccttattttctcca
o7-7	type rm luxI rev	ggaacagcgactggcggtttcatgataataataatcatcgcaagacttgatcggtg
o7-8	get lasI type for	gcaccgatcaagctctcgcatgattattattatcatgaaaccgcagtcgctgt
o7-9	get lasI type rev	ctggagaaaataaggaggaaaaaaatgatcgtaaaaattggcggcgcaaga

3.6.3 Experimental Conditions and Procedure

Liquid Culture Experiments:

Parts testing experiments in liquid culture were run in EZ Rich defined medium (Teknova M2105) in 96-well deep well plates (DWP) with 1.7 mL round wells using an AeraSeal breathable sealing film (Sigma Aldrich A9224) to cover. All liquid culture growth was performed in an INFORS HT Multitron Standard shaker with 25 mm throw at 37°C and 900 rpm. Antibiotic concentrations of 25 $\mu\text{g}/\text{mL}$ for Chloramphenicol (Cm) and 50 $\mu\text{g}/\text{mL}$ for Kanamycin (Kan) and Ampicillin (Amp) were used.

Each bacterial strain was streaked out from glycerol freezer stock onto LB plates with the appropriate antibiotic(s) and colonies were grown overnight in a 37°C warm room. Six colonies of each strain were transferred to 400 μL wells of a DWP with EZ Rich and the appropriate antibiotic(s). These cell growth plates were placed on the shaker to grow for about 14 hours. Upon removal from the shaker, the experimental plates were sampled and 150 μL from each well was transferred into 96-well black plates with clear flat bottoms (Corning #3631). Bulk fluorescence measurements were taken using a Molecular Devices SpectraMax M2 microplate reader. Settings used were OD measured at 600 nm and RFP measured with 565/620 excitation and emissions wavelengths taken with 30 reads and medium sensitivity flash mode.

In each experimental plate, two controls were used: an “empty” plasmid without RFP, and a “blank” row with just media (no cells). When analyzing the data, background OD levels were calculated by averaging the OD measurement of the “blank” wells for each plate and then subtracted from the OD measurements of the other wells. Next the background fluorescence was calculated by dividing the fluorescence measurements of the “empty” wells by their adjusted OD measurements and averaging for each plate. Final fluorescence values for the wells of interest were calculated by dividing their fluorescence measurements by their adjusted OD measurements and then subtracting off the background fluorescence. Each experiment was run with 6 replicates, average and standard error values were calculated and plotted against the positive induction data in Microsoft Excel.

For AHL induction, sender plasmids were grown up overnight to saturating OD in EZ Rich media with the appropriate antibiotic(s) and then filter sterilized through a 0.22 μm Corning bottle top filter. The resulting spent media contains antibiotic and the AHL of interest at an unknown concentration. The AHL degrades over time so it is recommended to filter a fresh batch the same day as its expected use. We induced each AHL at 25X (i.e. 16 μL spent media into 400 μL total well volume).

For ATc induction, we induce at 20X from a stock of 5 $\mu\text{g}/\text{mL}$, so the final concentration should be 0.54 μM .

Channel Experiments:

See Section 3.6.4 below for a full discussion on the design of our devices.

Each bacterial strain was cultured from glycerol freezer stock directly into EZ Rich media with the appropriate antibiotic(s) and grown up to saturating OD overnight in 37°C warm

room shaker at 200 rpm. Antibiotic concentrations of 25 $\mu\text{g}/\text{mL}$ for Chloramphenicol (Cm) and 50 $\mu\text{g}/\text{mL}$ for Kanamycin (Kan) and Ampicillin (Amp) were used. For small experiments on a single plate, 5 mL of culture in 50 mL tubes is sufficient. For larger experiments with multiple plates using the robotics facilities, larger volumes were grown in 250 mL conical flasks. The OD600 was measured on a Beckman Coulter DU 800 Spectrophotometer and then each culture was diluted to 0.25 OD. Using either a multichannel pipette or the robotics facility, compartments were seeded by placing $\approx 0.5 \mu\text{L}$ of dilute culture on top of the agar. Controls consisted of placing colonies of senders (negative), receivers (leakage), and/or cell types (crosstalk) in channels by themselves.

Experiments were run with the devices covered by their lids on a flat surface in a 30°C warm room with the channels face down. This orientation mimics standard practice for agar plates and empirically seemed to reduce the colony spreading and improve experiment behavior. Periodically the devices were removed from the warm room for imaging on the a UVP BioSpectrum imaging system. We used the VisionWorksLS analysis software to capture images using the Ethidium bromide filter (570-640 nm) with UV transillumination provided by a FirstLight UV Illuminator. Images were taken with on chip integration and camera settings (aperture, zoom, exposure, gain) that were manually adjusted. Devices were then returned to the warm room for further growth.

3.6.4 Development of Experimental Assay

Imaging:

There currently isn't a quantitative method that we know of to measure the fluorescence of colonies on agar. Fluorescence microscopy is ill-suited to this task due to the colonies being separated by significant space and the colonies also not being single layers of cells. And although ideally we want to be able to see the colonies fluorescing with our naked eyes, photographs are unreliable due to factors such as inconsistent background lighting and colors, glare, and lack of sensitivity. Instead, we decided on ultraviolet (UV) imaging. The closed, dark environment should block out background interference and exciting the proper wavelengths that match our RFP will illuminate intermediate levels of fluorescence that might not be apparent in white light. The imaging width is also on the order of inches, which is appropriate for the scale of our experiments. We are aware that UV imaging will suffer from some inconsistency issues as anything involving a lamp will depend on how "warm" it is at any given time, but we are most concerned with comparing the fluorescence of the compartments to each other and not as concerned with comparing between different time points.

Channel Creation:

For engineering the channels, all initial experiments will be the simplest case where two compartments are connected by a single channel (as in Figure 3.10). Because we use the same width for the compartments and the channel, we are creating small, rectangular cavities that we fill with agar. We decided to build devices by laser cutting clear acrylic using a Universal

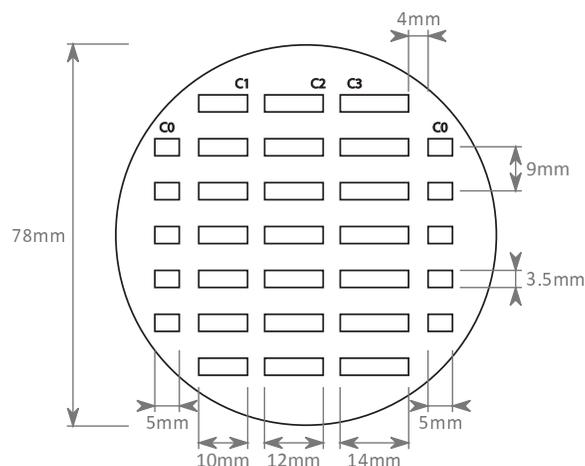


Figure 3.21: Layout and dimensions for first insert. Fits within a 90 mm diameter petri dish. If dimension is not explicitly shown, assume that channels are evenly spaced apart. Channels C0 are intended for control colonies, while C1 to C3 are increasing in length, since we were initially unsure of the diffusive capabilities of our plasmids. Rows are evenly spaced at 9 mm intervals to match the spacing of a standard multichannel pipetter to aid with manual seeding.

Laser Systems VLS3.50 laser cutter with a 50W CO₂ laser available at the CITRIS Invention Lab on campus (<http://invent.citris-uc.org>). This laser is meant to cut all the way through materials and can handle acrylic of thickness up to 1/4". For now we are using 1/8" thick extruded acrylic sheets that are commercially available at most hardware stores. The laser cutter can precisely cut out rectangular footprints through our acrylic, but they need to be attached to something in order to create a backing to hold the agar.

Following standard lab protocol, colonies are often grown up in warm rooms in circular beds of agar in petri dishes to provide nutrients to cells while keeping out contaminants and keeping in moisture. Our initial channel inserts were designed to fit into the smaller end of a 100 × 15 mm petri dish (VWR 25384-088). We created our insert designs using Adobe Illustrator (.ai files) and fed these into the Laser Interface+ software that came with the laser cutter. Our initial insert design is shown in Figure 3.21. Because we were unsure of the diffusivity of our AHLs, we made a range of channel lengths (10-14 mm) and were prepared to redesign if necessary. The width was arbitrarily chosen to be 3.5 mm based loosely on the size of colonies we saw on other plates. We fit seven rows of channels vertically spaced 9 mm apart to match the spacing of tips on a multichannel pipetter, which allows us to seed an entire column of compartments simultaneously. With the extra space on the sides, we added single compartments C0 to place controls. Figures 3.19 and 3.20 show experiments using Insert 1. However these same figures display some of the problems that arose with Insert 1:

- The smaller the channel length, the higher the danger of dehydration (see the C0 channels especially).

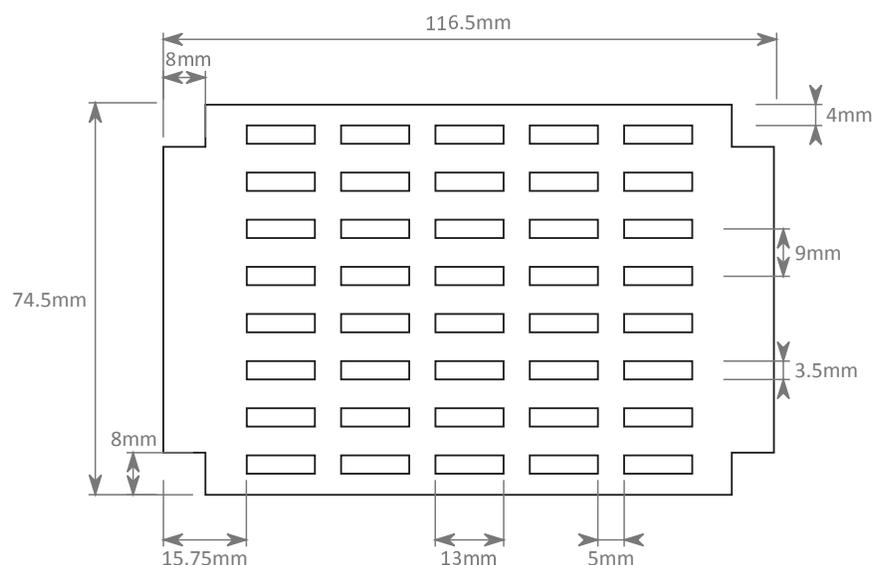


Figure 3.22: Layout and dimensions for second insert. Fits inside a single-well plate and are for use with robotics facilities. Insert is symmetric and channels are evenly spaced apart. Because we can fit up to forty channels now per plate, controls we will place in the same channels, which are now 13 mm in length. Rows are evenly spaced at 9 mm intervals to match the spacing of a multichannel pipetter.

- Agar was pipetted into the channels manually and proved to be very inconsistent. Due to factors such as hardening within pipette tips, splashing, and uneven aspiration or dispensation, the channels were sometimes rough and uneven, which affected both diffusion and colony growth.
- Seeding colonies with a multichannel pipette proved nontrivial. With such small volumes of liquid culture, it often would not leave the pipette tips and end up beading. Contacting the agar surface with the pipette tip sometimes formed indentations that affected colony growth and spread. When trying to seed many rows at once, it was difficult to prevent one of the ends from poking the agar surface. Sometimes in the process one or more rows would fail to take (see the left compartment of the 5th row of channel C2).
- The limited number of channels per plate (7 of C1-C3) made it difficult to draw conclusions when many of the replicates suffered from the issues listed above.

We tried to address these issues by developing a second insert. The biggest source of problems came from human error in pipetting, so our goal from the start was to try to take advantage of the robotics facilities at our disposal to set up and run our experiments consistently.

Most laboratory automation robots, depending on their function, are designed to use microplates, which follow the standards ANSI/SLAS 1 to 4 - 2004, including footprint di-

mensions. Therefore we needed to make sure our devices also fit into these standards. We decided against creating a makeshift petri dish holder and instead decided to redesign our inserts to be placed inside of Nunc OmniTray single-well plates (VWR 62409-600), which conveniently also come with lids. Our new insert design is shown in Figure 3.22. Notably Insert 2 is much larger and more regular. The insert has outer dimensions 74.5×116.5 mm, which fits snugly within the single-well plate to keep consistent positioning while allowing enough room for the insert to lie flat against the bottom. We have settled on a length of 13 mm for our channels and fit 40 of them on a single insert. We know that our AHLs can reach across this distance (Figure 3.19 - cell types fluoresced highly in receiving sensitivity test in C3, which was 14 mm) and a 13 mm length has the added benefit of placing the center of our compartments directly where the center of the wells on a 96-well plate would be. This gives us some flexibility in terms of what robot we want to use, as some are better-suited to using 96 tips at a time and others are better at using 8 (a column) at a time. With this in hand we are working on developing protocols to use a robot to both fill the insert with agar and seed colonies (see Section 3.6.5).

For more complicated geometries our inserts will likely need to be created in a different manner. Take, for example, the square geometry shown on the left of Figure 3.5. This graph contains a cycle, meaning that all of the inner acrylic would fall out once the outer cut is made. While we can still cut out a middle piece, inconsistency becomes an issue again while trying to reposition this floating element in the middle. One future solution might be 3D printing [19].

Insert Attachment:

Initially we tried an acid-free, clear spray adhesive. These are easily bought, dry quickly, and create a fairly strong bond between insert and petri dish. Unfortunately despite the claims of drying clear, the spray adhesive would leave an inconsistent amount of white residue based on the spray application (i.e. spread and thickness) that was visible through the insert. We could avoid having residue in the channels by spraying the insert and then pushing it down against the petri dish, but the residue added extra background noise to the UV imaging. Next we tried rubber cement, which could be applied more consistently, but needed a thicker application layer and did not bind the insert to the petri dish strongly enough.

Finally we settled on using polydimethylsiloxane (PDMS), which is an inert, non-toxic, non-flammable organic polymer. PDMS has a wide range of applications including the creation of microfluidic devices [39], but here we care most about its viscoelastic properties and that it hardens optically clear when cured. We use the Dow Corning Sylgard 184 Silicone Elastomer Kit with the base to curing agent in a 10:1 ratio. We mix up an appropriate amount (2.2 g for a petri dish and 3.52 g for a single-well plate) and spread a thin layer across the bottom of our plate. We then leave on a flat surface to even out naturally before placing our insert on top. We place the device in a vacuum desiccator to remove trapped bubbles and then cure on a 80°C heater. The PDMS hardens and bonds the insert to the bottom of the container strongly. Because the PDMS covers the entire surface, it does occupy

a small amount of the volume within each channel.

Agar:

Lysogeny Broth (LB)-based agar is the most common growth medium for *E. coli* but has different formulations based on sodium chloride levels and suffers from variations from batch to batch. Additionally, LB agar has a slight yellowish-brown color that add background noise to our UV imaging. Instead we use an EZ Rich-based agar because it is defined and has more of a clear white color. We mix together EZ Rich defined medium (Teknova M2105) minus the glucose and add 1.5 g granulated agar (BD 214530) per 100 mL of medium. After autoclaving, we allow the molten agar to cool to below 60°C before adding glucose and antibiotic(s). These come after to prevent inactivation of the antibiotics and the browning of the glucose due to heat.

To fill the channels with agar, we pipette the appropriate amounts of molten agar directly into the channels. Pouring is not recommended as it leads to inconsistent volumes and scraping away excess can often leave a thin layer on top of the insert that connects channels or at least gives additional volume for AHL to diffuse into. The volumes of the channels can easily be calculated (e.g. C2 is 3.5 mm × 12 mm × 3.175 mm = 133.35 μL), but remember to account for the PDMS in the channel as well.

3.6.5 Robotics Protocols

We decided to use the Beckman Coulter Biomek NX^P, which actuates eight pipette tips (one column) at a time and has other nice features that help us. The setup can be seen in Figure 3.23. The following protocols are written in BioMek Software version 3.3.

Pouring Agar:

Our Biomek NX^P is equipped with a heater and a removable metal reservoir in which we can keep our agar hot to prevent the premature solidification. We clean and preheat the reservoir to 60°C and pour our molten agar into it. Typically we add at least 100 mL of agar to get a minimum height for the protocol, but the reservoir can hold up to 250 mL depending on how many plates we are filling. The channels have a volume of 144.4625 μL, but we transfer 140 μL to each channel due to the PDMS. This protocol uses wide-bore (genomic) tips.

The biggest difficulties of this protocol are to prevent cooling-related problems and to prevent bubble formation. The protocol is described below and is designed to overcome these difficulties:

1. Dip tips into agar and pipette up and down twice to warm the tips.
2. Aspirate desired volume of agar while moving tips laterally within the agar. This should prevent the uptake of any bubbles lingering on the tips.
3. Wipe tips against both sides of the reservoir to remove excess agar.

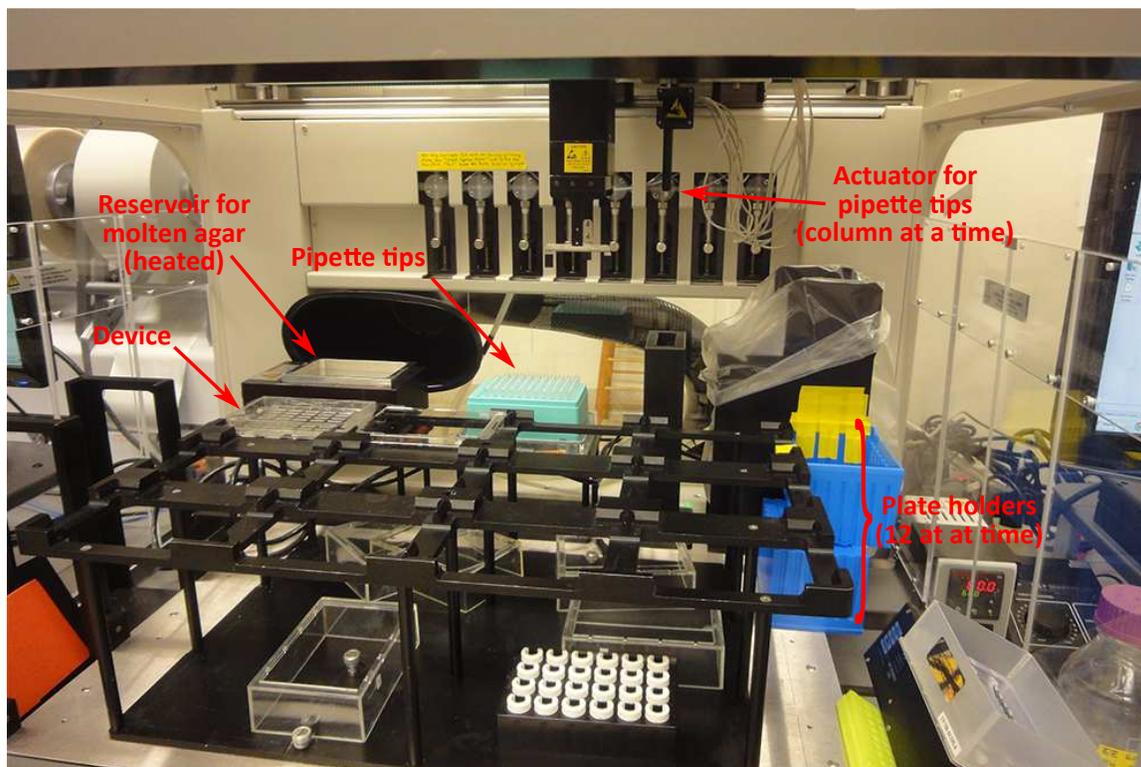


Figure 3.23: Robotics setup for channel device using a Beckman Coulter Biomek NX^P to improve experimental consistency. The single-well plates that work with Insert 2 fit into the plate holders. We developed two separate protocols: one for filling the channels with agar and the other for seeding compartments. This robot allows us to pipette one column at a time, which is well-suited for our application, and our protocols can be looped over many plates at once (up to twelve).

4. Position tips on one side of the next column of channels to fill.
5. Dispense agar as tips move to the other side of the channels.
6. Once agar is dispensed, move tips twice across the length of the channel to smooth the layer and make sure it reaches both ends.
7. Return tips to reservoir and pipette agar up and down to rewarm the tips and melt any solidified agar in or around the tips.
8. Loop back to Step 2 until desired number of plates (up to twelve) have been filled.
9. Release tips into waste bin.

Note that this protocol is run entirely with the lids off. Because the volume of agar is so small, each device solidifies within 5-10 min of dispensation and is ready to use.

Seeding Colonies:

This protocol is still under development. This protocol has to seed an entire column of compartments at a time. To change this, we would need to specially prepare our tip rack to be strategically missing tips. We want this protocol to be able to seed many different kinds of cells in one go, so we will use deep-well column reservoirs to hold our diluted liquid cultures. Depending on how many we need, there are 4-column reservoirs (Phenix Research RRI-3051) and 12-column reservoirs (E&K Scientific EK-2034). This reservoir sits on a plate holder and needs to be manually filled beforehand.

The only difficulty of this protocol is dealing with such small culture volume ($0.5 \mu\text{L}$). Beyond that this protocol is very straightforward: pick up specified culture from reservoir column, move to specified compartments, and dispense.

Chapter 4

Conclusion

The engineering of cooperative ensembles of cells, whether in the context of designer microbial communities or other synthetic multicellular systems will require tractable model systems which exhibit spontaneous symmetry breaking and pattern formation, both fundamental prerequisites for any kind of replicating or “programmed” heterogeneity of form or function. Here we examined two distinctly different methods for achieving pattern formation, diffusion-driven instability and lateral inhibition, in the hopes of better understanding their appearance in natural systems and working towards synthetic implementations of multicellular behavior. These two examples were motivated by two different cell-to-cell communication methods, namely quorum sensing and contact-based signaling.

For Turing patterning, we proposed a new system using oscillating subsystems. To our knowledge, this is the first attempt of this kind and significant effort was devoted to providing researchers with an experimentally tractable road map towards implementation. This work also implicitly suggests that natural systems may have arisen where oscillating subsystems, initially evolved for other purposes, provide the backbone not just for coordinated oscillation (as in the diffusively coupled systems demonstrated by others [33, 98, 47]) but for robust Turing-type pattern formation phenomena. These motifs are also present in protein-protein systems [73]; while outside the scope of the present work, the general results presented (i.e. coupled multi-step negative feedback oscillators with one diffusible component can exhibit Turing instability) would likely apply to kinase loops [73].

While attempting a partial implementation of our quenched oscillator system, our goal of building a new synthetic ring oscillator resulted in the creation of new synthetic inverters constructed from ZFPs and sRNAs, two synthetic components with exciting potential to create large libraries of synthetic parts. We attempted to take the parts as had been constructed and characterized by others and to combine them in a modular fashion. And while the resulting inverter performance was insufficient to build the ring oscillator, there’s good reason to believe that our goals can be achieved with redesigns of our parts or newer technologies such as CRISPRi [110].

For lateral inhibition, we developed a new theoretical framework based on graph theory for analyzing pattern formation in large networks with restricted communication channels

like in CDI. The analysis has a few limitations, particularly maintaining a stable, equitable graph throughout, but is a very elegant and computationally inexpensive way to examine the existence and stability of many different possible patterns at once. With the lack of an experimentally-tractable CDI system at our disposal, we proposed a CDI-like system that we called a compartmental lateral inhibition system. We then adjusted our analysis to accommodate the use of two different cell lines and communication channels and achieved similar analytical results.

Our compartmental lateral inhibition laboratory implementation is underway and we maintain high hopes for patterning success in the near future. The current limiting factor is the difference in signaling strengths between our chosen diffusible molecules, which wouldn't be an issue in an actual CDI system. The real excitement will come once a synthetic contact-based system is achieved and we can try to verify our analysis using the communication channel it was intended for. Beyond our patterning systems, a synthetic contact-mediated communication channel would be hugely important as the first of its kind for any number of synthetic multicellular systems.

The work presented here is just the beginning towards unlocking engineered multicellular behaviors. The new analysis techniques developed should prove to be useful on future designs and the promise of new and better parts to try always remains. The systems we came up with here were intended to be experimentally achievable and should be within the realm of possibility. We look forward to when our engineering ambition becomes realizable in biological systems.

Bibliography

- [1] Y.Y. Adar and S. Ulitzur. “GroESL Proteins Facilitate Binding of Externally Added Inducer by LuxR Protein-Containing *E. coli* cells”. In: *Journal of Bioluminescence and Chemiluminescence* 8.5 (1993), pp. 261–266. ISSN: 1099-1271.
- [2] U. Alon. *An Introduction to Systems Biology: Design Principles of Biological Circuits*. CRC Press, 2006.
- [3] J.B. Andersen et al. “New Unstable Variants of Green Fluorescent Protein for Studies of Transient Gene Expression in Bacteria”. In: *Applied and Environmental Microbiology* 64.6 (1998), pp. 2240–2246.
- [4] J.C. Anderson, C.A. Voigt, and A.P. Arkin. “Environmental signal integration by a modular AND gate”. In: *Molecular Systems Biology* 3.1 (2007), p. 133.
- [5] D. Angeli and E.D. Sontag. “Monotone Control Systems”. In: *Automatic Control, IEEE Transactions on* 48.10 (2003), pp. 1684–1698.
- [6] S.K. Aoki et al. “A widespread family of polymorphic contact-dependent toxin delivery systems in bacteria”. In: *Nature* 468.7322 (2010), pp. 439–442.
- [7] S.K. Aoki et al. “Contact-Dependent Growth Inhibition Causes Reversible Metabolic Downregulation in *Escherichia coli*”. In: *Journal of Bacteriology* 191.6 (2009), pp. 1777–1786.
- [8] S.K. Aoki et al. “Contact-dependent growth inhibition requires the essential outer membrane protein BamA (YaeT) as the receptor and the inner membrane transport protein AcrB”. In: *Molecular Microbiology* 70.2 (2008), pp. 323–340.
- [9] M. Arcak. “Pattern Formation by Lateral Inhibition in Large-Scale Networks of Cells”. In: *Automatic Control, IEEE Transactions on* 58.5 (2013), pp. 1250–1262.
- [10] A.P. Arkin. “Synthetic cell biology”. In: *Current Opinion in Biotechnology* 12.6 (2001), pp. 638–644.
- [11] F.K. Balagaddé et al. “A synthetic *Escherichia coli* predator–prey ecosystem”. In: *Molecular Systems Biology* 4.1 (2008), p. 187.
- [12] S. Basu et al. “A synthetic multicellular system for programmed pattern formation”. In: *Nature* 434.7037 (2005), pp. 1130–1134.

- [13] S. Basu et al. “Spatiotemporal control of gene expression with pulse-generating networks”. In: *Proceedings of the National Academy of Sciences* 101.17 (2004), pp. 6355–6360.
- [14] J. Beal and J. Bachrach. “Infrastructure for Engineered Emergence on Sensor/Actuator Networks”. In: *Intelligent Systems, IEEE* 21.2 (2006), pp. 10–19.
- [15] A. Berman and R.J. Plemmons. “Nonnegative Matrices in the Mathematical Sciences”. In: *Classics in Applied Mathematics* 9 (1979).
- [16] L. Bintu et al. “Transcriptional regulation by the numbers: applications”. In: *Current Opinion in Genetics & Development* 15.2 (2005), pp. 125–135. ISSN: 0959-437X.
- [17] E.G. Biondi et al. “Regulation of the bacterial cell cycle by an integrated genetic circuit”. In: *Nature* 444.7121 (2006), pp. 899–904. ISSN: 0028-0836.
- [18] J. Boch. “TALEs of genome targeting”. In: *Nature Biotechnology* 29.2 (2011), pp. 135–136.
- [19] A. Bonyár et al. “3D Rapid Prototyping Technology (RPT) as a powerful tool in microfluidic development”. In: *Procedia Engineering* 5 (2010), pp. 291–294.
- [20] M. Boyer and F. Wisniewski-Dyé. “Cell–cell signalling in bacteria: not simply a matter of quorum”. In: *FEMS Microbiology Ecology* 70.1 (2009), pp. 1–19.
- [21] J.A.N. Brophy and C.A. Voigt. “Principles of genetic circuit design”. In: *Nature Methods* 11.5 (2014), pp. 508–520.
- [22] N.E. Buchler and F.R. Cross. “Protein sequestration generates a flexible ultrasensitive response in a genetic network”. In: *Molecular Systems Biology* 5.1 (2009).
- [23] R.E. Campbell et al. “A monomeric red fluorescent protein”. In: *Proceedings of the National Academy of Sciences* 99.12 (2002), pp. 7877–7882.
- [24] B. Canton, A. Labno, and D. Endy. “Refinement and standardization of synthetic biological parts and devices”. In: *Nature Biotechnology* 26.7 (2008), pp. 787–793. ISSN: 1087-0156.
- [25] R.G. Casten and C.J. Holland. “Stability Properties of Solutions to Systems of Reaction-Diffusion Equations”. In: *SIAM Journal on Applied Mathematics* (1977), pp. 353–364.
- [26] D. Chen. “Designing Genetic Circuits for Memory and Communication”. PhD thesis. Berkeley: Univ. of California, Berkeley, Dec. 2014.
- [27] D.J. Cohen, R.C. Morfino, and M.M. Maharbiz. “A Modified Consumer Inkjet for Spatiotemporal Control of Gene Expression”. In: *PLoS ONE* 4 (2009), e7086. DOI: 10.1371/journal.pone.0007086.
- [28] J.R. Collier et al. “Pattern Formation by Lateral Inhibition with Feedback: a Mathematical Model of Delta-Notch Intercellular Signalling”. In: *Journal of Theoretical Biology* 183.4 (1996), pp. 429–446.

- [29] C.H. Collins, F.H. Arnold, and J.R. Leadbetter. “Directed evolution of *Vibrio fischeri* LuxR for increased sensitivity to a broad spectrum of acyl-homoserine lactones”. In: *Molecular Microbiology* 55.3 (2005), pp. 712–723.
- [30] C.H. Collins, J.R. Leadbetter, and F.H. Arnold. “Dual selection enhances the signaling specificity of a variant of the quorum-sensing transcriptional activator LuxR”. In: *Nature Biotechnology* 24.6 (2006), pp. 708–712.
- [31] B. Conway and E.P. Greenberg. “Quorum-Sensing Signals and Quorum-Sensing Genes in *Burkholderia vietnamiensis*”. In: *Journal of Bacteriology* 184.4 (2002), pp. 1187–1191.
- [32] F.R. Cross. “Two Redundant Oscillatory Mechanisms in the Yeast Cell Cycle”. In: *Cell* 4.5 (2003), pp. 741–752. ISSN: 1534-5807.
- [33] T. Danino et al. “A synchronized quorum of genetic clocks”. In: *Nature* 463.7279 (2010), pp. 326–330. ISSN: 0028-0836.
- [34] D. Del Vecchio. “Modularity, context-dependence, and insulation in engineered biological circuits”. In: *Trends in Biotechnology* 33.2 (2015), pp. 111–119.
- [35] C.A. Desoer and M. Vidyasagar. *Feedback Systems: Input-Output Properties*. Vol. 55. SIAM, 1975.
- [36] R. Dillon, P.K. Maini, and H.G. Othmer. “Pattern formation in generalized Turing systems”. In: *Journal of Mathematical Biology* 32.4 (1994), pp. 345–393.
- [37] I.B. Dodd et al. “Cooperativity in long-range gene regulation by the λ CI repressor”. In: *Genes & Development* 18.3 (2004), pp. 344–354. ISSN: 0890-9369.
- [38] K. Dreij et al. “A Method for Efficient Calculation of Diffusion and Reactions of Lipophilic Compounds in Complex Cell Geometry”. In: *PLoS ONE* 6.8 (Aug. 2011), e23128. DOI: 10.1371/journal.pone.0023128.
- [39] D.C. Duffy et al. “Rapid Prototyping of Microfluidic Systems in Poly(dimethylsiloxane)”. In: *Analytical Chemistry* 70.23 (1998), pp. 4974–4984.
- [40] K.A. Egland and E.P. Greenberg. “Conversion of the *Vibrio fischeri* Transcriptional Activator, LuxR, to a Repressor”. In: *Journal of Bacteriology* 182.3 (2000), pp. 805–811.
- [41] J.W. Ellefson et al. “Directed evolution of genetic parts and circuits by compartmentalized partnered replication”. In: *Nature Biotechnology* 32.1 (2014), pp. 97–101.
- [42] M.B. Elowitz and S. Leibler. “A synthetic oscillatory network of transcriptional regulators”. In: *Nature* 403.6767 (2000), pp. 335–338.
- [43] D. Endy. “Foundations for engineering biology”. In: *Nature* 438.7067 (2005), pp. 449–453.
- [44] J. Engebrecht and M. Silverman. “Identification of genes and gene products necessary for bacterial bioluminescence”. In: *Proceedings of the National Academy of Sciences* 81.13 (1984), pp. 4154–4158.

- [45] M. Englmann et al. “The hydrolysis of unsubstituted *N*-acylhomoserine lactones to their homoserine metabolites: Analytical approaches using ultra performance liquid chromatography”. In: *Journal of Chromatography A* 1160.1-2 (2007), pp. 184–193. ISSN: 0021-9673.
- [46] C. Fuqua, S.C. Winans, and E.P. Greenberg. “Census and Consensus in Bacterial Ecosystems: the LuxR-LuxI Family of Quorum-Sensing Transcriptional Regulators”. In: *Annual Reviews in Microbiology* 50.1 (1996), pp. 727–751.
- [47] J. Garcia-Ojalvo, M.B. Elowitz, and S.H. Strogatz. “Modeling a synthetic multicellular clock: Repressilators coupled by quorum sensing”. In: *Proceedings of the National Academy of Sciences* 101.30 (2004), pp. 10955–10960.
- [48] T.S. Gardner, C.R. Cantor, and J.J. Collins. “Construction of a genetic toggle switch in *Escherichia coli*”. In: *Nature* 403.6767 (2000), pp. 339–342.
- [49] X. Ge and M. Arcak. “A sufficient condition for additive *D*-stability and application to reaction-diffusion models”. In: *Systems & Control Letters* 58.10-11 (2009), pp. 736–741.
- [50] C. Godsil and G.F. Royle. *Algebraic Graph Theory*. Vol. 207. Springer Science & Business Media, 2013.
- [51] H.P.W. Gottlieb. “Eigenvalues of the Laplacian with Neumann boundary conditions”. In: *The Journal of the Australian Mathematical Society. Series B. Applied Mathematics* 26 (1985), pp. 293–309.
- [52] S.G. Grant et al. “Differential plasmid rescue from transgenic mouse DNAs into *Escherichia coli* methylation-restriction mutants”. In: *Proceedings of the National Academy of Sciences* 87.12 (1990), pp. 4645–4649.
- [53] K.M. Gray et al. “Interchangeability and Specificity of Components from the Quorum-Sensing Regulatory Systems of *Vibrio fischeri* and *Pseudomonas aeruginosa*”. In: *Journal of Bacteriology* 176.10 (1994), pp. 3076–3080.
- [54] E. Gur and R.T. Sauer. “Evolution of the *ssrA* degradation tag in *Mycoplasma*: Specificity switch to a different protease”. In: *Proceedings of the National Academy of Sciences* 105.42 (2008), pp. 16113–16118.
- [55] R. Haberman. *Elementary Applied Partial Differential Equations: with Fourier series and boundary value problems*. 3rd. New Jersey: Prentice Hall, 1998.
- [56] D.K. Hawley, A.D. Johnson, and W.R. McClure. “Functional and Physical Characterization of Transcription Initiation Complexes in the Bacteriophage λO_R Region”. In: *The Journal of Biological Chemistry* 260.14 (1985), pp. 8618–8626. ISSN: 0021-9258.
- [57] C.S. Hayes, S.K. Aoki, and D.A. Low. “Bacterial Contact-Dependent Delivery Systems”. In: *Annual Review of Genetics* 44 (2010), pp. 71–90.
- [58] A. Hemsley et al. “A simple method for site-directed mutagenesis using the polymerase chain reaction”. In: *Nucleic Acids Research* 17.16 (1989), pp. 6545–6551.

- [59] C. Herman et al. “Degradation of carboxy-terminal-tagged cytoplasmic proteins by the *Escherichia coli* protease HflB (FtsH)”. In: *Genes & Development* 12.9 (1998), pp. 1348–1355.
- [60] N.J. Hillson, R.D. Rosengarten, and J.D. Keasling. “j5 DNA Assembly Design Automation Software”. In: *ACS Synthetic Biology* 1.1 (2011), pp. 14–21.
- [61] W.J. Holtz. “Engineering Scalable Combinational Logic in *Escherichia coli* Using Zinc Finger Proteins”. PhD thesis. Berkeley: Univ. of California, Berkeley, May 2011.
- [62] J. Hsia et al. “A Feedback Quenched Oscillator Produces Turing Patterning with One Diffuser”. In: *PLoS Computational Biology* 8.1 (2012), e1002331–e1002331.
- [63] J. Hsia et al. “A Quenched Oscillator Network for Pattern Formation in Gene Expression”. In: *American Control Conference (ACC), 2011*. IEEE, 2011, pp. 2284–2289.
- [64] D. Huang, W.J. Holtz, and M.M. Maharbiz. “A genetic bistable switch utilizing non-linear protein degradation”. In: *Journal of Biological Engineering* 6.9 (2012).
- [65] I. Hwang et al. “TraI, a LuxI homologue, is responsible for production of conjugation factor, the Ti plasmid *N*-acylhomoserine lactone autoinducer”. In: *Proceedings of the National Academy of Sciences* 91.11 (1994), pp. 4639–4643.
- [66] E. Jabri. “Tag, you’re degraded”. In: *Nature Structural & Molecular Biology* 10.9 (2003), p. 676.
- [67] S. Jang et al. “Specification and Simulation of Multicelled Behaviors”. In: *ACS Synthetic Biology* 22.24 (2012), pp. 3067–3074.
- [68] Y. Jia and S.S. Patel. “Kinetic Mechanism of Transcription Initiation by Bacteriophage T7 RNA Polymerase”. In: *Biochemistry* 36.14 (1997), pp. 4223–4232. ISSN: 0006-2960.
- [69] M. Jinek et al. “A Programmable Dual-RNA–Guided DNA Endonuclease in Adaptive Bacterial Immunity”. In: *Science* 337.6096 (2012), pp. 816–821.
- [70] E. Kaszkurewicz and A. Bhaya. *Matrix Diagonal Stability in Systems and Computation*. Birkhauser, 2000.
- [71] A.S. Khalil and J.J. Collins. “Synthetic biology: applications come of age”. In: *Nature Reviews Genetics* 11.5 (2010), pp. 367–379.
- [72] A. Khlebnikov, T. Skaug, and J.D. Keasling. “Modulation of gene expression from the arabinose-inducible *araBAD* promoter”. In: *Journal of Industrial Microbiology & Biotechnology* 29.1 (2002), pp. 34–37.
- [73] B.N. Kholodenko. “Cell-signalling dynamics in time and space”. In: *Nature Reviews Molecular Cell Biology* 7.3 (2006), pp. 165–176. ISSN: 1471-0072.
- [74] A.M. Kierzek, J. Zaim, and P. Zielenkiewicz. “The Effect of Transcription and Translation Initiation Frequencies on the Stochastic Fluctuations in Prokaryotic Gene Expression”. In: *The Journal of Biological Chemistry* 276.11 (2001), pp. 8165–8172. ISSN: 0021-9258.

- [75] J. Kim and E. Winfree. “Synthetic *in vitro* transcriptional oscillators”. In: *Molecular Systems Biology* 7.1 (2011), p. 465.
- [76] H. Kitano. “Biological robustness”. In: *Nature Reviews Genetics* 5.11 (2004), pp. 826–837.
- [77] L. Kizer et al. “Application of Functional Genomics to Pathway Optimization for Increased Isoprenoid Production”. In: *Applied and Environmental Microbiology* 74.10 (2008), pp. 3229–3241.
- [78] A. Klug. “The Discovery of Zinc Fingers and Their Applications in Gene Regulation and Genome Manipulation”. In: *Annual Review of Biochemistry* 79 (2010), pp. 213–231.
- [79] P. Kokotović, H.K. Khalil, and J. O’Reilly. *Singular Perturbation Methods in Control: Analysis and Design*. Academic Press, 1986.
- [80] S. Kondo and T. Miura. “Reaction-Diffusion Model as a Framework for Understanding Biological Pattern Formation”. In: *Science* 329.5999 (2010), p. 1616.
- [81] H.E. Kubitschek and J.A. Friske. “Determination of Bacterial Cell Volume with the Coulter Counter”. In: *Journal of Bacteriology* 168.3 (1986), pp. 1466–1467. ISSN: 0021-9193.
- [82] T. Lederer et al. “Tetracycline Analogs Affecting Binding to Tn10-Encoded Tet Repressor Trigger the Same Mechanism of Induction”. In: *Biochemistry* 35.23 (1996), pp. 7439–7446.
- [83] T.S. Lee et al. “BglBrick vectors and datasheets: A synthetic biology platform for gene expression”. In: *Journal of Biological Engineering* 5.1 (2011), pp. 1–14.
- [84] M. Lies and M.R. Maurizi. “Turnover of Endogenous SsrA-tagged Proteins Mediated by ATP-dependent Proteases in *Escherichia coli*”. In: *The Journal of Biological Chemistry* 283.34 (2008), pp. 22918–22929. ISSN: 0021-9258.
- [85] M. Lis et al. “Efficient stochastic simulation of reaction-diffusion processes via direct compilation”. In: *Bioinformatics* 25.17 (2009), pp. 2289–2291.
- [86] R.T. Liu, S.S. Liaw, and P.K. Maini. “Oscillatory Turing Patterns in a Simple Reaction-Diffusion System”. In: *Journal of the Korean Physical Society* 50.1 (2007), pp. 234–238.
- [87] J.C.W. Locke et al. “Experimental validation of a predicted feedback loop in the multi-oscillator clock of *Arabidopsis thaliana*”. In: *Molecular Systems Biology* 2.1 (2006). DOI: 10.1038/msb4100102.
- [88] E.M. Lucchetta et al. “Dynamics of *Drosophila* embryonic patterning network perturbed in space and time using microfluidics”. In: *Nature* 434.7037 (2005), pp. 1134–1138.

- [89] J.B. Lucks et al. “Versatile RNA-sensing transcriptional regulators for engineering genetic networks”. In: *Proceedings of the National Academy of Sciences* 108.21 (2011), p. 8617.
- [90] R. Lutz and H. Bujard. “Independent and tight regulation of transcriptional units in *Escherichia coli* via the LacR/O, the TetR/O and AraC/I₂-I₂ regulatory elements”. In: *Nucleic Acids Research* 25.6 (1997), p. 1203. ISSN: 0305-1048.
- [91] M.M. Maharbiz. “Synthetic multicellularity”. In: *Trends in Cell Biology* 22.12 (2012), pp. 617–623.
- [92] I. Meier, L.V. Wray, and W. Hillen. “Differential regulation of the Tn10-encoded tetracycline resistance genes *tetA* and *tetR* by the tandem *tet* operators O₁ and O₂”. In: *The EMBO journal* 7.2 (1988), pp. 567–572.
- [93] H. Meinhardt. *Models of Biological Pattern Formation*. Vol. 6. London: Academic Press, 1982.
- [94] H. Meinhardt and A. Gierer. “Pattern formation by local self-activation and lateral inhibition”. In: *BioEssays* 22.8 (2000), pp. 753–760.
- [95] C.B. Michalowski and J.W. Little. “Positive Autoregulation of *cI* Is a Dispensable Feature of the Phage λ Gene Regulatory Circuitry”. In: *Journal of Bacteriology* 187.18 (2005), pp. 6430–6442.
- [96] S. Mukherji and A. van Oudenaarden. “Synthetic biology: understanding biological design from synthetic circuits”. In: *Nature Reviews Genetics* 10.12 (2009), pp. 859–871.
- [97] J.D. Murray. *Mathematical Biology I: An Introduction, vol. 17 of Interdisciplinary Applied Mathematics*. 2002.
- [98] J.D. Murray. *Mathematical Biology II: Spatial Models and Biomedical Applications*. 3rd. New York: Springer, 2003.
- [99] M.A.T. Muskavitch. “Delta-Notch Signaling and *Drosophila* Cell Fate Choice”. In: *Developmental Biology* 166.2 (1994), pp. 415–430.
- [100] V.K. Mutalik et al. “Precise and reliable gene expression via standard transcription and translation initiation elements”. In: *Nature Methods* 10.4 (2013), pp. 354–360.
- [101] V.K. Mutalik et al. “Rationally designed families of orthogonal RNA regulators of translation”. In: *Nature Chemical Biology* 8.5 (2012), pp. 447–454.
- [102] D. Na et al. “Metabolic engineering of *Escherichia coli* using synthetic small regulatory RNAs”. In: *Nature Biotechnology* 31.2 (2013), pp. 170–174.
- [103] A.V. Oppenheim and R.W. Schafer. *Discrete-Time Signal Processing*. 3rd. New Jersey: Prentice Hall, 2009.

- [104] H.G. Othmer. “Synchronized and Differentiated Modes of Cellular Dynamics”. English. In: *Dynamics of Synergetic Systems*. Ed. by Hermann Haken. Vol. 6. Springer Series in Synergetics. Springer Berlin Heidelberg, 1980, pp. 191–204. ISBN: 978-3-642-67594-2. DOI: 10.1007/978-3-642-67592-8_16.
- [105] M.R. Parsek, A.L. Schaefer, and E.P. Greenberg. “Analysis of random and site-directed mutations in *rhlI*, a *Pseudomonas aeruginosa* gene encoding an acylhomoserine lactone synthase”. In: *Molecular Microbiology* 26.2 (1997), pp. 301–310.
- [106] M.R. Parsek et al. “Acyl homoserine-lactone quorum-sensing signal generation”. In: *Proceedings of the National Academy of Sciences* 96.8 (1999), p. 4360.
- [107] J.P. Pearson, C. Van Delden, and B.H. Iglewski. “Active Efflux and Diffusion Are Involved in Transport of *Pseudomonas aeruginosa* Cell-to-Cell Signals”. In: *Journal of Bacteriology* 181.4 (1999), pp. 1203–1210.
- [108] J.P. Pearson et al. “Structure of the autoinducer required for expression of *Pseudomonas aeruginosa* virulence genes”. In: *Proceedings of the National Academy of Sciences* 91.1 (1994), pp. 197–201.
- [109] K. Potrykus, G. Wegrzyn, and V.J. Hernandez. “Multiple Mechanisms of Transcription Inhibition by ppGpp at the λ_{pR} Promoter”. In: *The Journal of Biological Chemistry* 277.46 (2002), pp. 43785–43791. ISSN: 0021-9258.
- [110] L.S. Qi et al. “Repurposing CRISPR as an RNA-Guided Platform for Sequence-Specific Control of Gene Expression”. In: *Cell* 152.5 (2013), pp. 1173–1183.
- [111] J. Quan and J. Tian. “Circular Polymerase Extension Cloning of Complex Gene Libraries and Pathways”. In: *PLoS ONE* 4.7 (2009), e6441.
- [112] R. Rauhut and G. Klug. “mRNA degradation in bacteria”. In: *FEMS Microbiology Reviews* 23.3 (1999), pp. 353–370. ISSN: 1574-6976.
- [113] J.R. Regalbutto. “Cellulosic Biofuels—Got Gasoline?” In: *Science* 325.5942 (2009), pp. 822–824.
- [114] *Registry of Standard Biological Parts*. 2009. URL: <http://partsregistry.org>.
- [115] A.D. Riggs, H. Suzuki, and S. Bourgeois. “*lac* Repressor-Operator Interaction: I. Equilibrium Studies”. In: *Journal of Molecular Biology* 48.1 (1970), pp. 67–83. ISSN: 0022-2836.
- [116] A.S. Rufino Ferreira and M. Arcaç. “A Graph Partitioning Approach to Predicting Patterns in Lateral Inhibition Systems”. In: *SIAM Journal on Applied Dynamical Systems* 12.4 (2013), pp. 2012–2031.
- [117] A.S. Rufino Ferreira et al. “Pattern Formation with a Compartmental Lateral Inhibition System”. In: *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*. IEEE. 2014, pp. 5413–5418.
- [118] Z.C. Ruhe et al. “Receptor Polymorphism Restricts Contact-Dependent Growth Inhibition to Members of the Same Species”. In: *mBio* 4.4 (2013), e00480–13.

- [119] A.L. Schaefer et al. “Generation of cell-to-cell signals in quorum sensing: Acyl homoserine lactone synthase activity of a purified *Vibrio fischeri* LuxI protein”. In: *Proceedings of the National Academy of Sciences* 93.18 (1996), pp. 9505–9509.
- [120] N.C. Shaner et al. “Improved monomeric red, orange and yellow fluorescent proteins derived from *Discosoma* sp. red fluorescent protein”. In: *Nature Biotechnology* 22.12 (2004), pp. 1567–1572.
- [121] H.L. Smith. *Monotone Dynamical Systems: An Introduction to the Theory of Competitive and Cooperative Systems*. Mathematical Surveys and Monographs. American Mathematical Society, 2008. ISBN: 9780821844878.
- [122] T. Sohka, R.A. Heins, and M. Ostermeier. “Morphogen-defined patterning of *Escherichia coli* enabled by an externally tunable band-pass filter”. In: *Journal of Biological Engineering* 3.1 (2009), p. 10.
- [123] E.D. Sontag. “Contractive Systems with Inputs”. English. In: *Perspectives in Mathematical System Theory, Control, and Signal Processing*. Vol. 398. Lecture Notes in Control and Information Sciences. Springer Berlin Heidelberg, 2010, pp. 217–228. ISBN: 978-3-540-93917-7. DOI: 10.1007/978-3-540-93918-4_20.
- [124] D. Sprinzak et al. “Mutual Inactivation of Notch Receptors and Ligands Facilitates Developmental Patterning”. In: *PLoS Computational Biology* 7.6 (2011), e1002069.
- [125] P.S. Stewart. “Diffusion in Biofilms”. In: *Journal of Bacteriology* 185.5 (2003), pp. 1485–1491.
- [126] J. Stricker et al. “A fast, robust and tunable synthetic gene oscillator”. In: *Nature* 456.7221 (2008), pp. 516–519.
- [127] D.E. Strier and S.P. Dawson. “Role of complexing agents in the appearance of Turing patterns”. In: *Physical Review E* 69.6 (2004), p. 66207.
- [128] A. Tamsir, J.J. Tabor, and C.A. Voigt. “Robust multicellular computing using genetically encoded NOR gates and chemical ‘wires’”. In: *Nature* 469.7329 (2011), pp. 212–215.
- [129] A.M. Turing. “The Chemical Basis of Morphogenesis”. In: *Philosophical Transactions of the Royal Society of London, Series B* 237 (1952), pp. 37–72.
- [130] H.H. Tuson and D.B. Weibel. “Bacteria–surface interactions”. In: *Soft Matter* 9.17 (2013), pp. 4368–4380.
- [131] A.A. Twite et al. “Direct Attachment of Microbial Organisms to Material Surfaces Through Sequence-Specific DNA Hybridization”. In: *Advanced Materials* 24.18 (2012), pp. 2380–2385.
- [132] B. Wang et al. “Engineering modular and orthogonal genetic logic gates for robust digital-like synthetic biology”. In: *Nature Communications* 2 (2011), p. 508.

- [133] C.M. Waters and B.L. Bassler. “Quorum Sensing: Cell-to-Cell Communication in Bacteria”. In: *Annual Review of Cell and Developmental Biology* 21 (2005), pp. 319–346.
- [134] J.S. Webb et al. “Delivery of CdiA Nuclease Toxins into Target Cells during Contact-Dependent Growth Inhibition”. In: *PLoS ONE* 8.2 (2013), e57609.
- [135] W. Weber and M. Fussenegger. “Emerging biomedical applications of synthetic biology”. In: *Nature Reviews Genetics* 13.1 (2012), pp. 21–35.
- [136] A.T. Winfree. *The Geometry of Biological Time*. Interdisciplinary Applied Mathematics. Springer New York, 2001. ISBN: 9780387989921.
- [137] H.X. Zhou. “Enhancement of Protein-Protein Association Rate by Interaction Potential: Accuracy of Prediction Based on Local Boltzmann Factor”. In: *Biophysical Journal* 73.5 (1997), pp. 2441–2445. ISSN: 0006-3495.

Appendix A

Sample Code

A.1 Analysis Code - Quenched Oscillator

```

1 % mod_qosc.m
2 %%% get default parameter values
3 vals = qosc(0, 'load', 1);
4 extract_qosc;
5
6 aC = nthroot(sigL+1, 2*nC);
7 aT = nthroot(sigC+1, 2*nT);
8 aL = nthroot(sigT+1, 2*nL);
9 aA = 1/(pR/KRA*nthroot(sigRA+1, 2*nRA)-1);
10 aR = 1;
11 %aR = 0.9705;
12 %aR = 0.75;
13
14 epC = gC*gmO*KC/(VC*NC) * aC / (1/(1+aT^nT) + 1/sigC);
15 epTO = gT*gmO*KT/(VT*NT) * aT / (1/(1+aL^nL) + 1/sigT) * aR/(1+aR);
16 epTQ = gT*gmQ*KT/(VRA*NRA) * aT / (1/(1+(KRA/pR*(1+aA)/aA)^nRA) ...
17     + 1/sigRA) * 1/(1+aR);
18 epL = gL*gmO*KL/(VL*NL) * aL / (1/(1+aC^nC) + 1/sigL);
19 epI = gI*gA*gmQ*(kr/kf)/(VI*NI*v3) * aA / (1/(1+aT^nT) + 1/sigI);
20
21 XC = (VC*NC* epC/KC/gC)*nC*aC^(nC-1)/(1 + aC^nC)^2;
22 XT = (VT*NT*epTO/KT/gT)*nT*aT^(nT-1)/(1 + aT^nT)^2;
23 XL = (VL*NL* epL/KL/gL)*nL*aL^(nL-1)/(1 + aL^nL)^2;
24 X = -nthroot(XC*XT*XL, 3)/gmO;
25 fX = 3*X^2/(4+2*X);
26
27
28 vals = qosc(disp, 'epC', epC, 'epTO', epTO, 'epTQ', epTQ, 'epL', epL, 'epI', epI);
29 extract_qosc;
30
31 temp = max(real(eigA));
32 th = 1/temp * log(0.5);

```

```

33
34 temp = max(real(eigAo));
35 td = 1/temp * log(2);
36
37 D = zeros(11);
38 for d = 0:1e-5:1
39     D(9,9) = d;
40     if(max(real(eig(A-D))) >= 0)
41         break;
42     end
43 end
44 omega = 2*pi/sqrt(d/Dahl);
45
46 fprintf(1, 'half-life: %1.5g\hours\n', th/3600);
47 fprintf(1, 'doubling time: %1.5g\hours\n', td/3600);
48 fprintf(1, 'max unstable wavelength: %1.5g\n\n', omega);

```

A.2 Analysis Code - Quenched Oscillator Function

```

1 % function vals = qosc(dispatch, varargin)
2 %
3 % Solve quenched oscillator for steady-state and Jacobian
4 % Also, truncated parameter values used.
5 %
6 % Inputs:
7 %     disp      - whether to display values (1) or not (0) in command window
8 %     varargin  - optional inputs used to override hard-coded parameter
9 %                 values. must be in parameter name & value pairs.
10 %                 Changeable parameters are: gI, gA, epI, KRA, Kf, Kr, sigI,
11 %                 sigRA, VI, VRA, nRA, pR, and Dahl
12 %                 Special option 'load' loads parameter set from external file
13 %
14 %     Examples:  vals = qosc(1);
15 %                 vals = qosc(1, 'KRA', 4e-8, 'nRA', 3);
16 %                 vals = qosc(1, 'load', '12-08-d');
17 %
18 % written by Justin Hsia
19 function vals = qosc(dispatch, varargin)
20     %% input checking (varargin is horizontal cell array)
21     narg = size(varargin, 2);
22     if (mod(narg, 2) ~= 0)
23         fprintf(1, 'Incorrect number of arguments.\n')
24         vals = -1;
25         return
26     end
27     for i = 1:narg/2
28         if ( ~strcmp(varargin{2*i-1}, 'load') && ( ~ischar(varargin{2*i-1}) ...
29             || ~isnumeric(varargin{2*i}) ) )
30             fprintf(1, 'Incorrect argument pair format: %s name', value\n');
31             vals = -2;

```

```

32         return
33     end
34 end
35
36 C = 1.5e-9;
37 %% Degradation Rates (1/s)
38 gC = 2.89e-3; %gC = log(2)/60/4;
39 gT = gC;
40 gL = gC;
41 gI = 2.89e-2; %gI = log(2)/60/0.2;
42 gA = 2.89e-2; %gA = log(2)/60/0.2;
43 gmO = 5.78e-3; %gmO = log(2)/60/2;
44 gmQ = 5.78e-2; %gmQ = log(2)/60/0.2;
45 %% Epsilons
46 epC = 3.7122e-4;
47 epTO = 1.8561e-4;
48 epTQ = 1.8561e-3;
49 epL = 3.7122e-4;
50 epI = 3.8149e-2;
51 %% Dissociation Constants (M)
52 KC = 2*C;
53 KT = 2*C;
54 KL = 2*C;
55 KRA = 45*C;
56 kf = 7.5/C;
57 kr = 15;
58 %% Leakage Ratios (no units)
59 sigC = 1000;
60 sigT = 1000;
61 sigL = 1000;
62 sigI = sigC;
63 sigRA = 1000;
64 %% Velocity Constants (1/s)
65 VC = 4;
66 VT = 4;
67 VL = 4;
68 VI = VC;
69 VRA = 4;
70 %% Copy Numbers (range from 1.5 to 60)
71 NC = 4;
72 NT = 4;
73 NL = 4;
74 NI = 4;
75 NRA = 4;
76 %% Hill Coefficients (no units)
77 nC = 2;
78 nT = 2;
79 nL = 2;
80 nRA = 2;
81 %% other parameters

```

```

82     v3     = 0.01; % 1/s
83     pR     = 12*C;
84     Dahl   = 1.667e-11; % m^2/s
85     L      = 1e-6;      % m
86
87
88     %%% override values with variable arguments
89     for i = 1:narg/2
90         switch varargin{2*i-1}
91             case 'gmO'
92                 gmO = gmO/varargin{2*i};
93             case 'gmQ'
94                 gmQ = gmQ/varargin{2*i};
95             case 'gI'
96                 gI  = varargin{2*i};
97             case 'gA'
98                 gA  = varargin{2*i};
99             case 'epTO'
100                epTO = varargin{2*i};
101             case 'epTQ'
102                epTQ = varargin{2*i};
103             case 'epC'
104                epC  = varargin{2*i};
105             case 'epL'
106                epL  = varargin{2*i};
107             case 'epI'
108                epI  = varargin{2*i};
109             case 'kf'
110                kf   = varargin{2*i};
111             case 'kr'
112                kr   = varargin{2*i};
113             case 'sigRA'
114                sigRA = varargin{2*i};
115             case 'NRA'
116                NRA  = varargin{2*i};
117             case 'nRA'
118                nRA  = varargin{2*i};
119             case 'pR'
120                pR   = varargin{2*i};
121             case 'D'
122                Dahl = varargin{2*i};
123             case 'load'
124                d = 1; scale = 1;
125                switch(varargin{2*i})
126                    case 1
127                        params1
128                    case 2
129                        params2
130                end
131             otherwise

```



```

182 BC = (epC/gC/gmO) * (VC*NC*C/KC);
183 BT = (epTO/gT/gmO) * (VT*NT*C/KT);
184 BL = (epL/gL/gmO) * (VL*NL*C/KL);
185 BI = (v3/gA)*(epI/gI/gmQ) * (VI*NI*C/(kr/kf));
186 BR = (epTQ/gT/gmQ) * (VRA*NRA*C/KT);
187 vals.Bs = [BC BT BL BI BR];
188
189 %%% solve for alphas numerically
190 init = 10;
191 aT = fzero(@(aT) findAlphaT(aT,vals),init);
192 aC = BC * (1/(1+aT^nT) + 1/sigC);
193 aA = BI * (1/(1+aT^nT) + 1/sigI);
194 aL = BL * (1/(1+aC^nC) + 1/sigL);
195 aR = aT/BR / (1/(1+(KRA/pR*(1+aA)/aA)^nRA) + 1/sigRA) - 1;
196 vals.alphas = [aC aT aL aA aR];
197
198 pC = aC*KC; pT = aT*KT; pL = aL*KL; AHL = aA*(kr/kf);
199 pI = gA*AHL/v3; pRA = pR * aA/(1 + aA);
200 mTO = VT*NT*C * (1/(1+aL^nL) + 1/sigT) / gmO;
201 mTQ = VRA*NRA*C * (1/(1 + (KRA/pR*(1+aA)/aA)^nRA) + 1/sigRA) / gmQ;
202 mC = gC/epC * pC; mL = gL/epL * pL; mI = gI/epI * pI;
203 vals.ps = [pC pT pL pI AHL pRA];
204 vals.ms = [mC mTO mTQ mL mI];
205
206 %%% build linearized matrix
207 c1 = epC; c3 = epTO; c5 = epL;
208 c7 = epI; c9 = kf*pR/(1 + aA); c11 = epTQ;
209 a1 = gmO; a3 = gmO; a5 = gmO;
210 a2 = gC; a4 = gT; a6 = gL;
211 a7 = gmQ; a8 = gI; a9 = c9+gA; a10 = kr*(1 + aA); a11 = gmQ;
212
213 b2 = (VL*NL*C/KC)*nC*aC^(nC-1)/(1 + aC^nC)^2;
214 b4 = (VC*NC*C/KT)*nT*aT^(nT-1)/(1 + aT^nT)^2;
215 b42 = (VI*NI*C/KT)*nT*aT^(nT-1)/(1 + aT^nT)^2;
216 b6 = (VT*NT*C/KL)*nL*aL^(nL-1)/(1 + aL^nL)^2;
217 b10 = (VRA*NRA*C/pR)*nRA*(KRA/pR*(1+aA)/aA)^nRA/ ...
218 (1 + (KRA/pR*(1+aA)/aA)^nRA)^2 * (1+aA)/aA;
219
220 Ao = [-a1 0 0 -b4 0 0; ...
221 c1 -a2 0 0 0 0; ...
222 0 0 -a3 0 0 -b6; ...
223 0 0 c3 -a4 0 0; ...
224 0 -b2 0 0 -a5 0; ...
225 0 0 0 0 c5 -a6];
226 Au = [0 0 0 0 0; ...
227 0 0 0 0 0; ...
228 0 0 0 0 0; ...
229 0 0 0 0 c11; ...
230 0 0 0 0 0; ...
231 0 0 0 0 0];

```

```

232     A1 = [0 0 0 -b42 0 0; ...
233           0 0 0 0 0 0; ...
234           0 0 0 0 0 0; ...
235           0 0 0 0 0 0; ...
236           0 0 0 0 0 0];
237     Af = [-a7 0 0 0 0; ...
238           c7 -a8 0 0 0; ...
239           0 v3 -a9 a10 0; ...
240           0 0 c9 -a10 0; ...
241           0 0 0 b10 -a11];
242     A = [Ao Au; A1 Af];
243
244     %% Diffusion
245     D = zeros(11); D(9,9) = Dahl*(pi/L)^2;
246
247     vals.Ao = Ao;
248     vals.A = A;
249     vals.AD = A-D;
250
251     XC = (VC*NC*C* epC/KC/gC)*nC*aC^(nC-1)/(1 + aC^nC)^2;
252     XT = (VT*NT*C*epTO/KT/gT)*nT*aT^(nT-1)/(1 + aT^nT)^2;
253     XL = (VL*NL*C* epL/KL/gL)*nL*aL^(nL-1)/(1 + aL^nL)^2;
254     X = -nthroot(XC*XT*XL,3)/gmO;
255     fX = 3*X^2/(4+2*X);
256     vals.fX = fX;
257
258     fb = c7*v3*c9*b10*b42*c11;
259     vals.fb = fb;
260
261
262     if (disp ~ = 0)
263         fprintf(1, '%% Steady State Values \n');
264         fprintf(1, '    aC = %1.4f \t \t pC = %1.5g \t \t mC = %1.5g \n', aC, pC, mC);
265         fprintf(1, '    aT = %1.4f \t \t pT = %1.5g \t \t mTO = %1.5g \n', aT, pT, mTO);
266         fprintf(1, '    aL = %1.4f \t \t pL = %1.5g \t \t mL = %1.5g \n', aL, pL, mL);
267         fprintf(1, '    \t \t \t \t \t pI = %1.5g \t \t mI = %1.5g \n', pI, mI);
268         fprintf(1, '    aA = %1.4f \t \t AHL = %1.5g \n', aA, AHL);
269         fprintf(1, '    aR = %1.4f \t \t pRA = %1.5g \t \t mTQ = %1.5g \n', aR, pRA, mTQ);
270         fprintf(1, '%% End Alphas \n');
271
272         fprintf(1, '%% Stability Check \n');
273         fprintf(1, '    X = %1.5g \n \t \t beta = %1.5g \n', X, beta);
274         fprintf(1, '    fb = %1.5g \n \n', fb);
275         fprintf(1, '    maxEig_rep = %1.5g \n', max(real(eig(Ao))));
276         fprintf(1, '    maxEig_sys = %1.5g \n', max(real(eig(A))));
277         fprintf(1, '    maxEig_diff = %1.5g \n', max(real(eig(A-D))));
278         fprintf(1, '%% End Stability Check \n');
279     end
280
281

```

```

282
283 % subfunction for using fzero (converge aL)
284 function diff = findAlphaT(aT, vals)
285     % extract constants
286     [sigC sigT sigL sigI sigRA] = deal(vals.sigmas(1), vals.sigmas(2), ...
287         vals.sigmas(3), vals.sigmas(4), vals.sigmas(5));
288     [VC VT VL VI VRA] = deal(vals.Vs(1), vals.Vs(2), vals.Vs(3), ...
289         vals.Vs(4), vals.Vs(5));
290     [NC NT NL NI NRA] = deal(vals.Ns(1), vals.Ns(2), vals.Ns(3), ...
291         vals.Ns(4), vals.Ns(5));
292     [nC nT nL nRA] = deal(vals.ns(1), vals.ns(2), vals.ns(3), vals.ns(4));
293     [BC BT BL BI BR] = deal(vals.Bs(1), vals.Bs(2), vals.Bs(3), ...
294         vals.Bs(4), vals.Bs(5));
295     C = vals.C;
296     pR = vals.pR;
297
298     % solve for alphas
299     aC = BC * (1/(1+aT^nT) + 1/sigC);
300     aA = BI * (1/(1+aT^nT) + 1/sigI);
301     aL = BL * (1/(1+aC^nC) + 1/sigL);
302     aR = aT/BR / (1/(1+(KRA/pR*(1+aA)/aA)^nRA) + 1/sigRA) - 1;
303
304     % recalculate and compare (use VT instead of aT to guarantee aR ~ -1)
305     VT2 = aR/(1 + aR) * aT / (1/(1 + aL^nL) + 1/sigT) / BT * VT;
306     diff = VT - VT2;
307 end
308 end

```

A.3 Analysis Code - Extract Parameters

```

1 % extract_qosc.m
2 % extract constants from qosc
3 [gC gT gL gI gA gmO gmQ] = deal(vals.gammas(1), vals.gammas(2), ...
4     vals.gammas(3), vals.gammas(4), vals.gammas(5), vals.gammas(6), vals.gammas(7));
5 [epC epTO epTQ epL epI] = deal(vals.eps(1), vals.eps(2), vals.eps(3), ...
6     vals.eps(4), vals.eps(5));
7 [KC KT KL KRA Kf Kr] = deal(vals.Ks(1), vals.Ks(2), vals.Ks(3), ...
8     vals.Ks(4), vals.Ks(5), vals.Ks(6));
9 [sigC sigT sigL sigI sigRA] = deal(vals.sigmas(1), vals.sigmas(2), ...
10     vals.sigmas(3), vals.sigmas(4), vals.sigmas(5));
11 [VC VT VL VI VRA] = deal(vals.Vs(1), vals.Vs(2), vals.Vs(3), vals.Vs(4), ...
12     vals.Vs(5));
13 [NC NT NL NI NRA] = deal(vals.Ns(1), vals.Ns(2), vals.Ns(3), vals.Ns(4), ...
14     vals.Ns(5));
15 [nC nT nL nRA] = deal(vals.ns(1), vals.ns(2), vals.ns(3), vals.ns(4));
16 [aC aT aL aA aR] = deal(vals.alphas(1), vals.alphas(2), vals.alphas(3), ...
17     vals.alphas(4), vals.alphas(5));
18 [pC pT pL pI AHL pRA] = deal(vals.ps(1), vals.ps(2), vals.ps(3), ...
19     vals.ps(4), vals.ps(5), vals.ps(6));
20 [mC mTO mTQ mL mI] = deal(vals.ms(1), vals.ms(2), vals.ms(3), ...

```

```

21     vals.ms(4),vals.ms(5));
22 [BC BT BL BI BR] = deal(vals.Bs(1),vals.Bs(2),vals.Bs(3),vals.Bs(4), ...
23     vals.Bs(5));
24
25 v3 = vals.v3;
26 pR = vals.pR;
27 Dahl = vals.D;
28 C = vals.C;
29 beta = vals.beta;
30 Ao = vals.Ao;
31 A = vals.A;
32 AD = vals.AD;
33
34 fX = vals.fX;
35 fb = vals.fb;

```

A.4 PDE Code - MATLAB Simulation Script

```

1  % run_qosc_pde.m
2  % Run full system on a line of cells
3
4  %%% SIMULATION PARAMETERS
5  N = 100;      % Number of grid point in spatial direction
6  L = 100e-6;  % spatial domain: [0,L]
7  d = 1;       % with diffusion (1) or without (0)
8
9  % a_1 u(1) + b_1 u'(1) = c_1
10 % a_N u(-1) + b_N u'(-1) = c_N
11 % bc = boundary condition matrix = [a_1 b_1 c_1; a_N b_N c_N]
12 bc = [0 1 0; 0 1 0];
13 [yvecT,D2T,D1T,phip,phim] = cheb2bc(N,bc);
14
15 %%% spatial coordinate xvecT \in [0,L]
16 xvecT = (L/2)*(yvecT + 1);
17
18 %%% import parameter list
19 global gC gT gL gmO gmQ gI gA
20 global epTO epTQ epC epL epI
21 global KC KT KL KRA kf kr
22 global sigC sigT sigL sigI sigRA
23 global VC VT VL VI VRA
24 global NC NT NL NI NRA
25 global nC nT nL nRA
26 global C v3 pR Dahl
27 params1
28
29 %%% calculate expected steady state for initial conditions
30 [aC aT aL aA aR] = steadystate();
31 pC = aC*KC; pT = aT*KT; pL = aL*KL; AHL = aA*(kr/kf); pI = gA*AHL/v3;
32 mIO = VT*NT*C * (1/(1+aL^nL) + 1/sigT) / gmO;

```

```

33 mTQ = VRA*NRA*C * (1/(1 + (KRA/pR*(1+aA)/aA)^nRA) + 1/sigRA) / gmQ;
34 mC = gC/epC * pC; mL = gL/epL * pL; mI = gI/epI * pI;
35 pRA = pR * AHL/(kr/kf + AHL);
36
37 %%% initial condition
38 ssvals = [mC pC mTO pT mL pL mI pI AHL pRA mTQ].';
39 ic = 2;
40 k = 3; % wave number in intial condition
41 i = 2;
42 switch(ic)
43     case 1 % cosine in all species
44         temp = ones(N,1) + 1/3*cos(k*pi*xvecT/L);
45         x0 = kron(ssvals ,temp);
46     case 2 % cosine in species i
47         x0 = kron(ssvals ,ones(N,1));
48         x0((i-1)*N+1:i*N) = ssvals(i)*(1 + 1/3*cos(k*pi*xvecT/L));
49     case 3 % random noise in all species
50         x0 = (randn(11*N,1)-0.5)/10 + 1;
51         x0 = x0 .* kron(ssvals ,ones(N,1));
52     case 4
53         k = 0;
54         x0 = kron(ssvals ,ones(N,1));
55         x0((i-1)*N+1:i*N) = ssvals(i)*1.5;
56 end
57
58
59 %%% simulation interval
60 Tspan = 0:3:30*3600;
61 i = 1; % which species to display
62 t1 = 3600/3*5+1;
63
64 options = odeset('RelTol',1e-9,'AbsTol',1e-10);
65
66 tic
67 [T,Y] = ode15s(@qosc_pde ,Tspan,x0,options ,L,N,D2T);
68 T = T/3600; % convert to hours
69 toc
70
71
72 names={'m_C', 'p_C', 'm_T_O', 'p_T', 'm_L', 'p_L', ...
73        'm_I', 'p_I', 'AHL', 'p_RA', 'm_T_Q'};
74 species = {'mC', 'pC', 'mTO', 'pT', 'mL', 'pL', 'mI', 'pI', 'A', 'pRA', 'mTQ'};
75 if(Dahl == 0)
76     dlab = 0;
77 else
78     dlab = -round(log10(Dahl));
79 end
80
81 tstart = 0; % initial display time
82 sv = 1; % save plot?

```

```

83 dx = 10;      % time skip (don't need to display every time point)
84
85 t1 = find(T>=tstart,1);
86 %for i = 1:length(names) % show all
87 for i = 4:4    % show selected
88     figure('Position',[100 100 280 210]) % resize figure
89     imagesc(T(t1:end),xvecT,Y(t1:dx:end,(i-1)*N+1:i*N).')
90     set(gca,'YDir','normal')
91     xlabel('time_(hr)')
92     ylabel('spatial_position')
93     title(names{i})
94     colorbar
95
96     if(sv)
97         set(gcf,'PaperPositionMode','auto');
98         print('-depesc','-tiff','-r300','sprintf('images/mat-k%i-d%i-%s', ...
99             k,dlab,species{i}))
100    end
101 end

```

A.5 PDE Code - Equations

```

1 % for use in run_qosc_packed
2 function xdot = qosc_pde(t,x,L,N,D2T)
3     %%% import parameter list
4     global gC gT gL gmO gmQ gI gA
5     global epTO epTQ epC epL epI
6     global KC KT KL KRA kf kr
7     global sigC sigT sigL sigI sigRA
8     global VC VT VL VI VRA
9     global NC NT NL NI NRA
10    global nC nT nL nRA
11    global C v3 pR Dahl
12
13    %%% STATES: [mC pC mTO pT mL pL mI pI A pRA mTQ]
14    xdot = zeros(11*N,1);
15    idx = cell(11,1);
16    for i = 1:11
17        idx{i} = 1+(i-1)*N:i*N;
18    end
19
20
21    %%% DIFFERENTIAL EQUATIONS
22    xdot(idx{1}) = VC*NC*C*(1./(1+(x(idx{4})/KT).^nT)+1/sigC) -gmO*x(idx{1});
23    xdot(idx{2}) = epC*x(idx{1}) - gC*x(idx{2});
24    xdot(idx{3}) = VT*NT*C*(1./(1+(x(idx{6})/KL).^nL)+1/sigT) -gmO*x(idx{3});
25    xdot(idx{4}) = epTO*x(idx{3}) + epTQ*x(idx{11}) - gT*x(idx{4});
26    xdot(idx{5}) = VL*NL*C*(1./(1+(x(idx{2})/KC).^nC)+1/sigL) -gmO*x(idx{5});
27    xdot(idx{6}) = epL*x(idx{5}) - gL*x(idx{6});
28    xdot(idx{7}) = VI*NI*C*(1./(1+(x(idx{4})/KT).^nT)+1/sigI) -gmQ*x(idx{7});

```

```

29     xdot(idx{8}) = epI*x(idx{7}) - gI*x(idx{8});
30     xdot(idx{9}) = v3*x(idx{8}) - kf*x(idx{9}).*(pR-x(idx{10})) ...
31     + kr*x(idx{10}) - gA*x(idx{9}) + (2/L)^2*Dahl*D2T*x(idx{9});
32     xdot(idx{10}) = kf*x(idx{9}).*(pR-x(idx{10})) - kr*x(idx{10});
33     xdot(idx{11}) = VRA*NRA*C*(1./(1+(KRA./x(idx{10}))).^nRA)+1/sigRA) ...
34     - gmQ*x(idx{11});
35 end

```

A.6 PDE Code - Parameters List

```

1  % params1.m
2  %%% PARAMETER LIST (PDE Simulations)
3  scale = 1;
4  %%% Degradation Rates (1/s)
5  gC = 2.89e-4; gT = gC; gL = gC;
6  gI = 1.16e-3; gA = 7.70e-4;
7  gmO = 5.78e-4; gmQ = 5.78e-3;
8  %%% Epsilons
9  epC = 4.4702e-4;
10 epTO = 2.2689e-6;
11 epL = 2.1129e-9;
12 epI = 2.6548e-5;
13 epTQ = 6.2240e-6;
14 %%% Dissociation Constants (M)
15 KC = 2.5e-8 * scale;
16 KT = 1.786e-10 * scale;
17 KL = 1e-13 * scale;
18 KRA = 1.5e-9 * scale;
19 kf = 1e9 / scale;
20 kr = 50;
21 %%% Leakage Ratios (no units)
22 sigC = 5050;
23 sigT = 620;
24 sigL = 131;
25 sigI = sigC;
26 sigRA = 167;
27 %%% Velocity Constants (1/s)
28 VC = 0.3;
29 VT = 0.23;
30 VL = 0.06;
31 VI = VC;
32 VRA = 0.26;
33 %%% Copy Numbers (range from 1.5 to 60) (was 1.5 before)
34 NC = 5;
35 NT = 5;
36 NL = 5;
37 NI = 5;
38 NRA = 5;
39 %%% Hill Coefficients (no units)
40 nC = 2;

```

```

41 nT = 2;
42 nL = 2;
43 nRA = 2;
44 %%% other parameters
45 C = 1.5e-9 * scale; % M (concentration of 1 molecule in 1 cell)
46 v3 = 0.267/20; % 1/s
47 pR = 1e-8 * scale; % M (1e-8 to 1e-6)
48 if(d)
49     Dahl = 1.667e-12; % m^2/s (with diffusion)
50 else
51     Dahl = 0; % (w/o diffusion)
52 end

```

A.7 PDE Code - Steady State Function

```

1 % steadystate.m
2 function [aC aT aL aA aR] = steadystate()
3 %%% import parameter list
4 global gC gT gL gmO gmQ gI gA
5 global epTO epTQ epC epL epI
6 global KC KT KL KRA kf kr
7 global sigC sigT sigL sigI sigRA
8 global VC VT VL VI VRA
9 global NC NT NL NI NRA
10 global nC nT nL nRA
11 global C v3 pR
12
13
14 %%% solve for alphas numerically
15 init = 10;
16 aT = fzero(@(aT) findAlphaT(aT), init);
17 aC = (VC*NC*C) * (epC/gC/gmO)/KC * (1/(1+aT^nT) + 1/sigC);
18 aA = (VI*NI*C) * (v3/gA)*(epI/gI/gmQ)/(kr/kf) * (1/(1+aT^nT) + 1/sigI);
19 aL = (VL*NL*C) * (epL/gL/gmO)/KL * (1/(1+aC^nC) + 1/sigL);
20 aR = aT/(epTQ/gT/gmQ/KT)/(VRA*NRA*C) / (1/(1+(KRA/pR*(1+aA)/aA)^nRA) ...
21     + 1/sigRA) - 1;
22
23
24 % subfunction for using fzero (converge aL)
25 function diff = findAlphaT(aT, vals)
26 % solve for alphas
27 aC = (VC*NC*C) * (epC/gC/gmO)/KC * (1/(1+aT^nT) + 1/sigC);
28 aA = (VI*NI*C) * (v3/gA)*(epI/gI/gmQ)/(kr/kf)*(1/(1+aT^nT) + 1/sigI);
29 aL = (VL*NL*C) * (epL/gL/gmO)/KL * (1/(1+aC^nC) + 1/sigL);
30 aR = aT/(epTQ/gT/gmQ/KT)/(VRA*NRA*C)/(1/(1+(KRA/pR*(1+aA)/aA)^nRA) ...
31     + 1/sigRA) - 1;
32
33 % recalculate and compare (use VT instead of aT to guarantee aR ~ -1)
34 VT2 = aR/(1 + aR) * aT/(1/(1 + aL^nL) + 1/sigT) / (epTO/gT/gmO/KT) ...
35     / (NT*C);

```

```

36         diff = VT - VT2;
37     end
38
39 end

```

A.8 SSC Code - Simulation Reaction File

```

1 diffusion A(R#) at ADiff
2
3 — TetR dimerization
4 rxn x:T(d#) y:T(d#) at TDimOn -> x.d # y.d
5 rxn T(d#1,p#) T(d#1,p#) at TDimOff -> break 1
6
7 — cI dimerization
8 rxn x:C(d#) y:C(d#) at CDimOn -> x.d # y.d
9 rxn C(d#1,p#) C(d#1,p#) at CDimOff -> break 1
10
11 — LacI dimerization
12 rxn x:L(d#) y:L(d#) at LDimOn -> x.d # y.d
13 rxn L(d#1,p#) L(d#1,p#) at LDimOff -> break 1
14
15 — AHL formation and binding
16 rxn y:I at Ic -> new A
17 rxn x:R(AHL#) y:A(R#) at AROn -> x.AHL # y.R
18 rxn R(AHL#1,d#) A(R#1) at AROff -> break 1
19
20 — LuxR dimerization (only with AHL bound)
21 rxn x:R(AHL#1,d#) A(R#1) y:R(AHL#2,d#) A(R#2) at RDimOn -> x.d # y.d
22 rxn R(AHL#1,d#3,p#) A(R#1) R(AHL#2,d#3,p#) A(R#2) at RDimOff -> break 3
23
24 — active promoters
25 rxn PrTO(op1#,op2#) at PrTMax -> new mC
26 rxn PrC(op1#,op2#) at PrCMax -> new mL
27 rxn PrL(op1#,op2#) at PrLMax -> new mTO
28 rxn PrTQ(op1#,op2#) at PrTMax -> new mI
29 rxn PrRA(op1#-,op2#-) at PrRAAct -> new mTQ
30
31 — leaking promoters
32 rxn PrTO at PrTLeak -> new mC
33 rxn PrC at PrCLeak -> new mL
34 rxn PrL at PrLLeak -> new mTO
35 rxn PrTQ at PrTLeak -> new mI
36 rxn PrRA at PrRALeak -> new mTQ
37
38 — binding promoters
39 rxn x:PrTO(op1#,op2#) y:T(p#,d#1) z:T(p#,d#1) at TOn ->
40     x.op1 # y.p
41     x.op2 # z.p
42 rxn x:PrC(op1#,op2#) y:C(p#,d#1) z:C(p#,d#1) at COn ->
43     x.op1 # y.p

```

```

44         x.op2 # z.p
45 rxn x:PrL(op1#,op2#) y:L(p#,d#1) z:L(p#,d#1) at LOn ->
46         x.op1 # y.p
47         x.op2 # z.p
48 rxn x:PrTQ(op1#,op2#) y:T(p#,d#1) z:T(p#,d#1) at TOn ->
49         x.op1 # y.p
50         x.op2 # z.p
51 rxn x:PrRA(op1#,op2#) y:R(p#,d#1) z:R(p#,d#1) at RAOOn ->
52         x.op1 # y.p
53         x.op2 # z.p
54
55 — unbinding promoters
56 rxn PrTO(op1#1,op2#2) T(p#1) T(p#2) at TOff -> break 1; break 2
57 rxn PrC(op1#1,op2#2) C(p#1) C(p#2) at COff -> break 1; break 2
58 rxn PrL(op1#1,op2#2) L(p#1) L(p#2) at LOff -> break 1; break 2
59 rxn PrTQ(op1#1,op2#2) T(p#1) T(p#2) at TOff -> break 1; break 2
60 rxn PrRA(op1#1,op2#2) R(p#1) R(p#2) at RAOOff -> break 1; break 2
61
62 — protein degradation
63 rxn x:T(d#,p#) at pDeg -> destroy x
64 rxn x:C(d#,p#) at pDeg -> destroy x
65 rxn x:L(d#,p#) at pDeg -> destroy x
66 rxn x:I at IDeg -> destroy x
67
68 — mRNA degradation
69 rxn x:mTO at mODeg -> destroy x
70 rxn x:mC at mODeg -> destroy x
71 rxn x:mL at mODeg -> destroy x
72 rxn x:mTQ at mQDeg -> destroy x
73 rxn x:mI at mQDeg -> destroy x
74
75 — AHL degradation
76 rxn x:A(R#) at ADeg -> destroy x
77
78 — translation
79 rxn x:mTO at eTO -> new T
80 rxn x:mC at eC -> new C
81 rxn x:mL at eL -> new L
82 rxn x:mTQ at eTQ -> new T; new F
83 rxn x:mI at eI -> new I
84
85 — FP degradation
86 rxn x:F at FDeg -> destroy x

```

A.9 SSC Code - Simulation Parameter File

```

1 TStart = 28
2 CStart = 28
3 LStart = 28
4

```

```
5 mTStart = 44
6 mCStart = 88
7 mLStart = 44
8
9 PrStart = 4
10
11 TDimOn = 100
12 CDimOn = 100
13 LDimOn = 100
14 TDimOff = 500
15 CDimOff = 500
16 LDimOff = 500
17 TOn = 100
18 COn = 100
19 LOn = 100
20 TOff = 500
21 COff = 500
22 LOff = 500
23 PrTMax = 2
24 PrCMax = 2
25 PrLMax = 2
26 PrTLeak = 0.002
27 PrCLeak = 0.002
28 PrLLeak = 0.002
29 pDeg = 0.0028881
30 mDeg = 0.0057762
31 eT = 9.2683e-4
32 eC = 1.8537e-3
33 eL = 1.8537e-3
34
35
36
37 IStart = 12
38 AStart = 20
39 RStart = 12
40
41 mTOSTart = 44
42 mTQStart = 4
43 mIStart = 4
44
45 RDimOn = 100
46 RDimOff = 4500
47 AROn = 100
48 AROff = 1000
49 RAOOn = 100
50 RAOOff = 4500
51 PrRAAct = 2
52 PrRALeak = 0.002
53 mODeg = 0.0057762
54 mQDeg = 0.057762
```

```

55 ADeg = 0.057762
56 IDeg = 0.057762
57 eTO = 9.2683e-4
58 eTQ = 9.2683e-3
59 eI = 1.5230e-1
60
61 Ic = 0.1
62 ADiff = 16.67

```

A.10 SSC Code - SSC Simulation Script

```

1  #!/bin/bash
2
3  # Run within sims folder
4  #
5  # Takes a configuration file for a suite of simulations,
6  # 1) Starts with $RXNFILE (specified in configFile)
7  # 2) Initial condition set by createCosLine.pl
8  # 4) Submits job through the PBS scheduler
9  #
10 # Assumes the command ssc is located within your PATH variable
11
12 if [ "$1" = "" ]; then
13     echo "startCosSims_<configFile>"
14     exit 0
15 fi
16
17 CONFIG=$PWD/$1
18 source $CONFIG
19
20 echo "Running_ with_ configuration_ from_ $CONFIG"
21 echo "Setting_ up_ configuration_ set_ $SIMNAME_ for_ $FOLDER"
22
23
24 echo "Creating_ SSC_ binary_ ..."
25 RXN=$SSCBIN-$SIMNAME
26 cp $SSCBIN.rxn $RXN.rxn
27 $BIN/createCosLine.pl $SPACE $CELLS $IC $WAVE >> $RXN.rxn
28 ssc $RXN.rxn
29 mv $RXN.rxn $FOLDER/.
30 mv $RXN $FOLDER/.
31 echo "done"
32
33
34 echo -n "Copying_ parameter_ set_ ..."
35 cp params $FOLDER/params-$SIMNAME.cfg
36 echo "done"
37
38
39 echo "Submitting_ simulations_ ..."

```

```

40 EXE=$BASEDIR/$FOLDER/$RXN
41 cd $FOLDER
42 $BIN/submitJobs $CONFIG $EXE
43 echo "done"

```

A.11 SSC Code - Config File

```

1 # identify this simulation
2 FOLDER="qosc"
3 SIMNAME="12-15-s"
4
5 # filename of the compiled SSC binary to use for this simulation
6 SSCBIN="qosc-s"
7
8 # BASEDIR contains this config file , job.base file , and params file
9 BASEDIR="/work/jhsia/ssc-jhsia"
10
11 # length of simulation in HOURS
12 HOURS=30
13
14 # output values once every SAMPLE seconds
15 SAMPLE=15
16
17 # parameters for cosine initial conditions
18 SPACE=0
19 CELLS=100
20 IC=88
21 WAVE=4
22
23 # use PBS batch job scheduler? set to "yes" or "no"
24 # should always be yes when using Psi cluster
25 # should be no when running on your own private server
26 PBS="yes"
27
28 # contains the directory with the required scripts
29 BIN="/work/jhsia/ssc-jhsia/bin"

```

A.12 SSC Code - Generate Initial Condition

```

1 #!/usr/bin/perl -w
2
3 # creates the geometry of a line of cells for the SSC compiler and enforces
4 # an initial signal cos waveform across the line of cells
5 # the C levels range from 0 to the maxA parameter
6
7 use warnings;
8 use strict;
9
10 if ($#ARGV < 3) {

```

```

11     die "createCosLine.pl<spacing><cells><maxA><wave_#>\n";
12 }
13
14 my $spacing = abs(int($ARGV[0]));
15 my $cells = abs(int($ARGV[1]));
16 my $maxA = abs(int($ARGV[2]));
17 my $wave = abs(int($ARGV[3]));
18
19 sub tabs {
20     my $T = shift(@_);
21     my $out = "";
22     for (my $i=0; $i < $T; $i++) {
23         $out .= "_____";
24     }
25     return $out;
26 }
27
28 sub defCells {
29     my ($spacing, $cells, $named, $tabs) = @_;
30
31     my $extra = $named ne "";
32     for (my $x=0; $x < $cells; $x++) {
33         my $mx = ($x+1)*($spacing+1)-1;
34         if ($extra) {
35             print tabs($tabs)."region_". $named."_"."$x\n";
36         }
37         print tabs($tabs+$extra)."move_$.mx_0_0\n";
38         print tabs($tabs+$extra+1)."box_width_1_height_1_depth_1\n";
39     }
40     return;
41 }
42
43 sub initialCond {
44     my ($cells, $named) = @_;
45
46     for (my $x=0; $x < $cells; $x++) {
47         my $d = int((cos($wave*3.1416*$x/$cells)+1)*$maxA/2);
48         print "new_T_ at_TStart_in_$.named"."_"."$x.""\n";
49         print "new_C_ at_CStart_in_$.named"."_"."$x.""\n";
50         print "new_L_ at_LStart_in_$.named"."_"."$x.""\n";
51         print "new_I_ at_IStart_in_$.named"."_"."$x.""\n";
52         print "new_A_ at_AStart_in_$.named"."_"."$x.""\n";
53         print "new_R_ at_RStart_in_$.named"."_"."$x.""\n";
54         print "new_mTO_ at_mTOStart_in_$.named"."_"."$x.""\n";
55         print "new_mC_ at_d_in_$.named"."_"."$x.""\n";
56         print "new_mL_ at_mLStart_in_$.named"."_"."$x.""\n";
57         print "new_mTQ_ at_mTQStart_in_$.named"."_"."$x.""\n";
58         print "new_mI_ at_mIStart_in_$.named"."_"."$x.""\n";
59         print "new_PrTO_ at_PrStart_in_$.named"."_"."$x.""\n";
60         print "new_PrC_ at_PrStart_in_$.named"."_"."$x.""\n";

```

```

61         print "new_PrL_at_PrStart_in_$named"."_".$x."\n";
62         print "new_PrTQ_at_PrStart_in_$named"."_".$x."\n";
63         print "new_PrRA_at_PrStart_in_$named"."_".$x."\n";
64         print "record_T(d#,p#)_in_$named"."_".$x."\n";
65         print "record_L(d#,p#)_in_$named"."_".$x."\n";
66         print "record_C(d#,p#)_in_$named"."_".$x."\n";
67         print "record_I_in_$named"."_".$x."\n";
68         print "record_A(R#)_in_$named"."_".$x."\n";
69         print "record_mTO_in_$named"."_".$x."\n";
70         print "record_mC_in_$named"."_".$x."\n";
71         print "record_mL_in_$named"."_".$x."\n";
72         print "record_mTQ_in_$named"."_".$x."\n";
73         print "record_mI_in_$named"."_".$x."\n";
74     }
75     return;
76 }
77
78 my $maxDem = ($spacing+1)*($cells+1)-1;
79 my $maxDemDiv2 = $maxDem/2-0.5;
80
81 defCells($spacing, $cells, "Cell", 0);
82 print "region_External\n";
83 print "_diff\n";
84 print ".....move_$maxDemDiv2_0_0\n";
85 print ".....box_width_$maxDem_height_1_depth_1\n";
86 print ".....union\n";
87 defCells($spacing, $cells, "", 3);
88 print "subvolume_edge_1\n";
89 print "\n\n";
90 initialCond($cells, "Cell");
91 exit(0);

```

A.13 SSC Code - Submit Jobs

```

1  #!/bin/bash
2
3  # starts simulations running.
4  # if using PBS, it will setup the job files and submit them
5  # if not using PBS the jobs are all started at the same time and sent to the
6  # background
7
8  source $1
9  EXE=$2
10
11 # convert HOURS to seconds....
12 DURATION='echo "$HOURS*60*60" | bc'
13
14 # minimum time step that we can use to approximate t=0
15 MINSTEP="1e-100"
16

```

```

17 if [ "$PBS" = "yes" ]; then
18     JOBFILE=job-$$SIMNAME
19     cp $BASEDIR/job.base $PWD/$JOBFILE
20     echo "#PBS_N_$FOLDER-$$SIMNAME" >> $PWD/$JOBFILE
21     echo "$EXE -c $PWD/params-$$SIMNAME.cfg -e $MINSTEP -t $MINSTEP > \
22     .....$PWD/data-$$SIMNAME.txt ; \
23     .....$EXE -c $PWD/params-$$SIMNAME.cfg -e $DURATION -t $SAMPLE >>> \
24     .....$PWD/data-$$SIMNAME.txt" >> $PWD/$JOBFILE
25     qsub $PWD/$JOBFILE
26 else
27     $EXE -c $PWD/params-$$SIMNAME.cfg -e $MINSTEP -t $MINSTEP > \
28     $PWD/data-$$SIMNAME.txt
29     $EXE -c $PWD/params-$$SIMNAME.cfg -e $DURATION -t $SAMPLE >>> \
30     $PWD/data-$$SIMNAME.txt&
31 fi

```

A.14 SSC Code - Base Batch Job File

```

1  #!/bin/sh
2  ### Declare myprogram non-rerunnable
3  #PBS -r n
4
5  ### Uncomment to send email when the job is completed:
6  #PBS -m ae
7  #PBS -M jhsia@eecs.berkeley.edu
8
9  ### Optionally specify destinations for your myprogram's output
10 ### Specify localhost and an NFS filesystem to prevent file copy errors.
11 #PBS -e localhost:/work/jhsia/ssc-jhsia/sim.err
12 #PBS -o localhost:/work/jhsia/ssc-jhsia/sim.log
13
14 ### Set the queue to "batch", the only available queue.
15 #PBS -q batch
16
17 ### Specify the number of cpus for your job. This example will run on 16 cpus
18 ### using 8 nodes with 2 processes per node.
19 ### You MUST specify some number of nodes or Torque will fail to load balance.
20 #PBS -l nodes=1
21
22 ### You should tell PBS how much memory you expect your job will use.
23 ### e.g. mem=1g or mem=1024m
24 #PBS -l mem=250m
25
26 ### You can override the default 1 hour real-world time limit.
27 ### Usage: -l walltime=HH:MM:SS
28 ### Jobs on the public clusters are currently limited to 10 days walltime.
29 #PBS -l walltime=200:00:00
30 #PBS -l cput=200:00:00

```

A.15 SSC Code - Format SSC Output File

```

1  #!/usr/bin/perl -w
2
3  # table2matlab.pl
4  # converts from SSC output to matlab input. Takes a parameter in the form
5  # "# variable name". Will try to truncate data if necessary.
6
7  use warnings;
8  use strict;
9
10 my $pattern;
11
12 if ($#ARGV == -1) {
13     $pattern = "#_C";
14 } else {
15     $pattern = $ARGV[0];
16 }
17
18 my $l1 = <STDIN>;
19 chomp($l1);
20 my @head = split(/ /, $l1);
21 shift(@head); # remove time header
22 for (my $i = 0; $i <= $#head; $i++) {
23     if ($head[$i] =~ /$pattern/) {
24         $head[$i] =~ s/.* in Cell_//;
25     } else {
26         $head[$i] = -1;
27     }
28 }
29
30 my $count = 1;
31 print "clear_X\n";
32 print "X($count,:) _=_[";
33 my $l = <STDIN>;
34 chomp($l);
35 my @value = split(/ /, $l);
36 my $t = shift(@value);
37 $t = $t/3600; # seconds to hours conversion
38 print "$t\t";
39 my $cells = $#value;
40 for (my $i = 0; $i <= $#value; $i++) {
41     if ($head[$i] != -1) {
42         print "$value[$i]\t";
43     }
44 }
45 print "];\n";
46 $count++;
47
48 # remove blank line and second header row

```

```

49 $1 = <STDIN>;
50 $1 = <STDIN>;
51
52
53 while ($1 = <STDIN>) {
54     chomp($1);
55     if ($1 eq "") {next;}
56     my @value = split(/ /, $1);
57     if ($#value == $cells+1) {
58         print "X($count,;) _=_[";
59         my $t = shift(@value);
60         $t = $t/3600; # seconds to hours conversion
61         print "$t\t";
62         for (my $i = 0; $i <= $#value; $i++) {
63             if ($head[$i] != -1) {
64                 print "$value[$i]\t";
65             }
66         }
67         print "];\n";
68         $count++;
69     }
70 }
71 print "\n";
72
73 print "T=_X(:,1);";
74 print "X=_X(:,2:end);";
75
76 exit(0);

```

A.16 SSC Code - Read Output File into MATLAB

```

1  %%% Extract data for line of cell simulations
2
3  folder = 'qosc';
4  simname = '03-08-low';
5  ovr = 0;    % override option for extractdata
6
7  [mC t] = extractdata(folder, ovr, simname, 'mC');
8  [pC t] = extractdata(folder, ovr, simname, 'C');
9  [mIO t] = extractdata(folder, ovr, simname, 'mIO');
10 [pT t] = extractdata(folder, ovr, simname, 'T');
11 [mL t] = extractdata(folder, ovr, simname, 'mL');
12 [pL t] = extractdata(folder, ovr, simname, 'L');
13 [mI t] = extractdata(folder, ovr, simname, 'mI');
14 [pI t] = extractdata(folder, ovr, simname, 'I');
15 [A t] = extractdata(folder, ovr, simname, 'A');
16 [mIQ t] = extractdata(folder, ovr, simname, 'mIQ');
17 [F t] = extractdata(folder, ovr, simname, 'F');
18 [R t] = extractdata(folder, ovr, simname, 'R');
19 [PrRA1 t] = extractdata(folder, ovr, simname, 'PrRA');

```

```

20
21 % uneven data set possible, so truncate to shortest length
22 l = min([length(mC) length(pC) length(mIO) length(pT) length(mL) ...
23         length(pL) length(mI) length(pI) length(A) length(mIQ) length(F)]);
24 t = t(1:l);
25 mC = mC(1:l, :);    pC = pC(1:l, :);
26 mIO = mIO(1:l, :); pT = pT(1:l, :);
27 mL = mL(1:l, :);   pL = pL(1:l, :);
28 mI = mI(1:l, :);   pI = pI(1:l, :);
29 A = A(1:l, :);     mIQ = mIQ(1:l, :);
30 F = F(1:l, :);     R = R(1:l, :);
31 PrRA1 = PrRA1(1:l, :);
32
33 % split species for multiple reporters
34 total = 0; rec = 0;
35 if(~isempty(R))
36     rec = 1;
37     pR = R(:, 1:3:end);
38     pRA = R(:, 2:3:end);
39     pRA2 = R(:, 3:3:end)/2;
40 end
41 if(size(pC, 2) > size(mC, 2))
42     total = 1;
43     pCt = pC(:, 2:2:end);
44     pC = pC(:, 1:2:end);
45     pTt = pT(:, 2:2:end);
46     pT = pT(:, 1:2:end);
47     pLt = pL(:, 2:2:end);
48     pL = pL(:, 1:2:end);
49     At = A(:, 2:2:end);
50     A = A(:, 1:2:end);
51 end
52
53 beep
54
55 %% display extracted data
56 species = {'mC', 'pC', 'mIO', 'pT', 'mL', 'pL', 'mI', 'pI', ...
57           'AHL', 'mIQ', 'F', 'pR', 'pRA', 'pRA2', 'PrRA1'};
58 N = size(mC, 2);
59 k = 0; % imprinted wave number
60 tstart = 50;
61 sv = 1; % save image?
62
63 if(rec)
64     e = 15;
65 else
66     e = 11;
67 end
68
69 C = 1.5e-9;

```

```

70 idx = find(t>=tstart,1);
71 %for i = 1:e % show all species
72 for i = [2 9] % show selected species
73     switch i
74         case 1
75             X = mC;
76         case 2
77             if(total)
78                 X = pCt;
79             else
80                 X = pC;
81             end
82         case 3
83             X = mTO;
84         case 4
85             if(total)
86                 X = pTt;
87             else
88                 X = pT;
89             end
90         case 5
91             X = mL;
92         case 6
93             if(total)
94                 X = pLt;
95             else
96                 X = pL;
97             end
98         case 7
99             X = mI;
100        case 8
101            X = pI;
102        case 9
103            if(total)
104                X = At;
105            else
106                X = A;
107            end
108        case 10
109            X = mIQ;
110        case 11
111            X = F;
112        case 12
113            X = pR;
114        case 13
115            X = pRA;
116        case 14
117            X = pRA2;
118        case 15
119            X = PrRA1;

```

```

120     end
121
122     if(1)
123         figure('Position',[50*i 100 420 158])
124         imagesc(t,1:N,C*X.')
125         set(gca,'YDir','normal')
126         ylabel('spatial_position_(\mmm)')
127         xlabel('time_(hr)')
128         zlabel(species{i})
129         title(sprintf('SSC_Simulation_results_for_%s-%s,_viewing_%s', ...
130             folder,simname,species{i}))
131         colorbar
132
133         if(sv)
134             set(gcf,'PaperPositionMode','auto');
135             print('-depsc','-tiff','-r300', ...
136                 sprintf('images/%s-%s-%s',folder,simname,species{i}))
137         end
138     end
139 end

```

A.17 SSC Code - Read Output File Helper

```

1  % function [X T] = extractdata(simname,species,scale,atc)
2  %
3  % Extract data from SSC-produced output files
4  % Uses DOS command to run PERL script table2matlab.pl
5  % Inputs:
6  %     folder -
7  %     ovr - override option (even if file exists)
8  %     simname - extension on data text file folder
9  %               (i.e. '12-05' for data-12-05.txt)
10 %     species - {'C','mC','T','mT','L','mL'}
11 function [X T] = extractdata(folder,ovr,simname,species)
12     curdir = cd;
13     base = ['C:\Users\Justin\Documents\ssc-jhsia\' folder '\'];
14
15     fname = strrep(simname,'-','_');
16     ofile = ['matlabdata\' species '_' fname '.m'];
17     ifile = ['data-' simname '.txt'];
18
19     % move to proper directory
20     cd(base)
21
22     % extract data file if it doesn't exist already
23     if(ovr || ~exist(ofile,'file'))
24         if(~exist(ifile,'file'))
25             cd(curdir);
26             error('extractdata:dataFile','Input_data_file_does_not_exist')
27         end

```

```

28     dos(['perl..\table2matlab3.pl"#_' species '"<' ifile '>' ofile]);
29     fprintf(1, '\nProcessing file %s...\n', ifile);
30     end
31
32     % run script to get variables X and T
33     run([base ofile]);
34     fprintf(1, 'Loaded %s.\n', species)
35
36     % return to original directory
37     cd(curdir);
38 end

```

A.18 Discrete Cosine Transform Code

```

1  %% DCT of MATLAB simulation
2  i = 2;
3  tstart = 15;
4  tend = 30;
5  sv = 0;
6
7  idx1 = find(T>=tstart, 1);
8  idx2 = find(T>=tend, 1);
9  X = Y(:, (i-1)*N+1:i*N);
10 Z = dct(X(idx1:idx2, :). ');
11 M = mean(abs(Z. '));
12 drop = 1;
13
14 figure, hold on
15 stem(drop:N-1, M(drop+1:N));
16 if(k~=0)
17     stem(k, M(k+1), 'r');
18 end
19 hold off;
20 xlabel('wave_number');
21 title(sprintf('DCT of %s starting from time %i hr to %i hr', ...
22     species{i}, tstart, tend));
23
24 if(sv)
25     set(gcf, 'PaperPositionMode', 'auto');
26     print('-depsc', '-tiff', '-r300', sprintf('...
27         'images/ssc-k%i-d%i-%s-dct%i_%i', k, dlab, species{i}, tstart, tend))
28 end
29
30
31 %% DCT of SSC simulation data (stored in X) ignoring beginning time samples
32 k = 6;
33 tstart = 15;
34 idx = find(T>=tstart, 1);
35 L = size(X, 2);
36

```

```

37 Y = dct(X(idx:end,:), '. ');
38 M = mean(abs(Y. '));
39
40 drop = 1;
41 hold on;
42 stem(drop:L-1,M(drop+1:L));
43 stem(k,M(k+1), 'r ');
44 hold off;
45 xlabel('wave_number ');
46 title(sprintf('DCT of %s starting from time %i hr', simname, tstart));

```

A.19 Excel Macro - Adjust Data

```

1 Sub SpectramaxDataRep()
2 '
3 ' SpectramaxData Macro
4 ' Adjust data from Spectramax M2 with only OD and 1 RFP measurement.
5 ' Use when plates differentiated by replicate (all inductions on each plate).
6 '
7     DateStr = "2015-07-10"
8     DescStr = "LS"
9     DestFile = DateStr & "-" & DescStr & ".xslm"
10    Folder = "C:\Users\" & Environ$("UserName") & "\Dropbox\JBEI\data\" & _
11        DateStr & "-" & DescStr
12    ChDir Folder
13
14    NumPlts = 2 ' # of different plate types
15    NumRows = 7 ' # of rows used on each plate
16    NumReps = 6 ' # of replicates of each plate type
17
18    RowNum = [{"pJH9-68", "pJH9-69", "pJH4-55", "pJH9-77", _
19        "pJH4-60", "pJH9-20", "BLANK", ""}; _
20        "pJH9-70", "pJH9-71", "pJH9-72", "pJH9-73", _
21        "pJH4-60", "pJH9-20", "BLANK", ""}]
22    RowTxt = [{"ZFP1_sLS_sRNA", "ZFP1_sLS_pJ23106", "ZFP1_sLS_pJ23108", _
23        "ZFP1_sLS_pJ23111", "pos_ind", "empty", "blank", ""}; _
24        "ZFP2_sLS_sRNA", "ZFP2_sLS_pJ23106", "ZFP3_sLS_sRNA", _
25        "ZFP3_sLS_pJ23106", "pos_ind", "empty", "blank", ""}]
26    IndLvl = [{"40_uM", "10_uM", "5_uM", "1_uM", "500_uM", "200_uM", _
27        "100_uM", "50_uM", "20_uM", "10_uM", "5_uM", "no_arab"}]
28
29    For i = 1 To NumPlts
30
31        For j = 1 To NumReps
32
33            Filename = DateStr & "-" & DescStr & "-P" & i & "-R" & j
34            Workbooks.OpenText Filename:= _
35                Folder & "\" & Filename & ".txt", _
36                Origin:=932, StartRow:=1, DataType:=xlDelimited, _
37                TextQualifier:= xlDoubleQuote, ConsecutiveDelimiter:=False, _

```

```

38         Tab:=True, Semicolon:=False, Comma:=False, Space:=False, -
39         Other:=False, FieldInfo:=Array(1, 1), -
40         TrailingMinusNumbers:=True
41
42     Sheets(Filename).Select
43     Sheets(Filename).Move After:=Workbooks( -
44         DestFile).Sheets(NumReps * (i - 1) + j)
45
46     CurSheet = ActiveSheet.Name
47     Range("A13").ClearContents
48     Range("A25").ClearContents
49     Range("G2:W2").Cut
50     Range("H2").Select
51     ActiveSheet.Paste
52     Application.CutCopyMode = False
53     Range("A" & (3 + NumRows)).FormulaR1C1 = "0.95"
54     Range("C4:N" & (4 + NumRows - 1)).Select
55     Selection.FormatConditions.Add Type:=xlCellValue, -
56         Operator:=xlLess, Formula1:="=$A$" & (3 + NumRows)
57     Selection.FormatConditions(Selection.FormatConditions.Count -
58         ).SetFirstPriority
59     With Selection.FormatConditions(1).Font
60         .Color = -16383844
61         .TintAndShade = 0
62     End With
63     With Selection.FormatConditions(1).Interior
64         .PatternColorIndex = xlAutomatic
65         .Color = 13551615
66         .TintAndShade = 0
67     End With
68     Selection.FormatConditions(1).StopIfTrue = False
69
70
71     ' calc average BLANK OD (assumes last row is BLANK)
72     Range("H" & (4 + NumRows)).Select
73     Selection.FormulaR1C1 = "BLANK:"
74     Selection.Font.Bold = True
75     Selection.HorizontalAlignment = xlRight
76     Range("I" & (4 + NumRows)).Value = "=AVERAGE(C" & (3 + NumRows) -
77         & ":N" & (3 + NumRows) & ")"
78
79     ' calc adjusted ODs
80     Range("B26").Select
81     Selection.FormulaR1C1 = "OD_Adj"
82     Selection.Font.Bold = True
83     Range("C3:N3").Copy
84     Range("C27").Select
85     ActiveSheet.Paste
86     Range("C28").Value = "=IF(C4>=$A$" & (3 + NumRows) & -
87         ",C16/(C4-$I$" & (4 + NumRows) & "),""")"

```

```

88         Range("C28").Copy
89         Range("C28:N" & (27 + NumRows)).Select
90         ActiveSheet.Paste
91
92         ' calc average EMPTY RFP Fluorescence (2nd to last row is EMPTY)
93         Range("H" & (28 + NumRows)).Select
94         Selection.FormulaR1C1 = "EMPTY:"
95         Selection.Font.Bold = True
96         Selection.HorizontalAlignment = xlRight
97         Range("I" & (28 + NumRows)).Value = "=AVERAGE(C" & (26 + _
98             NumRows) & ":N" & (26 + NumRows) & ")"
99
100        ' calc adjusted RFP numbers
101        Range("B38").Select
102        Selection.FormulaR1C1 = "RFP_BG_Adj"
103        Selection.Font.Bold = True
104        Range("C39:N39").Value = IndLvl
105        Range("C39:N39").Font.Bold = True
106        Range("C39:N39").HorizontalAlignment = xlCenter
107        For k = 1 To NumRows
108            Range("A" & (39 + k)).Value = RowNum(i, k)
109            Range("A" & (39 + k)).Font.Bold = True
110            Range("B" & (39 + k)).Value = RowTxt(i, k)
111        Next
112        Range("C40").Value = "=IF(C4>=$A$" & (3 + NumRows) & ",C28-$I$" _
113            & (28 + NumRows) & ", """"")
114        Range("C40").Copy
115        Range("C40:N" & (39 + NumRows)).Select
116        ActiveSheet.Paste
117        With Selection
118            .Borders(xlEdgeLeft).LineStyle = xlContinuous
119            .Borders(xlEdgeRight).LineStyle = xlContinuous
120            .Borders(xlEdgeBottom).LineStyle = xlContinuous
121            .Borders(xlEdgeTop).LineStyle = xlContinuous
122        End With
123
124        Application.CutCopyMode = False
125        ActiveSheet.Name = Right(CurSheet, 5)
126    Next
127 Next
128
129 End Sub

```

A.20 Excel Macro - Aggregate Replicates

```

1 Sub TFDataRep()
2 '
3 ' TFData Macro
4 ' Generate data sheet for Transfer Functions for SpectraMax M2 data.
5 ' Use when plates differentiated by replicate (all inductions on each plate).

```

```

6  '
7  Dim i As Integer
8  Dim C As String
9
10 NumPlts = 2  ' # of different plate types
11 NumRows = 7  ' # of rows used on each plate
12 NumReps = 6  ' # of replicates of each plate type
13 hrow = 39    ' header row for adjusted data
14
15 Sheets.Add.Name = "TFs"
16 ActiveSheet.Move -
17     Before:=ActiveWorkbook.Sheets(1)
18
19 ' Header line for induction levels
20 Sheets("P1-R1").Select
21 Range("C" & hrow & ":N" & hrow).Copy
22 Sheets("TFs").Select
23 Range("B1").Select
24 ActiveSheet.Paste
25 Range("P1").Select
26 ActiveSheet.Paste
27 Range("AC1").Value = "Ind_Fold"
28 Range("AC1").Font.Bold = True
29 Range("AC1").HorizontalAlignment = xlCenter
30 Rows("1").Select
31 With Selection.Borders(xlEdgeBottom)
32     .LineStyle = xlContinuous
33     .Weight = xlThin
34 End With
35
36 ' Copy over values from other sheets
37 For i = 1 To NumPlts
38
39     For j = 1 To NumRows - 2  ' ignore BLANK and EMPTY rows
40
41         Sheets("P" & i & "-R1").Select
42         txt1 = Range("A" & (hrow + j)).Value
43         txt2 = Range("B" & (hrow + j)).Value
44         Sheets("TFs").Select
45         For k = 1 To NumReps
46             Row = (NumRows - 2) * NumReps * (i - 1) +
47                 NumReps * (j - 1) + 1 + k
48             Range("A" & Row).Value = txt1 & "_R" & k
49             Range("B" & Row & ":M" & Row).Value = "=INDEX('P" & i &
50                 "-R" & k & "'!'" & (hrow + j) & ":" & (hrow + j) &
51                 ",1,COLUMN()+1)"
52         Next
53
54         Row = (NumRows - 2) * NumReps * (i - 1) + NumReps * (j - 1) -
55             + 1 + Application.Ceiling(NumReps / 2, 1)

```

```

56     Range("O" & (Row - 1)).Value = txt1 & " _AVG"
57     Range("O" & Row).Value = txt2
58     Range("O" & (Row + 1)).Value = txt1 & " _STD_ERR"
59     Range("AC" & Row).Value = "=AA" & Row & "/P" & Row
60
61     up = Application.Floor(NumReps / 2, 1)
62     dn = Application.Floor((NumReps - 1) / 2, 1)
63
64     Range("P" & Row).Value = "=AVERAGE(B" & (Row - dn) & ":B" & _
65         (Row + up) & ")"
66     Range("P" & (Row + 1)).Value = "=STDEV.P(B" & (Row - dn) & ":B" _
67         & (Row + up) & ")/SQRT(COUNTA(B" & (Row - dn) & ":B" & _
68         (Row + up) & ")))"
69
70     Range("P" & Row & ":P" & (Row + 1)).Copy
71     Range("Q" & Row & ":AA" & Row).Select
72     ActiveSheet.Paste
73
74     Rows((NumRows - 2) * NumReps * (i - 1) + NumReps * j + 1).Select
75     With Selection.Borders(xlEdgeBottom)
76         .LineStyle = xlContinuous
77         .ColorIndex = 0
78         .TintAndShade = 0
79         .Weight = xlThin
80     End With
81
82     Next
83
84     Next
85
86     Application.CutCopyMode = False
87
88     Columns("A").Font.Bold = True
89     Columns("O").Font.Bold = True
90     Columns("O").HorizontalAlignment = xlRight
91     Columns("A").EntireColumn.AutoFit
92
93 End Sub

```

Appendix B

Full SSC Simulation Results

Simulation result plots are split into two figures on the following two pages. Apologies for the excessive white space...

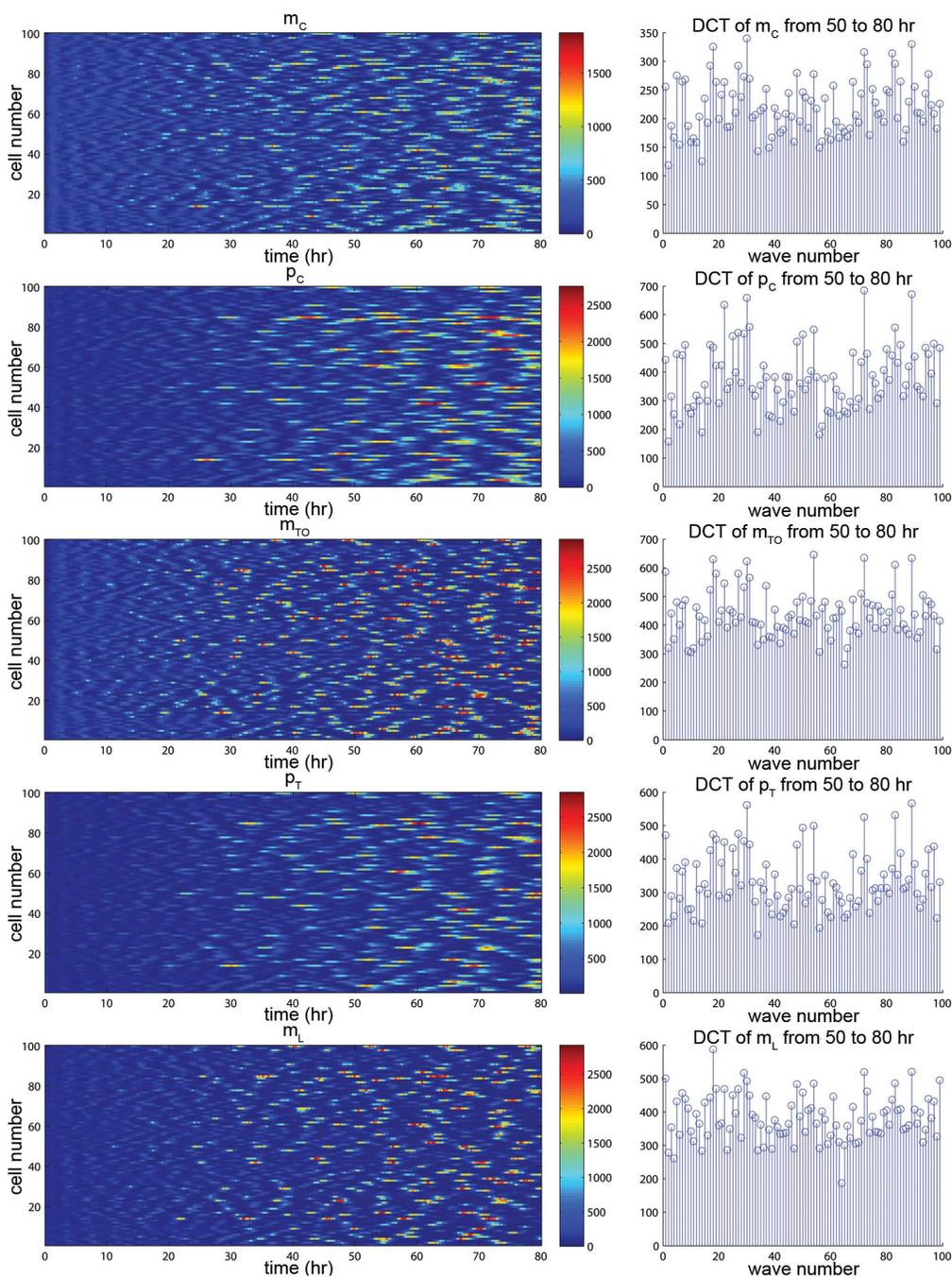


Figure B.1: Stochastic simulation results for Parameter Set 2 in line of cells with diffusion. Here $d_{AHL} = 1.667 \times 10^{-12} \text{m}^2/\text{s}$. Concentrations (colorbar) given in molecules per cell. All species set to steady state values rounded to nearest molecule. Stochasticity causes growing oscillations that eventually exhibit patterning. First five of the ten species are shown here. See Figure B.2 for the rest.

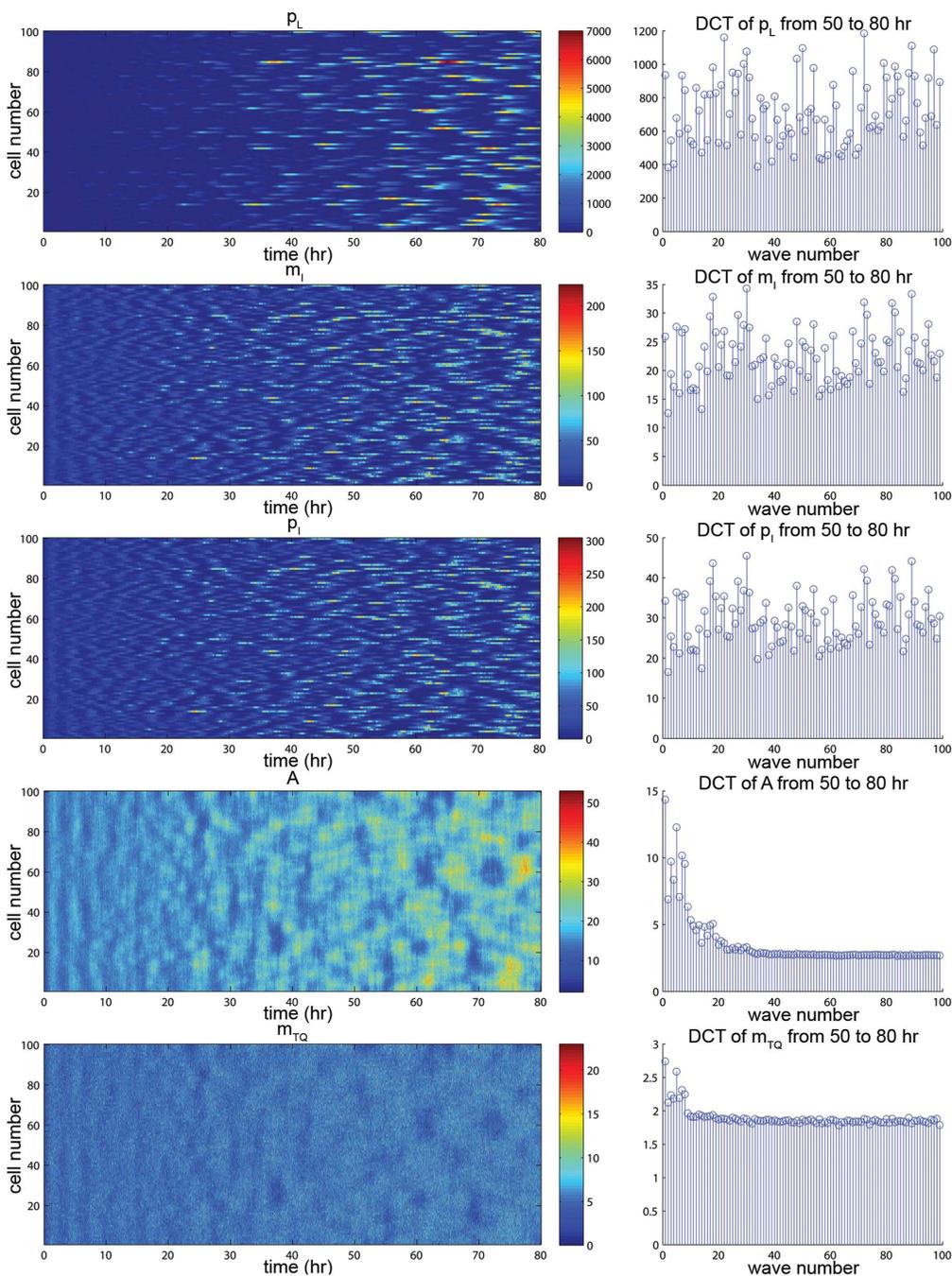


Figure B.2: Stochastic simulation results for Parameter Set 2 in line of cells with diffusion. Here $d_{AHL} = 1.667 \times 10^{-12} \text{m}^2/\text{s}$. Concentrations (colorbar) given in molecules per cell. All species set to steady state values rounded to nearest molecule. Stochasticity causes growing oscillations that eventually exhibit patterning. Last five of the ten species are shown here. See Figure B.1 for the rest.

Appendix C

Selected Sequence Info

C.1 Inv Plasmid 1: precursor (pJH4-21)

LOCUS pJH4-21 5116 bp DNA circular
DEFINITION

pWH29-77 operons moved onto Kan/p15a with sRNA binding site introduced.

pBAD_zfp-16-59, RBS AGGAGGAA, sense LS
pAraC_araC
pRPL-83_op-16-59_mCherry

FEATURES	Location/Qualifiers
CDS	complement(1524..2402) /gene="araC" /codon_start="0"
misc_feature	complement(587..1381) /gene="KanR" /note="encodes_nptII_(aka_AphA,_neoR),_kan_and_neo_re"
rep_origin	complement(4770..365) /gene="p15a"
CDS	3932..4636 /gene="mCherry" /codon_start="0"
CDS	3017..3553 /gene="ZFP16-59" /codon_start="0"
terminator	456..561 /gene="term_T0"
terminator	4658..4756 /gene="term_rrnD_T1"
CDS	2901..2996 /gene="repC_mini_cistron"

```

                                                    /codon_start="0"
misc_feature      2714..2804
                  /gene="sense_LS"
promoter          3869..3903
                  /gene="RPL-83"
promoter          complement(2553..2581)
                  /gene="pAraC"
promoter          2678..2705
                  /gene="pBAD"
misc_feature      3904..3931
                  /gene="Bujard_5'-UTR"
misc_feature      3851..3868
                  /gene="ZFP-3_binding_site"
misc_feature      2632..2645
                  /gene="CAP_site"
misc_feature      3003..3010
                  /gene="ZFP_RBS"
terminator        3578..3657
                  /gene="term_rrnB"
                  /note="Parts_Registry:_BBa_B0010"
terminator        3666..3706
                  /gene="term_T7"
                  /note="Parts_Registry:_BBa_B0012"
BASE COUNT      1356 a   1330 c   1211 g   1219 t
ORIGIN
1  tcagttccgg  gtaggcagtt  cgctccaagc  tggactgtat  gcacgaacce  cccgttcagt
61  ccgaccgctg  cgccttatec  ggtaactatc  gtcttgagtc  caaccgggaa  agacatgcaa
121 aagcaccact  ggcagcagcc  actggtaatt  gatttagagg  agttagtctt  gaagtcatgc
181 gccggttaag  gctaaactga  aaggacaagt  tttggtgact  gcgctctcc  aagccagtta
241 cctcggttca  aagagttggt  agctcagaga  acctcgaaa  aaccgcctg  caaggcggtt
301 ttttcgtttt  cagagcaaga  gattacgcgc  agacaaaac  gatctcaaga  agatcatctt
361 attaatcaga  taaaatattt  ctagatttca  gtgcaattta  tctctcaaa  tgtagcacct
421 gaagtcagcc  ccatacgata  taagttgtta  ctagtgcttg  gattctcacc  aataaaaaac
481 gccggcgccg  aaccgagcgt  tctgaacaaa  tccagatgga  gttctgaggt  cattactgga
541 tctatcaaca  ggagtccaag  cgagctctcg  aaccccagag  tcccgcctag  aagaactcgt
601 caagaaggcg  atagaaggcg  atgcgctgcg  aatcgggagc  ggcgataccg  taaagcacga
661 ggaagcggtc  agcccattcg  ccgccaagct  cttcagcaat  atcacgggta  gccaacgcta
721 tgtctgata  gcggtccgcc  acaccagcc  ggccacagtc  gatgaatcca  gaaaagcggc
781 cttttccac  catgatattc  ggcaagcagg  catcgccatg  ggtcacgacg  agatcctcgc
841 cgtcgggcat  gcgcgccttg  agcctggcga  acagttcggc  tggcgcgagc  ccctgatgct
901 cttcgtccag  atcctcctga  tcgacaagac  cggcttccat  ccgagtacgt  gctcgtcga
961 tgcgatgttt  cgcttggtgg  tcgaatgggc  aggtagccgg  atcaagcgta  tgcagccgcc
1021 gcattgcatc  agccatgatg  gatactttct  cggcaggagc  aaggtgagat  gacaggagat
1081 cctgccccgg  cacttcgccc  aatagcagcc  agtcccttcc  cgcttcagtg  acaacgtcga
1141 gcacagctgc  gcaaggaacg  cccgtcgtgg  ccagccacga  tagccgcgct  gcctcgtcct
1201 gcagttcatt  cagggcaccg  gacaggtcgg  tcttgacaaa  aagaaccggg  cgccccctgg
1261 ctgacagccg  gaacacggcg  gcatcagagc  agccgattgt  ctggtgtgce  cagtcatagc
1321 cgaatagcct  ctccacccaa  gcggccggag  aacctgcgtg  caatccatct  tgttcaatca
1381 tgcgaaacga  tctctatcct  gtctcttgat  cagatcatga  tcccctgcgc  catcagatcc
1441 ttggcggcaa  gaaagccatc  cagtttactt  tgcagggctt  cccaacctta  ccagaggcgc

```

```

1501 ccccagctgg caattccgac gtcttatgac aacttgacgg ctacatcatt cactttttct
1561 tcacaaccgg cacggaacte gctcgggctg gccccgggtc attttttaaa taccgcgag
1621 aaatagagtt gatcgtcaaa accaacattg cgaccgacgg tggcgatagg catccgggtg
1681 gtgctcaaaa gcagcttegc ctggctgata cgttggctct cgcgccagct taagacgcta
1741 atccctaact gctggcgcaa aagatgtgac agacgcgacg ggcacaagca aacatgctgt
1801 gcgacgctgg cgatatcaaa attgctgtct gccaggatgat cgctgatgta ctgacaagcc
1861 tcgcttaccg gattatccat cgggtgatgg agcgactcgt taatcgcttc catgcgccgc
1921 agtaacaatt gctcaagcag atttatcgcc agcagctccg aatagegcc tttcccttgc
1981 cggcggttaa tgatttgccc aaacaggteg ctgaaatgcg gctgggtgccc ttcatecggg
2041 cgaaagaacc ccgtattggc aaatattgac ggccagttaa gccattcatg ccagtaggcg
2101 cgcggacgaa agtaaaccce ctggtgatac cattcgcgag cctccggatg acgaccgtag
2161 tgatgaatct ctctggcggg gaacagcaaa atatacccg gtcggcaaac aaattctcgt
2221 cctgatattt tcaccacccc ctgaccgcga atggtgagat tgagaatata acctttcatt
2281 cccagcggtc ggtcgataaa aaaatcgaga taaccgttgg cctcaatcgg cgttaaacce
2341 gccaccagat gggcattaaa cgagtatccc ggcagcaggg gatcattttg cgttcagcc
2401 ataactttca tactcccgcc attcagagaa gaaaccaatt gtccatattg catcagacat
2461 tgccgtcact gcgcttttta ctggctcttc tcgctaacca aaccggtaac cccgcttatt
2521 aaaagcattc tgtaacaaag cgggaccaa gcatgacaa aaacgcgtaa caaaagtgtc
2581 tataatcacg gcagaaaagt ccacattgat tatttgcacg gcgtcacact ttgctatgcc
2641 atagcatttt tatccataag attagcggat tctacctgac gctttttatc gcaactctct
2701 actgtttctc cataaataaa aaggagtcgc tctgtccctc gccaaagttg cagaacgaca
2761 tcattcaaag aaaaaaacac tgagttgttt ttataatctt gtatatttag atattaaacg
2821 atattttaat atacataaag atatatattt ggggtgagcga ttccttaaac gaaattgaga
2881 ttaaggagtc gctctttttt atgtataaaa acaatcatgc aaatcattca aatcatttgg
2941 aaaatcacga tttagacaat ttttctaaaa ccggctactc taatagccgg ttgtaaggat
3001 ctaggaggaa ggatctatgc tggaaccagg atctaaaccg tacaaatgtc cggaatgtgg
3061 taaatccttc tcccgttctg ataatctggt acgtcatcaa cgtactcaca ctggatctaa
3121 accgtacaaa tgtccggaat gtggtaaatc cttctcccgt tctgataatc tggtacgtca
3181 tcaacgtact cacactggat ctaaaccgta caaatgtccg gaatgtggta aatccttctc
3241 ccagcgtgct catctggaac gtcacacacg tactcacact ggatctaaac cgtacaaatg
3301 tccggaatgt ggtaaatect tctcccacac ttccaatctg gttcgtcacc aacgtactca
3361 cactggatct aaaccgtaca aatgtccgga atgtggtaaa tcttctccc gttccgatca
3421 tctgactaat catcaacgta ctcacactgg atctaaaccg tacaaatgtc cggaatgtgg
3481 taaatccttc tcccgttctg ataatctggt acgtcatcaa cgtactcaca ctggatctaa
3541 aacctcttaa taaggatcca aactcgagta aggatctcca ggcacaaat aaaacgaaag
3601 gctcagtcga aagactgggc ctttcgtttt atctgttgtt tgcggtgaa cgctctctac
3661 tagagtcaca ctggctcacc ttcgggtggg ctttctgctg tttatataag taagtaagag
3721 tatacgtata tcggctaata acgtattaag gcgcttcggc gccttttttt atgggggtat
3781 tttcactcca atccacacgt ccaacgcaca gcaaacacca cgtcgacctc atcagctgcg
3841 tgctttctat gagagggaag gagaggagtt gacaattaat categgctca taacctttgt
3901 ggaacaattc attaaagagg agaaaaggtag catgcgtaaa ggagaagaag ataacatggc
3961 tatcattaaa gagttcatgc gttcaaaagt tcacatggag ggttctgtta acggtcacga
4021 gttcagatc gaaggcgaag gcgagggccg tccgtatgaa ggcaccaga ccgccaact
4081 gaaagtgact aaaggcggcc cgtgcctttt tgctgaggac atcctgagcc cgcaatttat
4141 gtacggttct aaagcgtatg ttaaacaccc agcggatata ccgactatc tgaagctgtc
4201 ttttcgggaa ggtttcaagt gggaacgcgt aatgaatttt gaagatggtg gtgtcgtgac
4261 cgctactcag gactcctccc tgcaggatgg cgagttcacc tataaagtta aactgcgtgg
4321 tactaatttt ccatctgatg gcccggtgat gcagaaaaag acgatgggtt gggaggcgtc
4381 tagcgaacgc atgtatccgg aagatggtgc gctgaaaggc gaaattaaac agcgctgaa
4441 actgaaagat ggcggccatt atgacgctga agtgaaaacc acgtacaaag ccaagaaacc

```

```

4501 tgtgcagctg cctggcgcgt acaatgtgaa tattaaactg gacatcacct ctcataatga
4561 agattatacg atcgtagagc aatatgagcg cgcgaggagggt cgtcattcta ccggtggcat
4621 ggatgaacta tacaataaat aaggatctca ggtctcatga tgggaactgc cagacatcaa
4681 ataaaacaaa aggctcagtc ggaagactgg gccttttgtt ttatctgttg tttgtcggtg
4741 aacactctcc cgggcgctag ggtacgggtg atatattccg cttectcgtc cactgactcg
4801 ctacgctcgg tcgttcgact gcggcgagcg gaaatggcct acgaacgggg cggagatttc
4861 ctggaagatg ccaggaagat acttaacagg gaagtgagag ggccgcggca aagccgtttt
4921 tccataggct ccgccccctt gacaagcadc acgaaatctg acgctcaaat cagtgggtggc
4981 gaaacccgac aggactataa agataccagg cgtttccccc tggcggctcc ctctgtcgtc
5041 ctctgttcc tgcctttcgg tttaccgggtg tcattccgct gttatggccc cgtttgtctc
5101 attccacgcc tgacac

```

//

C.2 Inv Plasmid 1: ZFP16-59, s04 (pJH7-17)

LOCUS pJH7-17 4853 bp DNA circular
DEFINITION

pJH5-57 with ZFP RBS changed to TCAGA.

pBAD_zfp-16-59, RBS TCAGA, sense 04

pAraC_araC

apFAB237_op-3_mCherry

FEATURES	Location/Qualifiers
CDS	complement(1524..2402) /gene="araC" /codon_start="0"
misc_feature	complement(587..1381) /gene="KanR" /note="encodes_nptII_(aka_AphA,_neoR),_kan_and_neo_re"
rep_origin	complement(4507..365) /gene="p15a"
CDS	3669..4373 /gene="mCherry" /codon_start="0"
CDS	2754..3290 /gene="ZFP16-59" /codon_start="0"
terminator	456..561 /gene="term_T0"
terminator	4395..4493 /gene="term_rrnD_T1"
misc_feature	2714..2749 /gene="sense_04"
promoter	3606..3640 /gene="RPL-83"
promoter	complement(2553..2581)

```

promoter      /gene="pAraC"
              2678..2705
misc_feature  /gene="pBAD"
              3641..3668
misc_feature  /gene="Bujard_5'-UTR"
              3588..3605
misc_feature  /gene="ZFP-3_binding_site"
              2632..2645
misc_feature  /gene="CAP_site"
              2729..2733
terminator    /gene="ZFP_RBS"
              3315..3394
terminator    /gene="term_rrnB"
              /note="Parts_Registry:_BBa_B0010"
              3403..3443
terminator    /gene="term_T7"
              /note="Parts_Registry:_BBa_B0012"
BASE COUNT    1261 a    1294 c    1167 g    1131 t
ORIGIN
1  tcagttccgg  gtaggcagtt  cgctccaagc  tggactgtat  gcacgaacce  cccgttcagt
61  ccgaccgctg  cgccttatec  ggtaactatc  gtcttgagtc  caaccgggaa  agacatgcaa
121 aagcaccact  ggcagcagcc  actggtaatt  gatttagagg  agttagtctt  gaagtcatgc
181 gccggttaag  gctaaactga  aaggacaagt  tttggtgact  gcgctctctc  aagccagtta
241 cctcggttca  aagagttggt  agctcagaga  accttcgaaa  aaccgcctctg  caaggcggtt
301 ttttcgtttt  cagagcaaga  gattacgcgc  agacaaaaac  gatctcaaga  agatcatctt
361 attaatcaga  taaaatattt  ctagatttca  gtgcaattta  tctcttcaaa  tgtagcacct
421 gaagtcagcc  ccatacgata  taagttgtta  ctagtgcttg  gattctcacc  aataaaaaac
481 gcccggcggc  aaccgagcgt  tctgaacaaa  tccagatgga  gttctgaggt  cttacttggc
541 tctatcaaca  ggagtccaag  cgagctctcg  aaccccagag  tcccgcctcg  aagaactcgt
601 caagaaggcg  atagaaggcg  atgcgctcgc  aatcgggagc  ggcgataccg  taaagcacga
661 ggaagcggtc  agcccattcg  ccgccaagct  cttcagcaat  atcacgggta  gccaacgcta
721 tgtctgata  gcggtccgcc  acaccagacc  ggccacagtc  gatgaatcca  gaaaagcggc
781 cttttccac  catgatattc  ggcaagcagg  catcgccatg  ggtcacgacg  agatctctcg
841 cgtegggcat  gcgcgccttg  agcctggcga  acagttcggc  tggecgagcg  cctgatgctt
901 cttcgtccag  atcctctgga  tcgacaagac  cggcttccat  ccgagtacgt  gctcgtctga
961 tgcgatgttt  cgcttggtgg  tcgaatgggc  aggtagccgg  atcaagcgtg  tgcagccgcc
1021 gcattgcate  agccatgatg  gatactttct  cggcaggagc  aaggtgagat  gacaggagat
1081 cctgccccgg  cacttcgccc  aatagcagcc  agtcccttcc  cgcttcagtg  acaacgtcga
1141 gcacagctgc  gcaaggaacg  cccgtcgtgg  ccagccacga  tagccgcgct  gcctcgtcct
1201 gcagttcatt  cagggcaccg  gacaggtcgg  tcttgacaaa  aagaaccggg  cgccccgctg
1261 ctgacagccg  gaacacggcg  gcatcagagc  agccgattgt  ctggttgctc  cagtcatagc
1321 cgaatagcct  ctccacccaa  gcggccggag  aacctgcgtg  caatccatct  tgttcaatca
1381 tgcgaaacga  tctctatcct  gtctcttgat  cagatcatga  tcccctgcgc  catcagatcc
1441 ttggcggcaa  gaaagccatc  cagtttactt  tgcagggctt  cccaacctta  ccagaggcgg
1501 ccccagctgg  caattccgac  gtcttatgac  aacttgacgg  ctacatcatt  cactttttct
1561 tcacaaccgg  cacggaactc  gctcgggctg  gcccccgtgc  attttttaa  taaccggcgg
1621 aaatagagtt  gatcgtcaaa  accaacattg  cgaccgacgg  ttggcgatagg  catccgggtg
1681 gtgctcaaaa  gcagcttcgc  ctggctgata  cgttggtcct  cgcgccagct  taagacgcta
1741 atccctaact  gctggcggaa  aagatgtgac  agacgcgacg  gcgacaagca  aacatgctgt
1801 gcgacgctgg  cgatatcaaa  attgctgtct  gccaggtgat  cgctgatgta  ctgacaagcc

```

```

1861 tcgcgtacce gattatccat cgggtgatgg agcgactcgt taategcttc catgcgccgc
1921 agtaacaatt gctcaagcag atttatcgcc agcagctccg aatagegccc ttccccctgc
1981 cggcggttaa tgatttgccc aaacaggteg ctgaaatgcg gctgggtgcg ttcatccggg
2041 cgaaagaacc ccgtattggc aaatattgac ggccagttaa gccattcatg ccagtaggcg
2101 cgcggacgaa agtaaaccce ctggtgatac cattcgcgag cctccggatg acgaccgtag
2161 tgatgaatct ctctggcggg gaacagcaaa atatacccg gtcggcaaac aaattctcgt
2221 cectgatttt tcaccacccc ctgaccgcga atggtgagat tgagaatata acctttcatt
2281 cccagcggtc ggtcgataaa aaaatcgaga taaccgttgg cctcaatcgg cgttaaacce
2341 gccaccagat gggcattaaa cgagtatccc ggcagcaggg gatcattttg cgcttcagcc
2401 ataactttca tactcccccc attcagagaa gaaaccaatt gtccatattg catcagacat
2461 tgccgtcact gcgtctttta ctggctcttc tcgctaacca aaccggtaac cccgcttatt
2521 aaaagcattc tgtaacaaag cgggacaaa gccatgacaa aaacgcgtaa caaaagtgtc
2581 tataatcacg gcagaaaagt ccacattgat tatttgcacg gcgtcacact ttgctatgcc
2641 atagcatttt tatccataag attagcggat tctacctgac gctttttatc gcaactctct
2701 actgtttctc catgccaaaa atcaataatc agacaacaag atgtgcgaac tcgatgctgg
2761 aaccaggatc taaaccgtac aaatgtccgg aatgtggtaa atccttctcc cgttctgata
2821 atctggtacg tcatcaacgt actcacactg gatctaaacc gtacaaatgt ccggaatgtg
2881 gtaaatecct ctccccgttct gataatctgg tacgtcatca acgtactcac actggatcta
2941 aaccgtacaa atgtccggaa tgtggtaaat ccttctccca gcgtgctcat ctggaacgtc
3001 atcaacgtac tcacactgga tctaaaccgt acaaatgtcc ggaatgtggt aaatecctct
3061 cccaatcttc caatctggtt cgteatcaac gtactcacac tggatctaaa ccgtacaaat
3121 gtccggaatg tggtaaatec ttctccccgt ccgatcatct gactaatcat caacgtactc
3181 aactggtatc taaaccgtac aaatgtccgg aatgtggtaa atccttctcc cgttctgata
3241 atctggtacg tcatcaacgt actcacactg gatctaaaac ctcttaataa ggatccaaac
3301 tcgagtaagg atctccaggc atcaaataaa acgaaaggct cagtcgaaag actgggcctt
3361 tcgtttttatc tgttgtttgt cgggtgaacgc tctctactag agtcacactg gctcaccttc
3421 ggggtgggcct ttctgcgttt atataagtaa gtaagagtat acgtatatcg gctaataacg
3481 tattaaggcg ctccggcgcc tttttttatg ggggtatfff cateccaate cacacgtcca
3541 acgcacagca aacaccacgt cgaccctatc agctgcgtgc tttctatgag agggaaggag
3601 aggagttgac aattaatcat cggctcataa cttttgtgga acaattcatt aaagaggaga
3661 aaggtaccat gcgtaaagga gaagaagata acatggctat cattaaagag ttcatgcgct
3721 tcaaagtcca catggagggt tctgttaacg gtcacgagtt cgagatcgaa ggccaaggcg
3781 agggccgctc gtatgaaggc accagaccgg ccaaactgaa agtgactaaa ggccggcccc
3841 tgccttttgc gtgggacatc ctgagccccg aatttatgta cggttctaaa gcgtatgfta
3901 aacaccacgc ggatateccg gactatctga agctgtcttt tccggaaggt ttcaagtggg
3961 aacgcgtaat gaatfffgtg gatggtggtg tcgtgaccgt cactcaggac tctccctgc
4021 aggatggcga gttcatctat aaagttaaac tgcgtggtac taatfffcca tctgatggcc
4081 cggatgatgca gaaaaagacg atgggttggg aggcgtctag cgaacgcatt tatccggaag
4141 atggtgcgct gaaaggcgaa attaaacagc gcctgaaact gaaagatggc ggccattatg
4201 acgctgaagt gaaaaccacg taaaagcca agaaacctgt gcagctgect ggccgctaca
4261 atgtgaatat taaactggac atcacctctc ataatagaaga ttatacgatc gtagagcaat
4321 atgagcgcgc ggagggtcgt cattctaccg gtggcatgga tgaactatac aaataataag
4381 gatctcaggt ctcatgatgg gaactgccag acatcaataa aaacaaaagg ctcagtcgga
4441 agactgggcc ttttgtttta tctgttgttt gtcggtgaac actctccccg gcgctagggt
4501 acgggtgata tattccgctt cctcgtcac tgactcgcta cgctcggtcg ttcgactgcg
4561 gcgagcggaa atggcttacg aacggggcgg agatttctg gaagatgcca ggaactactc
4621 taacagggaa gtgagagggc cgcggcaaaag ccgtttttcc atagctccg cccccctgac
4681 aagcatcacg aaatctgacg ctcaaatcag tgggtggcgaa acccgacagg actataaaga
4741 taccagcgct ttccccctgg cggtccctc gtgcgctctc ctgttctgca ctttcggttt
4801 accggtgtca ttccgctgtt atggcccgct ttgtctcatt ccacgcctga cac

```

//

C.3 Inv Plasmid 1: ZFP16-57, s05 (pJH9-56)

LOCUS pJH9-56 4853 bp DNA circular
DEFINITION

pJH9-54 with RBS changed to TC.

pBAD_zfp-16-57, RBS TC, sense 05
pAraC_araC
pBFRPL-69_op-18_mCherry

FEATURES	Location/Qualifiers
CDS	complement(1524..2402) /gene="araC" /codon_start="0"
misc_feature	complement(587..1381) /gene="KanR" /note="encodes_nptII_(aka_AphA,_neoR),_kan_and_neo_re"
rep_origin	complement(4507..365) /gene="p15a"
CDS	3669..4373 /gene="mCherry" /codon_start="0"
CDS	2754..3290 /gene="ZFP16-57" /codon_start="0"
terminator	456..561 /gene="term_T0"
terminator	4395..4493 /gene="term_rrnD_T1"
misc_feature	2714..2749 /gene="sense_05"
promoter	3606..3640 /gene="RPL-69"
promoter	complement(2553..2581) /gene="pAraC"
promoter	2678..2705 /gene="pBAD"
misc_feature	3641..3668 /gene="Bujard_5'-UTR"
misc_feature	3588..3605 /gene="ZFP-18_binding_site"
misc_feature	2632..2645 /gene="CAP_site"
misc_feature	2729..2730 /gene="ZFP_RBS"

```

terminator      3315..3394
                 /gene="term_rrnB"
                 /note="Parts_Registry:_BBa_B0010"
terminator      3403..3443
                 /gene="term_T7"
                 /note="Parts_Registry:_BBa_B0012"
BASE COUNT      1257 a    1293 c    1172 g    1131 t
ORIGIN
  1 tcagttccgg gtaggcagtt cgctccaagc tggactgtat gcacgaacce cccgttcagt
 61 ccgaccgctg cgccttatec ggtaactatc gtcttgagtc caaccgggaa agacatgcaa
121 aagcaccact ggcagcagcc actggtaatt gatttagagg agttagtctt gaagtcatgc
181 gccggttaag gctaaactga aaggacaagt ttgggtgact gcgctectcc aagccagtta
241 cctcggttca aagagttggt agctcagaga accttcgaaa aaccgccttg caaggcggtt
301 ttttcgtttt cagagcaaga gattacgcgc agacaaaac gatctcaaga agatcatctt
361 attaatcaga taaaatattt ctagatttca gtgcaattta tctctcaaa tgtagcacct
421 gaagtcagcc ccatacgata taagttgtta ctagtgtttg gattctcacc aataaaaaac
481 gcccgcgccg aaccgagcgt tctgaacaaa tccagatgga gttctgaggt cttacttggg
541 tctatcaaca ggagtcceag cgagctctcg aaccagagag tcccgtcagc aagaactcgt
601 caagaaggcg atagaaggcg atgcgctcgc aatcgggagc ggcgataacc taaagcacga
661 ggaagcggtc agcccattcg ccgccaagct cttcagcaat atcacgggta gccaacgcta
721 tgtcctgata gcggtccgcc acaccagccc ggccacagtc gatgaatcca gaaaagcggc
781 cttttccac  catgatattc ggcaagcagg cctcgccttg ggtcacgacg agatcctcgc
841 cgtegggcat gcgcgccttg agcctggcga acagttcggc tggcgcgagc cctgatgctt
901 cttegtccag atcactctga tgcacaagac cggcttccat ccgagtaagt gctcgtctga
961 tgcgatgttt cgcttggtgg tgaatgggc aggtagccgg atcaagcgtg tgcagccgcc
1021 gcattgcate agccatgatg gatactttct cggcaggagc aaggtgagat gacaggagat
1081 cctgccccgg cacttcgccc aatagcagcc agtcccctcc cgttccagtg acaacgtcga
1141 gcacagctgc gcaaggaacg cccgtcgtgg ccagccacga tagccgcgct gcctcgtcct
1201 gcagttcatt cagggcaccg gacaggtcgg tcttgacaaa aagaaccggg cgccccgtcg
1261 ctgacagccg gaacacggcg gcatcagagc agccgattgt ctggtgtgcc cagtcatagc
1321 cgaatagcct ctccacccaa gcggccggag aacctgcgtg caatccatct tgttcaatca
1381 tgcgaaaaga tctcctcctt gtctcttgat cagatcatga tcccctgcgc catcagatcc
1441 ttggcggcaa gaaagccatc cagtttactt tgcagggcct cccaacctta ccagagggcg
1501 ccccagctgg caattccgac gtcttatgac aacttgacgg ctacatcatt cactttttct
1561 tcacaaccgg cacggaactc gctcgggctg gcccccgtgc attttttaa taccgcgagc
1621 aatatagatt gatcgtcaaa accaacattg cgaccgacgg tggcgatagg catccgggtg
1681 gtgctcaaaa gcagcttcgc ctggctgata cgttggtcct cgcgccagct taagacgcta
1741 atccctaact gctggcggaa aagatgtgac agacgcgacg gcgacaagca aacatgctgt
1801 gcgacgctgg cgatatcaaa attgctgtct gccaggtgat cgtgatgta ctgacaagcc
1861 tcgctgacct gattatccat cgggtgatgg agcgactcgt taatcgcttc catgcgccgc
1921 agtaacaatt gctcaagcag atttatcgcc agcagctcgc aatagcgcce tccccctgc
1981 ccggcgttaa tgatttgccc aaacaggtcg ctgaaatgcg gctggtgcgc ttcattccggg
2041 cgaaagaacc ccgtattggc aatatattgac ggccagttaa gccattcatg ccagtagggc
2101 cgcggacgaa agtaaaccba ctggtgatac cattcgcgag cctccggatg acgaccctag
2161 tgatgaatct ctctggcggg gaacagcaaa atatcaccgc gtcggcaaac aaattctcgt
2221 cctgatthtt teaccacccc ctgaccgcga atggtgagat tgagaatata acctttcatt
2281 cccagcggtc ggtcgataaa aaaatcgaga taaccgttgg cctcaatgg cgtaaacc
2341 gccaccagat gggcattaaa cgagtatccc ggcagcaggg gatcattttg cgttcagccc
2401 atacttttca tactccccgc attcagagaa gaaaccaatt gtccatattg catcagacat
2461 tgccgtcact gcgtctttta ctggctcttc tcgctaacca aaccggtaac cccgcttatt

```

```

2521 aaaagcattc tgtaacaaag cgggaccaa gccatgacaa aaacgcgtaa caaaagtgtc
2581 tataatcacg gcagaaaagt ccacattgat tatttgcacg gcgtcacact ttgctatgcc
2641 atagcatttt tatccataag attagecgat tctacctgac gctttttatc gcaactctct
2701 actgtttctc catgcgtaaa atcaataatc agacaacaag atgtgcgaac tcgatgctgg
2761 aaccaggatc taaaccgtac aaatgtccgg aatgtggtaa atccttctcc cgttctgata
2821 atctggtaag tcatcaacgt actcacactg gatctaaacc gtacaaatgt ccggaatgtg
2881 gtaaatecct ctcccagcgt gctcatctgg aacgtcatca acgtactcac actggatcta
2941 aaccgtacaa atgtccggaa tgtggtaaat ccttctcccg ctctgatgaa ctggtaagtc
3001 atcaacgtac tcacactgga tctaaaccgt acaaatgtcc ggaatgtggt aaatccttct
3061 cccagaaate ttccctgatt gcccatcaac gtactcacac tggatctaaa ccgtacaaat
3121 gtccggaatg tggtaaatec ttctcccgtt ctgataatct ggtacgtcat caacgtactc
3181 aactgggata taaaccgtac aaatgtccgg aatgtggtaa atccttctcc cgttctgate
3241 atctgactac tcatcaacgt actcacactg gatctaaaac ctcttaataa ggatccaaac
3301 tcgagtaagg atctccaggc atcaaataaa acgaaaggct cagtcgaaag actgggcctt
3361 tcgtttttatc tgtttgtttgt cgggtgaacgc tctctactag agtcacactg gctcaccttc
3421 gggtgggcct ttctgcgttt atataagtaa gtaagagtat acgtatatcg gctaataacg
3481 tattaaggcg ctteggcgcc tttttttatg ggggtatfff catcccatec cacacgtcca
3541 acgcacagca aacaccacgt cgaccctatc agctgcgtgc tttctattgg gagatagtgg
3601 gagagtggac aattaatcat cggctcgtag ggtatgtgga acaattcatt aaagaggaga
3661 aaggtaccat gcgtaaagga gaagaagata acatggctat cattaagagag ttcatgcgct
3721 tcaaagtcca catggagggt tctgttaacg gtcacgagtt cgagatcgaa ggcggaaggcg
3781 agggcgcgtc gtatgaagge acccagaccg ccaaactgaa agtgactaaa ggcgccccgc
3841 tgccttttgc gtgggacatc ctgagccccg aatttatgta cggttctaaa gcgatatgta
3901 aacaccacgc ggatateccg gactatctga agctgtcttt tccggaaggt tcaagtggg
3961 aacgcgtaat gaattttgaa gatggtggtg tegtaccgt cactcaggac tctccctgc
4021 aggatggcga gttcatctat aaagttaaac tgcgtggtac taattttcca tctgatggcc
4081 cggatgatgca gaaaaagacg atgggttggg aggcgtctag cgaacgcattg tatccggaag
4141 atggtgcgct gaaagcgcaa attaaacagc gcctgaaact gaaagatggc ggccattatg
4201 acgctgaagt gaaaaccacg taaaagcca agaaacctgt gcagctgect ggcgcgta
4261 atgtgaatat taaactggac atcacctctc ataataaga ttatacgate gtagagcaat
4321 atgagcgcgc ggagggtcgt cattctaccg gtggcatgga tgaactatac aaataataag
4381 gatctcaggt ctcatgatgg gaactgccag acatcaaata aaacaaaagg ctcagtcgga
4441 agactgggccc ttttgtttta tctgttgttt gtcggtgaa actctcccgg gcgctagggt
4501 acgggtgata tattccgctt cctcgtcac tgactcgcta cgctcgctcg ttcgactgcg
4561 gcgagcggaa atggcttacg aacggggcgg agatttctg gaagatgcca ggaagatact
4621 taacagggaa gtgagagggc cgcggcaaag ccgtttttcc ataggctccg cccccctgac
4681 aagcatcacg aaatctgacg ctcaaactag tgggtggcga acccgacagg actataaaga
4741 taccagcgtt tccccctgg cggctccctc gtgcgctctc ctgttctctc ctttcggttt
4801 accggtgtca ttccgctggt atggcgcgct ttgtctcatt ccacgctga cac

```

//

C.4 Inv Plasmid 1: ZFP16-56, sLS (pJH4-64)

LOCUS pJH4-64 5116 bp DNA circular
DEFINITION

pJH4-40 with ZFP switched to ZFP 16-56 (respective op-30).

pBAD_zfp-16-56, RBS AAAGGATA, sense LS

pAraC_araC
pBFRPL-83_op-30_mCherry

FEATURES	Location/Qualifiers
CDS	complement(1524..2402) /gene="araC" /codon_start="0"
misc_feature	complement(587..1381) /gene="KanR" /note="encodes_nptII_(aka_AphA,_neoR),_kan_and_neo_re"
rep_origin	complement(4770..365) /gene="p15a"
CDS	3932..4636 /gene="mCherry" /codon_start="0"
CDS	3017..3553 /gene="ZFP16-56" /codon_start="0"
terminator	456..561 /gene="term_T0"
terminator	4658..4756 /gene="term_rrnD_T1"
CDS	2901..2996 /gene="repC_mini_cistron" /codon_start="0"
misc_feature	2714..2804 /gene="sense_LS"
promoter	3869..3903 /gene="RPL-83"
promoter	complement(2553..2581) /gene="pAraC"
promoter	2678..2705 /gene="pBAD"
misc_feature	3904..3931 /gene="Bujard_5'-UTR"
misc_feature	3851..3868 /gene="ZFP-30_binding_site"
misc_feature	2632..2645 /gene="CAP_site"
misc_feature	3003..3010 /gene="ZFP_RBS"
terminator	3578..3657 /gene="term_rrnB" /note="Parts_Registry:_BBa_B0010"
terminator	3666..3706 /gene="term_T7" /note="Parts_Registry:_BBa_B0012"
BASE COUNT	1365 a 1332 c 1201 g 1218 t
ORIGIN	

```

1 tcagttccgg gtaggcagtt cgtccaagc tggactgtat gcacgaacce cccgttcagt
61 ccgaccgctg cgccttatec ggtaactatc gtcttgagtc caaccggaa agacatgcaa
121 aagcaccact ggcagcagcc actggtaatt gatttagagg agttagtctt gaagtcatgc
181 gccggttaag gctaaactga aaggacaagt ttggtgact gcgctctcc aagccagtta
241 cctcggttca aagagttggt agctcagaga accttcgaaa aaccgcctg caaggcggtt
301 ttttcgtttt cagagcaaga gattacgcgc agacaaaac gatctcaaga agatcatctt
361 attaatcaga taaaatattt ctagatttca gtgcaattta tctctcaaa tgtagcacct
421 gaagtcagcc ccatacgata taagttgtta ctagtgttg gattctcacc aataaaaaac
481 gcccggcggc aaccgagcgt tctgaacaaa tccagatgga gttctgaggt cactactgga
541 tctatcaaca ggagtcceag cgagctctcg aaccccagag tcccgtcag aagaactcgt
601 caagaaggcg atagaaggcg atgcgctcgc aatcgggagc ggcgataccg taaagcacga
661 ggaagcggtc agcccattcg ccgccaagct cttcagcaat atcacgggta gccaacgcta
721 tgtctgata gcggtccgcc acaccagcc ggccacagtc gatgaatcca gaaaagcggc
781 cattttccac catgatattc ggcaagcagg catcgccatg ggtcacgacg agatctctgc
841 cgtcgggcat gcgcgccttg agcctggcga acagttcggc tggcgcgagc cctgatgct
901 ctctgtccag atcctctga tcgacaagac cggcttccat ccgagtaagt gctcgtctga
961 tgcgatgttt cgcttggtgg tcgaatgggc aggtagccgg atcaagcgt tgcagccgc
1021 gcattgcate agccatgatg gatactttct cggcaggagc aaggtgagat gacaggagat
1081 cctgccccgg cacttcgccc aatagcagcc agtcccctc cgttccagtg acaacgtcga
1141 gcacagctgc gcaaggaacg cccgtcgtgg ccagccacga tagccgcgct gccctgctct
1201 gcagttcatt cagggcaccg gacaggtcgg tcttgacaaa aagaaccggg cgccccgtcg
1261 ctgacagccg gaacacggcg gcatcagagc agccgattgt ctggtgtgcc cagtcatagc
1321 cgaatagcct ctccacccaa gcggccggag aacctgcgtg caatccatct tgttcaatca
1381 tgcgaaaega tctctactct gtctcttgat cagatcatga tcccctgcgc catcagatcc
1441 ttggcggcaa gaaagccatc cagtttactt tgcagggctt cccaacctta ccagagggcg
1501 cccagctggc caattccgac gtcttatgac aacttgacgg ctacatcatt cacttttct
1561 tcacaaccgg cacggaacte gctcgggctg gccccggtgc attttttaa taccgcgag
1621 aatatagatt gatcgtcaaa accaacattg cgaccgacgg tggcgatagg catccgggtg
1681 gtgctcaaaa gcagcttcgc ctggctgata cgttggtcct cgcgccagct taagacgcta
1741 atccctaact gctggcggaa aagatgtgac agacgcgacg gcgacaagca aacatgctgt
1801 gcgacgctgg cgatatcaaa attgctgtct gccaggtgat cgtgatgta ctgacaagcc
1861 tcgctatccc gattatccat cgggtgatgg agcgactcgt taatcgctt catgcgccg
1921 agtaacaatt gctcaagcag atttatcgcc agcagctccg aatagegccc ttccccttgc
1981 ccggcgttaa tgatttgccc aaacaggtcg ctgaaatgcg gctggtgcgc ttcatccggg
2041 cgaaaagaacc ccgtattggc aatatattgac ggccagttaa gccattcatg ccagtagggc
2101 cgcggacgaa agtaaaacca ctggtgatac cttcgcgag cctccgatg acgaccgtag
2161 tgatgaatct ctctggcggg gaacagcaaa atatcaccg gtcggcaaac aaattctcgt
2221 cctgatthtt tcaccacccc ctgaccgcga atggtgagat tgagaatata acctttcatt
2281 cccagcggtc ggtcgataaa aaaatcgaga taaccggttg cctcaatcgg cgttaaacc
2341 gccaccagat gggcattaaa cgagtatccc ggcagcaggg gatcattttg cgttcagcc
2401 atacttttca tactcccgcc attcagagaa gaaaccaatt gtccatattg catcagacat
2461 tgccgtcact gcgtctttta ctggctcttc tcgtaacca aaccggtaac cccgcttatt
2521 aaaagcattc tgtaacaaag cgggacaaa gccatgacaa aaacgcgtaa caaaagtgtc
2581 tataatcacg gcagaaaagt ccacattgat tatttcacg gcgtcacact ttgctatgcc
2641 atagcatttt tatccataag attagcggat tctacctgac gctttttatc gcaactctct
2701 actgtttctc cataaataaa aaggagtcgc tctgtccctc gccaaagttg cagaacgaca
2761 tcttcaaaag aaaaaaacac tgagttgttt ttataatctt gtatatttag atattaaacg
2821 atattttaa atacataaag atatataatt gggtgagcga ttcttaaac gaaattgaga
2881 ttaaggagtc gctctttttt atgtataaaa acaatcatgc aatcattca aatcatttg
2941 aaaatcacga tttagacaat ttttcaaaa ccggctactc taatagccgg ttgtaaggat

```

```

3001 ctaaaggata ggatctatgc tggaaccagg atctaaaccg tacaaatgtc cggaatgtgg
3061 taaatccttc tcccgttctg ataatctggg acgtcatcaa cgtactcaca ctggatctaa
3121 accgtacaaa tgtccggaat gtggtaaate cttctcccgt tctgatcatc tgactactca
3181 tcaacgtact cacactggat ctaaaccgta caaatgtccg gaatgtggta aatccttctc
3241 ccaatcttcc aatctggttc gtcatcaacg tactcacact ggatctaaac cgtacaaatg
3301 tccggaatgt ggtaaatect tctcccgcga agataacctg aaaaaccatc aacgtactca
3361 cactggatct aaaccgtaca aatgtccgga atgtggtaaa tccttctccc gttctgatca
3421 tctgactact catcaacgta ctcacactgg atctaaaccg tacaaatgtc cggaatgtgg
3481 taaatccttc tcccgtgaag ataacctgca tactcatcaa cgtactcaca ctggatctaa
3541 aacctcttaa taaggatcca aactegagta aggatctcca ggcatcaaat aaaacgaaag
3601 gctcagtcga aagactgggc ctttcgtttt atctgttgtt tgtcggtgaa cgctctctac
3661 tagagtcaca ctggctcacc ttcgggtggg ctttctgcg tttatataag taagtaagag
3721 tatacgtata tcggctaata acgtattaag gcgcttcggc gccttttttt atgggggtat
3781 tttcatccca atccacacgt ccaacgcaca gcaaaccaca cgtcgacctc atcagctgcg
3841 tgctttctat tagtgggaagg aatgggagtt gacaattaat catcggtcca taacctttgt
3901 ggaacaatc attaaagagg agaaaggtag catgcgtaaa ggagaagaag ataacatggc
3961 tateattaaa gagttcatgc gttcctaaagtcacatggag ggttctgtta acggtcacga
4021 gttcagatc gaaggecgaag gcgagggccg tccgtatgaa ggcaccaga ccgcccact
4081 gaaagtgact aaaggecggc cgtctccttt tgcgtgggac atcctgagcc cgcaatztat
4141 gtacggttct aaagcgtatg ttaaaccacc agcggatata ccggactata tgaagctgct
4201 ttttccggaa ggtttcaagt gggaacgcgt aatgaatttt gaagatgggt gtgtcgtgac
4261 cgtcactcag gactcctccc tgcaggatgg cgagttcctc tataaagtta aactgcgtgg
4321 tactaatttt ccatctgatg gcccggatg gcagaaaaag acgatgggtt gggaggegtc
4381 tagcgaacgc atgtatccgg aagatgggtgc gctgaaaggc gaaattaaac agcgcctgaa
4441 actgaaagat ggcggccatt atgacgctga agtgaaaacc acgtacaaag ccaagaaacc
4501 tgtgcagctg cctggecgtg acaatgtgaa tattaactg gacatcactc ctcataatga
4561 agattatacg atcgtagagc aatatgagcg cgcggagggt cgtcattcta ccggtggcat
4621 ggatgaacta tacaataat aaggatctca ggtctcatga tgggaactgc cagacatcaa
4681 ataaaacaaa aggctcagtc ggaagactgg gccttttgtt ttatctgttg tttgtcggtg
4741 aacactctcc cgggcgctag ggtacgggtg atatatccg ctctctcgtc cactgactcg
4801 ctacgctcgg tcgttcgact gcggcgagcg gaaatggctt acgaacgggg cggagatttc
4861 ctggaagatg ccaggaagat acttaacagg gaagtgagag ggccgcggca aagccgtttt
4921 tccataggct ccgccccctt gacaagcctc acgaaatctg acgctcaaat cagtgggtggc
4981 gaaacccgac aggactataa agataaccagg cgtttcccc tggcggctcc ctctgctgct
5041 ctctgttcc tgccttctgg tttaccgggt teattcctgct gttatggccg cgtttgtctc
5101 attccacgcc tgacac

```

//

C.5 Inv Plasmid 2: J23108, a04 (pJH5-56)

LOCUS pJH5-56 2053 bp DNA circular
DEFINITION

pJH4-74 with sRNA changed to A04 (Vivek).

pJ23108_antisense_04

FEATURES Location/Qualifiers

```

rep_origin      complement (848..1530)
                 /gene="ColE1"
misc_feature    complement (1657..263)
                 /gene="CmR"
gene            447..561
                 /gene="sRNA_04"
terminator      1536..1641
                 /gene="term_T0"
promoter        412..446
                 /gene="pJ23108"
terminator      586..665
                 /gene="term_rrnB"
                 /note="Parts_Registry:_BBa_B0010"
terminator      674..714
                 /gene="term_T7"
                 /note="Parts_Registry:_BBa_B0012"
BASE COUNT     543 a      499 c      481 g      530 t
ORIGIN
1  acactatccc atatcaccag ctcaccgtct ttcattgcca tacgaaatc cggatgagca
61  ttcacaggc  gggcaagaat gtgaataaag gccggataaa acttgtgctt atttttcttt
121 acggtcttta aaaaggccgt aatatccagc tgaacggctc ggttataggt acattgagca
181 actgactgaa atgcctcaaa atgttcttta cgatgccatt gggatataatc aacggtggtgta
241 tatecagtga tttttttctc cattttagct tccttagctc ctgaaaatct cgataactca
301 aaaaatacgc ccggtagtga tcttatttca ttatggtgaa agttggaacc tcttacgtgc
361 cgatcaacgt ctcattttcg ccagatateg aattctaaag atctggcacg tctgacagct
421 agctcagtec taggtataat gctagctcgc acatcttggt gtctgattat tgatttttgg
481 cgaaaaccatt tgatcatatg acaagatgtg tateccacctt aacttaatga tttttaccaa
541 aatcattagg ggattcatca gggatcctaa ctcgagtaag gatctccagg catcaaataa
601 aacgaaaggc tcagtcgaaa gactgggcct ttcgttttat ctgttgtttg tcggtgaacg
661 ctctctacta gagtcacact ggctcacctt cgggtggggcc tttctgcgctt tatacctagg
721 gcgttcggct gcggcgagcg gtatcagctc actcaaaggc ggtaatacgg ttatccacag
781 aatcagggga taacgcagga aagaacatgt gagcaaaagg ccagcaaaag gccaggaacc
841 gtaaaaaggc cgcgttgctg gcgtttttcc ataggctccg cccccctgac gagcatcaca
901 aaaatcgacg ctcaagtcag aggtggcgaa acccgacagg actataaaga taccaggcgt
961 tteccccctg aagctccctc gtgcgctctc ctgttccgac cctgcccgtt accggatacc
1021 tgtccgcctt tctcccttcg ggaagcgtgg cgctttctca tagctcacgc tgtaggtatc
1081 tcagttcggg gttaggtcgtt cgtccaagc tgggctgtgt gcacgaacce cccgttcagc
1141 ccgaccgctg cgccttatcc ggtaactatc gtcttgagtc caaccggta agacacgact
1201 tategccact ggcagcagcc actggtaaca ggattagcag agcgaggtat gtaggcggtg
1261 ctacagagtt cttgaagtgg tggcctaact acggctacac tagaaggaca gtatttggtgta
1321 tctgcgctct gctgaagcca gttaccttcg gaaaaagagt tggtagctct tgatccggca
1381 aacaaaccac cgctggtagc ggtggttttt ttgtttgcaa gcagcagatt acgcgcagaa
1441 aaaaaggatc tcaagaagat cctttgatct tttctacggg gtctgacgct cagtggaaacg
1501 aaaactcacg ttaagggatt ttggtcatga ctagtcttg gattctcacc aataaaaaaac
1561 gccggcgggc aaccgagcgt tctgaacaaa tcagatgga gttctgaggt cattactgga
1621 tctatcaaca ggagtccaag cgagctcgat atcaaattac gccccgcctt gccactctc
1681 gcagtaactg tgtaattcat taagcattct gccgacatgg aagccatcag aaacggcatg
1741 atgaacctga atcgccagcg gcatcagcac cttgtcgctt tgcgtataat atttgcccct
1801 ggtgaaaacg ggggcgaaga agttgtccat attggccacg ttaaatcaa aactggtgaa
1861 actcaccag  ggattggctg agacgaaaaa catattctca ataaaccctt tagggaaata

```

```

1921 ggccaggttt tcaccgtaac acgccacatc ttgcgaatat atgtgtagaa actgccggaa
1981 atcgctgtgg tattcactcc agagcgatga aaacgtttca gtttgctcat ggaaaacggt
2041 gtaacaaggg tga

```

//

C.6 Inv Plasmid 2: J23118, a05 (pJH6-26)

LOCUS pJH6-26 2053 bp DNA circular
DEFINITION

pJH5-75 with sRNA changed to A05 (Vivek).

pJ23118_antisense_05

FEATURES	Location/Qualifiers
rep_origin	complement(848..1530) /gene="ColE1"
misc_feature	complement(1657..263) /gene="CmR"
gene	447..561 /gene="sRNA_05"
terminator	1536..1641 /gene="term_T0"
promoter	412..446 /gene="pJ23118"
terminator	586..665 /gene="term_rrnB" /note="Parts_Registry:_BBa_B0010"
terminator	674..714 /gene="term_T7" /note="Parts_Registry:_BBa_B0012"

BASE COUNT 541 a 499 c 482 g 531 t

ORIGIN

```

1  acactatccc  atatcaccag  ctcaccgtct  ttcattgcca  tacgaaatc  cggatgagca
61  ttcatacaggc  gggcaagaat  gtgaataaag  gccgataaaa  acttgtgctt  atttttcttt
121 acggtcttta  aaaaggccgt  aatatccagc  tgaacggtct  ggttataggt  acattgagca
181 actgactgaa  atgcctcaaa  atgtttctta  cgatgccatt  gggatatatac  aacggtggta
241 tatccagtga  tttttttctc  cattttagct  tccttagctc  ctgaaaatct  cgataactca
301 aaaaatacgc  ccggtagtga  tcttatttca  ttatggtgaa  agttggaacc  tcttacgtgc
361 cgatcaacgt  ctcattttcg  ccagatateg  aattctaaag  atctggcagc  tttgacggct
421 agctcagtc  taggtattgt  gctagctcgc  acatcttggt  gtctgattat  tgattttacg
481 cgaaacctcc  tgatcatatg  acaagatgtg  tatccacctt  aacttaatga  tttttaccaa
541 aatcattagg  ggattcatca  gggatcctaa  ctcgagtaag  gatctccagg  catcaataaa
601 aacgaaagc  tcagtcgaaa  gactgggcct  ttcgttttat  ctggtgtttg  tcggtgaacg
661 ctctctacta  gagtcacact  ggctcacctt  cgggtgggccc  tttctgctgt  tatacctagg
721 gcgttcggct  gcggcgagcg  gtatcagctc  actcaaagc  ggtaatacgg  ttatccacag
781 aatcagggga  taacgcagga  aagaacatgt  gagcaaaagg  ccagcaaaag  gccaggaacc
841 gtaaaaagc  cgcgttgctg  gcgtttttcc  ataggtctcc  cccctctgac  gagcatcaca

```

```

901 aaaatcgacg ctcaagtcag aggtggcgaa acccgacagg actataaaga taccaggcgt
961 ttccccctgg aagctccctc gtgcgctctc ctgttccgac cctgcccgtt accggatacc
1021 tgtccgcctt tctcccttcg ggaagcgtgg cgctttctca tagctcacgc tgtaggtatc
1081 tcagttcggg gtaggtcggt cgctccaagc tgggctgtgt gcacgaacc cccgttcagc
1141 ccgaccgctg cgccttatec ggtaactatc gtcttgagtc caaccggta agacacgact
1201 tategccact ggcagcagcc actggtaaca ggattagcag agcgaggat gttaggcggtg
1261 ctacagagtt cttgaagtgg tggcctaact acggctacac tagaaggaca gtatttggtg
1321 tctgcgctct gctgaagcca gttaccttcg gaaaaagagt tggtagctct tgatccggca
1381 aacaaaccac cgctggtagc ggtggttttt ttgtttgcaa gcagcagatt acgcgcagaa
1441 aaaaaggate tcaagaagat cctttgatct tttctacggg gtctgacgct cagtggaacg
1501 aaaactcacg ttaagggatt ttggtcatga ctagtgcttg gattctcacc aataaaaaac
1561 gccccggcggc aaccgagcgt tctgaacaaa tccagatgga gttctgaggt cactactgga
1621 tctatcaaca ggagtcgaag cgagctcgat atcaaattac gccccgcct gccactcgc
1681 gcagtactgt tgtaattcat taagcattct gccgacatgg aagccatcac aaacggcatg
1741 atgaacctga atcgccagcg gcatcagcac cttgtcgct tgcgtataat atttgccat
1801 ggtgaaaacg ggggcgaaga agttgtccat attggccacg tttaaatcaa aactggtgaa
1861 actcaccag ggattggctg agacgaaaaa catattctca ataaaccctt tagggaaata
1921 ggccaggttt tcaccgtaac acgccacatc ttgcgaatat atgtgtagaa actgccggaa
1981 atcgtcgtgg tattcactcc agagcgatga aaacgtttca gtttgctcat gaaaaacggt
2041 gtaacaaggg tga

```

//

C.7 Inv Plasmid 2: J23106, aLS (pWH39-27)

LOCUS pWH39-27 2029 bp DNA circular

DEFINITION

Lucks sRNA plasmid from Will.

pJ23106_antisense_LS

FEATURES	Location/Qualifiers
rep_origin	complement(824..1506) /gene="ColE1"
misc_feature	complement(1633..263) /gene="CmR"
terminator	1512..1617 /gene="term_T0"
gene	447..537 /gene="sRNA_LS"
promoter	412..446 /gene="pJ23106"
terminator	562..641 /gene="term_rrnB" /note="Parts_Registry:_BBa_B0010"
terminator	650..690 /gene="term_T7" /note="Parts_Registry:_BBa_B0012"

```

BASE COUNT      533 a      493 c      480 g      523 t
ORIGIN
  1  acactatccc  atatcaccag  ctcaccgtct  ttcattgcca  tacgaaattc  cggatgagca
 61  ttcatcaggc  gggcaagaat  gtgaataaag  gccggataaa  acttgtgctt  atttttcttt
121  acggtcttta  aaaaggccgt  aatatccagc  tgaacggctc  ggttataggt  acattgagca
181  actgactgaa  atgcctcaaa  atgttcttta  cgatgccatt  gggatataatc  aacggtggtg
241  tatccagtga  tttttttctc  ctttttagct  tccttagctc  ctgaaaatct  cgataactca
301  aaaaatacgc  ccggtagtga  tcttatttca  ttatggtgaa  agttggaacc  tcttacgtgc
361  cgatcaacgt  ctcattttcg  ccagatateg  aattctaaag  atctggcacg  ttttacggct
421  agctcagtc  taggtatagt  gctagcatac  aagattataa  aaacaactca  gtgttttttt
481  ctttgaatga  tgtcgttctg  caactttggc  gagggacaga  gcgactcctt  tttatttggg
541  tcctaactcg  agtaaggatc  tccaggcacc  aaataaaaac  aaaggctcag  tcgaaagact
601  gggectttcg  ttttatctgt  tgtttgtcgg  tgaacgctct  ctactagagt  cacactggct
661  caccttcggg  tgggectttc  tgcgtttata  cctagggcgt  tcggctgcgg  cgagcgggat
721  cagctcactc  aaaggcggta  atacggttat  ccacagaatc  aggggataac  gcaggaaaga
781  acatgtgagc  aaaaggccag  caaaaggcca  ggaaccgtaa  aaaggcccg  ttgctggcgt
841  ttttccatag  gctccgcccc  cctgacgagc  atcacaaaaa  tcgacgctca  agtcagaggt
901  ggcgaaacce  gacaggacta  taaagatacc  aggcgtttcc  ccttggaaagc  tccctcgtgc
961  gctctcctgt  tccgacctgt  ccgcttaccg  gataacctgt  cgcctttctc  ccttcgggaa
1021  gcgtggcgct  ttctcatagc  tcacgctgta  ggtatctcag  ttcggtgtag  gtcgttcgct
1081  ccaagctggg  ctgtgtgcac  gaaccccccg  ttcagcccga  ccgctgcgcc  ttatccggta
1141  actatcgtct  tgagtccaac  ccggtaaagc  acgacttacc  gccactggca  gcagccactg
1201  gtaacaggat  tagcagagcg  aggtatgtag  gcggtgctac  agagtctctg  aagtgggtggc
1261  ctaactacgg  ctacactaga  aggacagtat  ttggtatctg  cgctctgctg  aagccagtta
1321  ccttcggaaa  aagagttggt  agctcttgat  ccggcaaaca  aaccaccgct  ggtagcgggtg
1381  gtttttttgt  ttgcaagcag  cagattacgc  gcagaaaaaa  aggatctcaa  gaagatcctt
1441  tgatcttttc  tacggggtct  gacgctcagt  ggaacgaaaa  ctcacgttaa  gggattttgg
1501  tcatgactag  tgcttggatt  ctcaccaata  aaaaacgccc  ggcggcaacc  gacgcttctg
1561  aacaaatcca  gatggagttc  tgaggtcatt  actggatcta  tcaacaggag  tccaagcgag
1621  ctcgatatca  aattacgccc  cgccttgcca  ctcatcgcag  tactgttgta  attcattaag
1681  cattctgccc  acatggaagc  catcacaaac  ggcatgatga  acctgaatcg  ccagcggcat
1741  cagcaccttg  tcgccttgcg  tataatattt  gcccatggtg  aaaacggggg  cgaagaagtt
1801  gtccatattg  gccacgttta  aatcaaaact  ggtgaaactc  acccagggat  tggctgagac
1861  gaaaaacata  ttctcaataa  accctttagg  gaaataggcc  aggttttcac  cgtaacacgc
1921  cacatcttgc  gaatatatgt  gtagaaactg  ccggaaatcg  tcgtggtatt  cactccagag
1981  cgatgaaaaa  gtttcagttt  gctcatggaa  aacggtgtaa  caagggtga

```

//

C.8 Inv Control: positive ind (pJH4-37)

```

LOCUS      pJH4-37      3946 bp      DNA      circular
DEFINITION

```

Originally pBbA8k-RFP from JBEI stock.

```

pBAD_mRFP1
pAraC_araC

```

FEATURES	Location/Qualifiers
CDS	complement(1524..2402) /gene="araC" /codon_start="0"
misc_feature	complement(587..1381) /gene="KanR" /note="encodes_nptII_(aka_AphA,_neoR),_kan_and_neo_re"
rep_origin	complement(3600..365) /gene="p15a"
CDS	2763..3440 /gene="mRFP1" /codon_start="0"
terminator	456..561 /gene="term_T0"
promoter	complement(2553..2581) /gene="pAraC"
promoter	2678..2705 /gene="pBAD"
misc_feature	2743..2762 /gene="RFP_RBS"
misc_feature	2632..2645 /gene="CAP_site"
terminator	3465..3544 /gene="term_rrnB" /note="Parts_Registry:_BBa_B0010"
terminator	3553..3593 /gene="term_T7" /note="Parts_Registry:_BBa_B0012"

BASE COUNT 995 a 1095 c 966 g 890 t

ORIGIN

```

1 tcagttccgg gtaggcagtt cgctccaagc tggactgtat gcacgaacce cccgttcagt
61 ccgaccgctg cgccttatcc ggtaactatc gtcttgagtc caaccgggaa agacatgcaa
121 aagcaccact ggcagcagcc actggtaatt gatttagagg agttagtctt gaagtcatgc
181 gccggttaag gctaaactga aaggacaagt tttggtgact gcgctctcc aagccagtta
241 cctcggttca aagagttggt agctcagaga accttcgaaa aaccgccttg caaggcggtt
301 ttttcgtttt cagagcaaga gattacgcgc agacaaaaac gatctcaaga agatcatctt
361 attaatcaga taaaatattt ctagatttca gtgcaattta tctcttcaa tgtagcacct
421 gaagtcagcc ccatacgata taagttgtta ctagtgcttg gattctcacc aataaaaaaac
481 gcccggcggc aaccgagcgt tctgaacaaa tccagatgga gttctgaggt cttactgga
541 tctatcaaca ggagtccaag cgagctctcg aaccccagag tcccgtcag aagaactcgt
601 caagaaggcg atagaaggcg atgcgctgcg aatcgggagc ggcgataccg taaagcacga
661 ggaagcggtc agcccattcg ccgccaagct cttcagcaat atcacgggta gccaacgcta
721 tgtcctgata gcggtccgcc acaccagcgc ggccacagtc gatgaatca gaaaagcggc
781 cattttccac catgatattc ggcaagcagg catcgccatg ggtcacgacg agatcctcgc
841 cgtcgggcat gcgcgccttg agcctggcga acagttcggc tggecgagc cctgatgct
901 cttcgtccag atcatcctga tcgacaagac cggcttccat ccgagtacgt gctcgtcga
961 tgcgatgttt cgcttggtgg tcgaatgggc aggtagccgg atcaagcgta tgcagccgcc
1021 gcattgcatc agccatgatg gatactttct cggcaggagc aaggtgagat gacaggagat
1081 cctgcccggc cacttcgccc aatagcagcc agtccccttc cgcttcagtg acaacgtcga
1141 gcacagctgc gcaaggaacg cccgtcgtgg ccagccacga tagccgcgct gcctcgtcct

```

```

1201 gcagttcatt cagggcaccg gacaggtegg tcttgacaaa aagaaccggg cgcccctgcg
1261 ctgacagccg gaacacggcg gcatcagagc agccgattgt ctgttgtgcc cagtcatagc
1321 cgaatagcct ctccacccaa gcggccggag aacctgctgt caatccatct tgttcaatca
1381 tgcgaaacga tctcctcct gtctcttgat cagatcatga tcccctgctc catcagatcc
1441 ttggcgccaa gaaagccatc cagtttactt tgcagggcct cccaacctta ccagagggcg
1501 ccccagctgg caattccgac gtcttatgac aacttgacgg ctacatcatt cactttttct
1561 tcacaaccgg cacggaacte gctcgggctg gccccggctc attttttaa tacccegag
1621 aatatagatt gatcgtcaaa accaacattg cgaccgacgg tggcgatagg catccgggtg
1681 gtgctcaaaa gcagcttegc ctggctgata cgttggtcct cgcgccagct taagacgcta
1741 atccctaact gctggcgcaa aagatgtgac agacgcgacg gcgacaagca aacatgctgt
1801 gcgacgctgg cgatatcaaa attgctgtct gccagggtgat cgtgatgta ctgacaagcc
1861 tcgctgtaacc gattatccat cgggtgatgg agcgactcgt taatcgcttc catgceccgc
1921 agtaacaatt gctcaagcag atttatcgcc agcagctccg aatagegcc ttccccttgc
1981 ccggcgttaa tgatttgccc aaacaggteg ctgaaatgcg gctggtgctc ttcattccggg
2041 cgaaagaacc ccgtattggc aatatattgac ggccagttaa gccattcatg ccagtagggc
2101 cgcggacgaa agtaaaccce ctggtgatac cattcgcgag cctccggatg acgaccgtag
2161 tgatgaatct ctctggcggg gaacagcaaa atatacccg gtcggcaaac aaattctcgt
2221 cctgattttt tcaccacccc ctgaccgcga atggtgagat tgagaatata acctttcatt
2281 cccagcggtc ggtcgataaa aaaatcgaga taaccgttgg cctcaatcgg cgttaaacce
2341 gccaccagat gggcattaaa cgagtatccc ggcagcaggg gatcattttg cgttcagcc
2401 ataactttca tactcccgcc attcagagaa gaaaccaatt gtccatattg catcagacat
2461 tgccgtcact gcgtctttta ctggctcttc tcgtaacca aaccggtaac cccgcttatt
2521 aaaagcattc tgtaacaaag cgggaccaa gcatgacaa aaacgcgtaa caaaagtgtc
2581 tataatcacg gcagaaaagt ccacattgat tatttgacg gcgtcacact ttgctatgcc
2641 atagcatttt tatccataag attageggat tctacctgac gctttttatc gcaactctct
2701 actgtttctc catacccggt ttttgggaa ttcaaaagat ctttaagaa ggagatatac
2761 atatggcgag tagcgaagac gttatcaaag agttcatgct tttcaaagtt cgtatggaag
2821 gttccgttaa cggtcacgag ttcgaaatcg aaggtgaagg tgaaggtcgt ccgtacgaag
2881 gtaccagac cgctaaactg aaagttacca aaggtggtcc gctgcccgtc gcttgggaca
2941 tctgtceccc gcagttccag tacggttcca aagcttacgt taaacaccg gctgacatcc
3001 cggactacct gaaactgtcc tcccgggaag gtttcaaag ggaacgtggt atgaacttcg
3061 aagacgggtg tggtgttacc gttaccagc actcctcctc gcaagacggt gagttcatct
3121 acaaagttaa actgcgtggt accaacttcc cgtccgacgg tccggttatg cagaaaaaaa
3181 ccatgggttg ggaagcttcc accgaacgta tgtaccgga agacggtgct ctgaaaggtg
3241 aatcaaaaat gcgtctgaaa ctgaaagacg gtggtcacta cgacgctgaa gttaaaacca
3301 cctacatggc taaaaaaccg gttcagctgc cgggtgctta caaaaccgac atcaaactgg
3361 acatcacctc ccacaacgaa gactacacca tcgttgaaca gtacgaacgt gctgaaggtc
3421 gtcactccac cgggtgcttaa ggatccaaac tcgagtaagg atctccagc atcaataaaa
3481 acgaaaggct cagtcgaaag actgggcctt tcgttttatc tgttgtttgt cgggtgaacgc
3541 tctctactag agtcacactg gctcaccttc gggtgggcct ttctgctttt atacctaggg
3601 atatatcccg ctctctcgtc cactgactcg ctacgctcgg tcgttcgact gcggcgagcg
3661 gaaatggctt acgaacgggg cggagatttc ctggaagatg ccaggaagat acttaacagg
3721 gaagtgagag ggccgcggca aagccgtttt tccataggct cgcceccctc gacaagcatc
3781 acgaaatctg acgctcaaat cagtgggtggc gaaaccgac aggactataa agataccagg
3841 cgtttccccc tggcggctcc ctctgtcgtc ctctgttcc tgctttcgg tttaccggtg
3901 tcattccgct gttatggcgc cgtttgtctc attccacgcc tgacac

```

//

C.9 LatInh Testing: las sender (pJH9-22)

LOCUS pJH9-22 2560 bp DNA circular

DEFINITION

pJH6-30 moved onto ColE1/CmR backbone.

pLtetO-1_lasI

FEATURES Location/Qualifiers

rep_origin	complement(2203..325) /gene="ColE1"
misc_feature	complement(452..1111) /gene="CmR"
CDS	1324..1932 /gene="lasI" /codon_start="0"
terminator	331..436 /gene="term_T0"
promoter	1244..1306 /gene="pLtetO-1"
prot_bind	1244..1262 /gene="tetO2"
prot_bind	1269..1287 /gene="tetO2"
misc_feature	1306..1317 /gene="RBS_BB_a_B0034" /note="Parts_Registry:_BB_a_B0034"
terminator	1941..2020 /gene="term_rrnB" /note="Parts_Registry:_BB_a_B0010"
terminator	2029..2069 /gene="term_T7" /note="Parts_Registry:_BB_a_B0012"

BASE COUNT 659 a 641 c 647 g 613 t

ORIGIN

```

1  ccactggcag cagccactgg taacaggatt agcagagcga ggtatgtagg cggtgctaca
61  gagttcttga agtgggtggc taactacggc tacactagaa ggacagtatt tggtatctgc
121 gctctgctga agccagttac cttcggaaaa agagttggta gctcttgatc cggcaaaaaa
181 accaccgctg gtagecgtgg tttttttgtt tgcaagcagc agattacgcg cagaaaaaaaa
241 ggatctcaag aagatccttt gatcttttct acgggggtctg acgctcagtg gaacgaaaac
301 tcacgttaag ggattttggt catgactagt gcttggattc tcaccaataa aaaacgcccg
361 gcggcaaccg agcgttctga acaaateccag atggagttct gaggtcatta ctggatctat
421 caacaggagt ccaagcgagc tcatatacaa attacgcccc gccctgccac tcatcgcagt
481 actgttgtaa ttcattaagc attctgccga catggaagcc atcaaaaacg gcatgatgaa
541 cctgaategc cagcggcate agcaccttgt cgccttgcgt ataataattg cccatggtga
601 aaacgggggc gaagaagttg tccatattgg ccacgtttaa atcaaaaactg gtgaaactca
661 cccagggtatt ggctgagacg aaaaacatat tctcaataaa cccttagggg aaataggcca

```

```

721 ggttttcacc gtaacacgcc acatcttgcg aatatatgtg tagaaactgc cggaaatcgt
781 cgtgggtatc actccagagc gatgaaaacg tttcagtttg ctcatggaaa acgggtgtaac
841 aagggtgaac actatcccat atcaccagct caccgtcttt cattgccata cgaaattccg
901 gatgagcatt catcaggcgg gcaagaatgt gaataaagge cggataaaac ttgtgcttat
961 ttttctttac ggtctttaaa aaggccgtaa tatccagctg aacggctctgg ttataggtac
1021 attgagcaac tgactgaaat gcctcaaaat gttctttacg atgccattgg gatatatcaa
1081 cgggtggata tccagtgatt tttttctcca ttttagcttc cttagctcct gaaaatctcg
1141 ataactcaaa aaatacgcgc ggtagtgate ttatttcatt atggtgaaag ttggaacctc
1201 ttacgtgccg atcaacgtct cattttcgcc agatategac gtctccctat cagtgataga
1261 gattgacatc cctatcagtg atagagatac tgagcactac tagagaaaga ggagaaatac
1321 tagatgatec taaaaattgg tcggcgcgaa gagttcgata aaaaactgct gggcgagatg
1381 cacaagttgc gtgctcaagt gttcaaggag cgcaaaggct gggacgttag tgtcatcgac
1441 gagatggaac tcgatggtta tgacgcactc agtccttatt acatgttgat ccaggaagat
1501 actcctgaag cccaggtttt cggttgctgg cgaattctcg ataccactgg cccctacatg
1561 ctgaagaaca ctttcccgga gcttctgcac ggcaaggaag cgccttgctc gccgcacatc
1621 tgggaactca gccgtttctc catcaactct ggacagaaag gctcgtctggg cttttccgac
1681 tgtacgccgg aggcgatgag cgcgctggcc cgctacagcc tgcagaacga catccagacg
1741 ctggtgacgg taaccaccgt aggcgtggag aagatgatga tccgtgccgg cctggacgta
1801 tcgcgcttcg gtccgcacct gaagatcggc atcgagcgcg cgggtggcctt gcgcacgaa
1861 ctcaatgcca agaccagat cgcgctttac gggggagtgct tgggtggaaca gcgactggcg
1921 gtttcatgat aatactagag ccaggcatca aataaaacga aaggctcagt cgaaagactg
1981 ggcccttctg tttatctggt gtttctcggt gaacgctctc tactagagtc aactggctc
2041 accttcgggt ggccctttct gcgctttata ctagggcggt cggctgcggc gagecggatc
2101 agctcactca aaggcggtaa tacggttata cacagaatca ggggataacg caggaaagaa
2161 catgtgagca aaaggccagc aaaaggccag gaaccgtaaa aaggccgcgt tgcctggcgtt
2221 tttccatagg ctccgcccc ctgacgagca tcacaaaaat cgacgctcaa gtcagaggtg
2281 gcgaaaccgg acaggactat aaagatacca ggcgcttccc cctggaagct cctcgtgctg
2341 ctctcctggt ccgaccctgc cgttaccgg atacctgtcc gcctttctcc cttcgggaag
2401 cgtggcgctt tctcatagct cacgctgtag gtatctcagt tcggtgtagg tcgttcgctc
2461 caagctgggc tgtgtgcacg aacccccgt tcagccccgac cgtcgcgctt tatccggtaa
2521 ctatcgtctt gagtccaacc cggtaagaca cgacttatcg

```

//

C.10 LatInh Testing: lux receiver Amp (pJH5-4)

LOCUS pJH5-4 3995 bp DNA circular
DEFINITION

pJH5-1 with Bujard 5-UTR on mRFP1 swapped out for that of pBbA8k-RFP.

pLtetO-1_luxR
pLuxI_mRFP1

FEATURES

	Location/Qualifiers
rep_origin	complement(2914..3596) /gene="ColE1"
misc_feature	complement(3696..556) /gene="AmpR"

```

CDS                899..1654
                   /gene="luxR"
                   /codon_start="0"
misc_feature       1655..1679
                   /gene="barcode"
                   /note="http://parts.igem.org/Help:Barcodes"
CDS                1937..2614
                   /gene="mRFP1"
                   /codon_start="0"
terminator         2636..2734
                   /gene="term_rrnD_T1"
promoter           819..881
                   /gene="pLtetO-1"
prot_bind          819..837
                   /gene="tetO2"
prot_bind          844..862
                   /gene="tetO2"
promoter           1825..1879
                   /gene="pLuxI"
prot_bind          1825..1843
                   /gene="lux_box"
misc_feature       881..892
                   /gene="RBS_BBa_B0034"
                   /note="Parts_Registry:_BBa_B0034"
terminator         1688..1767
                   /gene="term_rrnB"
                   /note="Parts_Registry:_BBa_B0010"
terminator         1776..1816
                   /gene="term_T7"
                   /note="Parts_Registry:_BBa_B0012"
BASE COUNT        1168 a    903 c    880 g    1044 t
ORIGIN
1  ttgccattgc tacaggcac gtggtgtcac gctcgtcgtt tggatggct tcattcagct
61  ccggttccca acgatcaagg cgagttacat gatcccccac gttgtgcaaa aaagcgggta
121 gctccttcgg tctcctcgatc gttgtcagaa gtaagtggc cgcagtgtta tcaactatgg
181 ttatggcagc actgcataat tctcttactg tcatgccatc cgtaaatgac ttttctgtga
241 ctggtgagta ctcaaccaag tcattctgag aatagtgtat gcggcgaccg agttgctctt
301 gccggcgctc aatacgggat aataccgcgc cacatagcag aactttaaaa gtgctcatca
361 ttggaaaacg ttcttcgggg cgaaaactct caaggatctt accgctgttg agatccagtt
421 cgatgtaacc cactcgtgca cccaactgat cttcagcatc ttttactttc accagcgttt
481 ctgggtgagc aaaaacagga aggcaaaaatg ccgcaaaaaa gggaataagg gcgacacgga
541 aatggtgaa actcatactc ttcctttttc aatattattg aagcatttat cagggttatt
601 gtctcatgag cggatacata tttgaatgta tttagaaaaa taaacaaata ggggttccgc
661 gcacatttcc ccgaaaagtg ccacctgagc tctaagaaac cattattatc atgacattaa
721 cctataaaaa taggcgtatc acgaggcaga atttcagata aaaaaaatcc ttagctttcg
781 ctaagcatga tttctggaat tcgcggccgc ttttagagtc cctatcagtg atagcattg
841 acatccctat cagtgataga gatactgagc actactagag aaagaggaga aatactagat
901 gaaaaacata aatgccgacg acacatacag aataattaat aaaattaaag cttgtagaag
961 caataatgat attaatacat gcttatctga tatgactaaa atggtacatt gtgaatatta
1021 tttactcgcg atcatttate ctcattctat ggttaaactc gatatttcaa tccatagataa

```

```

1081 ttaccctaaa  aatggagge  aatattatga  tgacgctaat  ttaataaaat  atgatectat
1141 agtagattat  tctaactcca  atcattcacc  aattaattgg  aatataattg  aaaacaatgc
1201 tgtaaataaa  aaatctccaa  atgtaattaa  agaagegaaa  acatcagggtc  ttatcactgg
1261 gtttagtttc  cctattcata  cggctaacaa  tggcttcgga  atgcttagtt  ttgcacattc
1321 agaaaaagac  aactatatag  atagttttatt  ttacatgcg  tgtatgaaca  taccattaat
1381 tgttccttct  ctagttagata  attatcgaaa  aataaatata  gcaaataata  aatcaaacia
1441 cgatttaacc  aaaagagaaa  aagaatgttt  agcgtgggca  tgcgaaggaa  aaagctcttg
1501 ggatatttca  aaaatattag  gttgcagtga  gcgtactgtc  actttccatt  taaccaatgc
1561 gcaaatgaaa  ctcaatacaa  caaacgcgtg  ccaaagtatt  tctaaagcaa  ttttaacagg
1621 agcaattgat  tgcccatact  ttaaaaatta  ataactgta  tagtgctagt  gtagatcact
1681 actagagcca  ggcatcaaat  aaaacgaaag  gctcagtcga  aagactgggc  ctttcgtttt
1741 atctgtttgt  tctcggtgaa  cgtctcttac  tagagtcaca  ctggctcacc  ttcgggtggg
1801 ctttctgctg  tttatatact  agagacctgt  aggatcgtac  aggtttacgc  aagaaaatgg
1861 tttgttatag  tcgaataaat  tctccatacc  cgtttttttg  ggaattcaaa  agatctttta
1921 agaaggagat  atacatatgg  cgagtagega  agacgttata  aaagagttca  tgcgtttcaa
1981 agttcgtatg  gaaggttccg  ttaacggtca  cgagttcgaa  atcgaagggtg  aaggtgaagg
2041 tegtccgtac  gaaggtaccc  agaccgctaa  actgaaagtt  accaaagggtg  gtccgctgcc
2101 gttcgttgg  gacatcctgt  ccccgcagtt  ccagtacggt  tccaaagctt  acgttaaaca
2161 cccggtgac  atcccggact  acctgaaact  gtcttcccg  gaaggtttca  aatgggaacg
2221 tgttatgaac  ttcgaagacg  gtggtgttgt  taccgttacc  caggactcct  cctgcaaga
2281 cggtagattc  atctacaaag  ttaaactgct  tggtagcaac  tcccgtccg  acgggtccgt
2341 tatgcagaaa  aaaaccatgg  gttgggaagc  tccaccgaa  cgtatgtacc  cggaagacgg
2401 tgctctgaaa  ggtgaaatca  aaatgctct  gaaactgaaa  gacggtggtc  actacgacgc
2461 tgaagttaaa  accacctaca  tggctaaaaa  accggttcag  ctgcccgggtg  cttaaaaaac
2521 cgacatcaaa  ctggacatca  cctcccacia  cgaagactac  accatcgttg  aacagtacga
2581 acgtgctgaa  ggtcgtcact  ccaccggtgc  ttaataataa  ggatctcagg  tctcatgatg
2641 ggaactgcca  gacatcaaat  aaaacaaaag  gctcagtcgg  aagactgggc  cttttgtttt
2701 atctgtttgt  tctcggtgaa  cactctccc  ggcgtactag  tagcggccgc  tgcaggcttc
2761 ctcgctcact  gactcgtcgc  gctcggctgt  tccgctgctg  cgagcgggtat  cagctcactc
2821 aaaggcggta  atacggttat  ccacagaate  aggggataac  gcaggaaaga  acatgtgagc
2881 aaaaggccag  caaaaggcca  ggaaccgtaa  aaaggccgcg  ttgctggcgt  tttccacag
2941 gctccgcccc  cctgacgagc  atcaaaaaa  tcgacgctca  agtcagaggt  ggcaaaacce
3001 gacaggacta  taaagatacc  aggcgtttcc  cctggaagc  tccctcgtgc  gctctcctgt
3061 tccgacctg  ccgcttaccg  gatacctgtc  cgcctttctc  cttcgggaa  gcgtggcgtc
3121 ttctcatagc  tcacgctgta  ggtatctcag  ttcggtgtag  gtcgttcgct  ccaagctggg
3181 ctgtgtgcac  gaaccccccg  ttcagccccg  ccgctgcgcc  ttatccggtg  actatcgtct
3241 tgagtccaac  ccggtaaagc  acgacttata  gccactggca  gcagccactg  gtaacaggat
3301 tagcagagcg  aggtatgtag  gcggtgctac  agagttcttg  aagtgggtgg  ctaactacgg
3361 ctacactaga  agaacagtat  ttggtatctg  cgtctgctg  aagccagttg  cttcggaaa
3421 aagagttggt  agctcttgat  ccggcaaaa  aaccaccgct  ggtagcgggtg  gttttttgt
3481 ttgcaagcag  cagattacgc  gcagaaaaa  aggatctcaa  gaagatcctt  tgatctttc
3541 tacggggtct  gacgctcagt  ggaacgaaaa  ctcacgttaa  gggattttg  tcatgagatt
3601 atcaaaaagg  atcttcacct  agatcctttt  aaattaaaaa  tgaagtttta  aatcaatcta
3661 aagtatatat  gagtaaactt  ggtctgacag  ttaccaatgc  ttaatcagtg  aggcacctat
3721 ctcagcgatc  tgtctatttc  gttcaccat  agttgctga  ctecccgctg  ttagataaac
3781 tacgatacgg  gagggcttac  catctggccc  cagtgtgca  atgataccgc  gagaccacg
3841 ctcaccggct  ccagatttat  cagcaataaa  ccagccagcc  ggaagggccg  agcgcagaag
3901 tggctctgca  actttatecg  cctccatcca  gtctattaac  tgttgccggg  aagctagagt
3961 aagtagtctg  ccagttaata  gtttgcgcaa  cgttg

```

//

C.11 LatInh Testing: reporter Amp (pJH5-29)

LOCUS pJH5-29 3046 bp DNA circular
DEFINITION

pJH5-4 with LuxR operon removed. Can be used as a 'LuxRA_receiver.'

pLuxI_mRFP1

FEATURES Location/Qualifiers
rep_origin complement(1965..2647)
/gene="ColE1"
misc_feature complement(2747..556)
/gene="AmpR"
CDS 988..1665
/gene="mRFP1"
/codon_start="0"
terminator 1687..1785
/gene="term_rrnD_T1"
promoter 876..930
/gene="pLuxI"
prot_bind 876..894
/gene="lux_box"
terminator 827..867
/gene="term_T7"
/note="Parts_Registry:_BBa_B0012"

BASE COUNT 810 a 747 c 727 g 762 t

ORIGIN

```

1 ttgccattgc tacaggcatc gtggtgtcac gctcgtcgtt tggatatggct tcattcagct
61 ccggttccca acgatcaagg cgagttacat gatcccccat gttgtgcaaa aaagcggtta
121 gctccttcgg tctcggatc gttgtcagaa gtaagttggc cgcagtgtta tcaactcatgg
181 ttatggcagc actgcataat tctcttactg tcatgccatc cgtaagatgc ttttctgtga
241 ctggtgagta ctcaaccaag tcattctgag aatagtgtat gggcgaccg agttgctctt
301 gcccggcgtc aatacgggat aataccgcgc cacatagcag aactttaaaa gtgctcatca
361 ttggaaaacg ttcttcgggg cgaaaactct caaggatctt accgctgttg agatccagtt
421 cgatgtaacc cactcgtgca cccaactgat cttcagcatc ttttactttc accagcgttt
481 ctgggtgagc aaaaacagga aggcaaaatg ccgcaaaaaa gggaataagg gcgacacgga
541 aatgttgaat actcatactc ttcctttttc aatattattg aagcatttat cagggttatt
601 gtctcatgag cggatacata tttgaatgta tttagaaaaa taaacaaata ggggttccgc
661 gcacatttcc cggaaaagtg ccacctgacg tctaagaaac cattattatc atgacattaa
721 cctataaaaa taggcgtatc acgaggcaga atttcagata aaaaaaatcc ttagctttcg
781 ctaaggatga tttctggaat tcgcggccgc ttctagagta ctagagtcac actggctcac
841 cttegggtgg gcctttctgc gtttatatac tagagacctg taggatcgta caggtttaagc
901 caagaaaatg gtttgttata gtcgaataaa ttctccatac ccgttttttt gggaattcaa
961 aagatctttt aagaaggaga tatacatatg gcgagtagcg aagacgttat caaagagttc
1021 atgcgtttca aagttcgtat ggaagggtcc gttaacggtc acgagttcga aatcgaaggt
1081 gaaggtgaag gtcgtccgta cgaaggtacc cagaccgcta aactgaaagt taccaaaggt
1141 ggtccgctgc cgttcgttg ggacatcctg tccccgcagt tccagtaacgg ttccaaagct

```

```

1201 tacgttaaac acccggctga cateccggac tacctgaaac tgtcettccc ggaaggtttc
1261 aaatgggaac gtgttatgaa cttcgaagac ggtgggtgtt ttaccgttac ccaggactcc
1321 tccctgcaag acgggtgagtt catctacaaa gttaaactgc gtggtaccaa cttcccgtcc
1381 gacggctccgg ttatgcagaa aaaaaccatg ggttgggaag cttccaccga acgtatgtac
1441 ccggaagacg gtgctctgaa aggtgaaatc aaaatgcgtc tgaaactgaa agacggtggt
1501 cactacgacg ctgaagttaa aaccacctac atggctaaaa aaccggttca gctgccgggt
1561 gcttacaaaa ccgacatcaa actggacatc acctcccaca acgaagacta caccatcgtt
1621 gaacagtagc aacgtgctga aggtcgctac tccaccgggt ctttaataata aggatctcag
1681 gtctcatgat gggaactgcc agacatcaaa taaaacaaaa ggctcagtcg gaagactggg
1741 ccttttgttt tatctgttgt ttgtcgggtg acactctccc gggcgtaacta gtageggccc
1801 ctgcaggctt cctcgtctac tgactcgtcg cgctcggtcg ttcggctgcg gcgagcggta
1861 tcagctcact caaaggcgggt aatacggtta tccacagaat caggggataa cgcaggaaag
1921 aacatgtgag caaaaggcca gcaaaaggcc aggaaccgta aaaaggccgc gttgctggcg
1981 tttttccaca ggctccgccc cctgacgag catcacaaaa atcgacgctc aagtcagagg
2041 tggcgaaacc cgacaggact ataaagatac caggcgtttc ccctggaag ctccctcgtg
2101 cgctctcctg ttcgacctt gccgcttacc ggatacctgt ccgcttttct cccttcggga
2161 agcgtggegc tttctcatag ctcacgctgt aggtatctca gttcgggtgta ggtcgttcgc
2221 tccaagctgg gctgtgtgca cgaaccccc gttcagcccc accgctgcgc cttatccggt
2281 aactatcgtc ttgagtccaa cccggtaaaga cagcacttat cgcactggc agcagccact
2341 ggtaacagga ttagcagagc gaggtatgta ggcggtgcta cagagttctt gaagtggtagg
2401 cctaactacg gctacactag aagaacagta tttggtatct gcgctctgct gaagccagtt
2461 accttcggaa aaagagttgg tagctcttga tccggcaaac aaaccaccgc tggtagcgggt
2521 ggtttttttg tttgcaagca gcagattacg cgcagaaaaa aaggatctca agaagatect
2581 ttgatctttt ctacggggtc tgacgctcag tggaaacgaaa actcagttta agggattttg
2641 gtcattgagat tatcaaaaag gatcttcacc tagatccttt taaattaaaa atgaagtttt
2701 aatcaatct aaagtatata tgagtaaact tggcttgaca gttaccaatg cttaatcagt
2761 gaggcacctt tctcagcgat ctgtctatct cgttcatcca tagttgctg actceccgctc
2821 gtgtagataa ctacgatacg ggagggctta ccatctggcc ccagtgtgc aatgataccg
2881 cgagaccac gctcaccggc tccagattta tcagcaataa accagccagc cggaagggcc
2941 gagegcagaa gtggctctgc aactttatcc gcctccatcc agtctattaa ctgttgccgg
3001 gaagctagag taagtagttc gccagttaat agtttgcgca acgttg

```

//

C.12 LatInh Testing: lux receiver Cm (pJH9-35)

LOCUS pJH9-35 3650 bp DNA circular
DEFINITION

pJH5-4 moved onto ColE1/CmR backbone.

pLtetO-1_luxR
pLuxI_mRFP1

FEATURES

	Location/Qualifiers
rep_origin	complement(3293..325) /gene="ColE1"
misc_feature	complement(452..1111) /gene="CmR"

```

CDS                1324..2079
                   /gene="luxR"
                   /codon_start="0"
misc_feature       2080..2104
                   /gene="barcode"
                   /note="http://parts.igem.org/Help:Barcodes"
CDS                2362..3039
                   /gene="mRFP1"
                   /codon_start="0"
terminator        331..436
                   /gene="term_T0"
terminator        3061..3159
                   /gene="term_rrnD_T1"
promoter          1244..1306
                   /gene="pLtetO-1"
prot_bind         1244..1262
                   /gene="tetO2"
prot_bind         1269..1287
                   /gene="tetO2"
promoter          2250..2304
                   /gene="pLuxI"
prot_bind         2250..2268
                   /gene="lux_box"
misc_feature       1306..1317
                   /gene="RBS_BBa_B0034"
                   /note="Parts_Registry:_BBa_B0034"
terminator        2113..2192
                   /gene="term_rrnB"
                   /note="Parts_Registry:_BBa_B0010"
terminator        2201..2241
                   /gene="term_T7"
                   /note="Parts_Registry:_BBa_B0012"
BASE COUNT        1088 a      815 c      802 g      945 t
ORIGIN
   1 ccactggcag cagccactgg taacaggatt agcagagcga ggtatgtagg cggtgctaca
  61 gagttcttga agtggtggcc taactacggc tacactagaa ggacagtatt tggtatctgc
 121 gctctgctga agccagttac cttecgaaaa agagttggta gctcttgatc cggcaaaaaa
 181 accaccgctg gtagcgggtg tttttttggt tgcaagcagc agattacgcg cagaaaaaaaa
 241 ggatctcaag aagatccttt gatcttttct acgggggtctg acgctcagtg gaacgaaaac
 301 tcacgttaag ggattttggt catgactagt gcttggattc tcaccaataa aaaacgcccg
 361 gcggcaaccg agcgttctga acaaatccag atggagtctt gaggtcatta ctggatctat
 421 caacaggagt ccaagcgagc tcgatatcaa attacgcccc gcctgcccac tcatcgcagt
 481 actgttgtaa ttcattaagc attctgccga catggaagcc atcaciaaac gcatgatgaa
 541 cctgaatcgc cagcggcatc agcaccttgt cgccttgctg ataataattg cccatgggtga
 601 aaacggggggc gaagaagttg tccatatggg ccacgtttaa atcaaaactg gtgaaactca
 661 cccaggggatt ggctgagacg aaaaacatat tctcaataaa cccttagggg aaataggcca
 721 ggttttcacc gtaacacgcc acatcttgcg aatatatgtg tagaaactgc cggaaatcgt
 781 cgtgggtatc actccagagc gatgaaaacg tttcagtttg ctcatggaaa acgggtgtaac
 841 aaggggtgaac actatcccat atcaccagct caccgtcttt cattgccata cgaaatcccg
 901 gatgagcatt catcaggcgg gcaagaatgt gaataaaggc cggataaaac ttgtgcttat

```

```

961 ttttctttac ggtctttaa aaggccgtaa tatccagctg aacggctctgg ttataggtac
1021 attgagcaac tgactgaaat gectcaaaat gttctttacg atgccattgg gatatatcaa
1081 cggtggtata tccagtgatt tttttctcca ttttagcttc cttagctcct gaaaatctcg
1141 ataactcaaa aaatacgccc ggtagtgatc ttatttcatt atggtgaaag ttggaacctc
1201 ttacgtgccg atcaacgtct cattttcgcc agatategac gtctccctat cagtgataga
1261 gattgacatc cctatcagtg atagagatac tgagcactac tagagaaaga ggagaaatac
1321 tagatgaaaa acataaatgc cgacgacaca tacagaataa ttaataaaa taaagcttgt
1381 agaagcaata atgatattaa tcaatgctta tctgatatga ctaaaatggt acattgtgaa
1441 tattatttac tcgcatcat ttatcctcat tctatggta aatctgatat ttcaatccta
1501 gataattacc ctaaaaaatg gaggcaatat tatgatgacg ctaatttaat aaaatatgat
1561 cctatagtag attattctaa ctccaatcat tcaccaatta attggaatat atttgaaac
1621 aatgctgtaa ataaaaaatc tccaaatgta attaaagaag cgaaaacatc aggtcttate
1681 actgggttta gtttccctat tcatacggct aacaatggct tcggaatgct tagttttgca
1741 cattcagaaa aagacaacta tatagatagt ttatttttac atgcgtgtat gaacatacca
1801 ttaattgttc cttctctagt tgataattat cgaaaaataa atatagcaaa taataaatca
1861 aacaacgatt taaccaaag agaaaaagaa tgtttagcgt gggcatgcga aggaaaaagc
1921 tcttgggata tttcaaaaat attaggttgc agtgagcgta ctgtcacttt ccatttaacc
1981 aatgcgcaaa tgaaactcaa tacaacaac cgctgccaaa gtatttctaa agcaatttta
2041 acaggagcaa ttgattgccc ataactttaa aattaataac actgatagtg ctagtgtaga
2101 tcaactactag agccaggcat caaataaaac gaaaggctca gtcgaaagac tgggccttct
2161 gttttatctg ttgtttgtcg gtgaacgctc tctactagag tcacactggc tcacctcgg
2221 gtgggccttt ctgctgttat ataactagaga cctgtaggat cgtacagggt tacgcaagaa
2281 aatggtttgt tatagtcgaa taaattctcc ataccctgtt ttttgggaat tcaaaagatc
2341 ttttaagaag gagatataca tatggcgagt agcgaagacg ttatcaaaga gttcatgcgt
2401 ttcaaagttc gtatggaagg ttccgttaac ggtcacgagt tcgaaatcga aggtgaaggt
2461 gaaggtcgtc cgtacgaagg taccagacc gctaaactga aagttacca aggtggtccg
2521 ctgcccgttcg cttgggacat cctgtccccg cagttccagt acggttccaa agcttacgtt
2581 aaacaccggg ctgacatecc ggactacctg aaactgtcct tcccgggaagg ttcaaatgg
2641 gaacgtgtta tgaacttcga agacgggtgg gtgtttaccg ttaccagga ctctcctcg
2701 caagacggtg agttcatcta caaagttaaa ctgctgtgta ccaacttccc gtccgacggt
2761 ccggttatgc agaaaaaac catgggttgg gaagcttcca ccgaacgtat gtaccggaa
2821 gacggtgctc tgaaaggtga aatcaaaatg cgtctgaaac tgaaagacgg tggctactac
2881 gacgctgaag ttaaaaccac ctacatggct aaaaaaccgg ttcagctgcc gggtgcttac
2941 aaaaccgaca tcaaaactgga catcacctcc cacaacgaag actacaccat cgttgaacag
3001 tacgaacgtg ctgaaggtcg tcactccacc ggtgcttaat aataaggate tcaggtctca
3061 tgatgggaac tgccagacat caaataaaac aaaaggctca gtcggaagac tgggcctttt
3121 gttttatctg ttgtttgtcg gtgaacactc tcccgggcgc ctagggcgtt cggctgcggc
3181 gagecgtatc agctcactca aaggcggtaa tacggttate cacagaatca ggggataacg
3241 caggaaagaa catgtgagca aaaggccagc aaaaggccag gaaccgtaaa aaggccgcgt
3301 tgctggcggt tttccatagg ctccgcccc ctgacgagca tcacaaaaat cgacgctcaa
3361 gtcagaggtg gcgaaaccgg acaggactat aaagatacca ggcgtttccc cctggaagct
3421 cctcgtgceg ctctcctgtt ccgacctgc cgttaccgg atacctgtcc gcctttctcc
3481 ctccgggaag cgtggcgctt tctcatagct cacgctgtag gtatctcagt tcggtgtagg
3541 tcgttcgctc caagctgggc tgtgtgcacg aacccccctg tcagcccgac cgctgcgcct
3601 tatccggtaa ctatcgtctt gagtccaacc cggtaagaca cgacttatcg

```

//

C.13 LatInh Plasmid 1: reporter Cm (pJH4-22)

LOCUS pJH4-22 3347 bp DNA circular
DEFINITION

New plasmid based off of pBbE2c-RFP for microfluidic CDI experiments.

pLuxI_tetR_LVAdegtag_mRFP1

FEATURES	Location/Qualifiers
rep_origin	complement(2990..325) /gene="ColE1"
misc_feature	complement(452..1111) /gene="CmR"
CDS	1338..2000 /gene="tetR_with_deg_tag" /codon_start="0"
misc_feature	1962..1994 /gene="LVA_deg_tag"
CDS	2026..2703 /gene="mRFP1" /codon_start="0"
terminator	331..436 /gene="term_T0"
promoter	1244..1298 /gene="pLuxI"
prot_bind	1244..1262 /gene="lux_box"
misc_feature	1304..1337 /gene="UTR_(rbs5000)"
misc_feature	2006..2025 /gene="RFP_RBS"
terminator	2728..2807 /gene="term_rrnB" /note="Parts_Registry:_BBa_B0010"
terminator	2816..2856 /gene="term_T7" /note="Parts_Registry:_BBa_B0012"

BASE COUNT 949 a 776 c 792 g 830 t

ORIGIN

```

1 ccaactggcag cagccactgg taacaggatt agcagagcga ggtatgtagg cggtgctaca
61 gagttcttga agtgggtggc taactacggc tacactagaa ggacagtatt tggtatctgc
121 gctctgctga agccagttac ctteggaaaa agagttggta gctcttgatc cggcaaacaa
181 accaccgctg gtagcgggtg tttttttgtt tgcaagcagc agattacgcg cagaaaaaaaa
241 ggatctcaag aagatccttt gatcttttct acgggggtctg acgctcagtg gaacgaaaac
301 tcacgttaag ggattttggt catgactagt gcttggattc tcaccaataa aaaacgcccc
361 gcggaaccg agcgttctga acaaatecag atggagttct gaggtcatta ctggatctat
421 caacaggagt ccaagcgagc tcatatcaaa attacgcccc gcctgccac tcategcagt

```

```

481 actgttgtaa ttcattaage attctgccga catggaagcc atcacaacg gcgatgaa
541 cctgaategc cagcggcate agcaccttgt cgccttgcgt ataataattg cccatggtga
601 aaacgggggc gaagaagttg tccatattgg ccacgtttaa atcaaaactg gtgaaactca
661 cccagggatt ggctgagacg aaaaacatat tctcaataaa cccttagggg aaataggcca
721 ggttttcacc gtaacacgcc acatcttgcg aatataatgt tagaaactgc cggaaactgt
781 cgtgggtatc actccagagc gatgaaaacg tttcagtttg ctcatggaaa acgggtgtaac
841 aagggtgaac actatcccat atcaccagct caccgtcttt cattgccata cgaaattccg
901 gatgagcatt catcaggcgg gcaagaatgt gaataaaggc cggataaaac ttgtgcttat
961 ttttctttac ggtctttaaa aaggccgtaa tatccagctg aacggctctgg ttataggtac
1021 attgagcaac tgactgaaat gcctcaaaat gttctttacg atgccattgg gatataatca
1081 cgggtgtata tccagtgatt tttttctcca ttttagcttc cttagctcct gaaaatctcg
1141 ataactcaa aaatacgcgc ggtagtgatc ttatcttatt atggtgaaag ttggaacctc
1201 ttacgtgccg atcaacgtct cttttctgcc agatategac gtcacctgta ggatcgtaca
1261 ggtttacgca agaaaatggt ttgttatagt cgaataaaga tctaggaaaa agctcatata
1321 actagagtaa gaggtcaatg atgtctagat tagataaaag taaagtgatt aacagcgcac
1381 tagagctgct taatgaggtc ggaatcgaag gtttaacaac ccgtaaactc gcccagaagc
1441 taggtgtaga gcagcctaca ttgtattggc atgtaaaaaa taagcgggct ttgctcgagc
1501 ccttagccat tgagatgtta gataggcacc ataactcaact ttgcccttta gaaggggaaa
1561 gctggcaaga ttttttacgt aataacgcta aaagttttag atgtgcttta ctaagtcatc
1621 gcgatggagc aaaagtacat ttaggtacac ggcctacaga aaaacagtat gaaactctcg
1681 aaaatcaatt agccttttta tgccaacaag gtttttact agagaatgca ttatatgcac
1741 tcagcgtctg ggggcatttt acttttaggtt gcgtattgga agatcaagag catcaagtgc
1801 ctaaagaaga aagggaacaa cctactactg atagtatgcc gccattatta cgacaagcta
1861 tcgaattatt tgatcaccaa ggtgcagagc cagccttctt attcggcctt gaattgatca
1921 tatgcggatt agaaaaacaa cttaaatgtg aaagtgggtc tgctgcaaac gacgaaaact
1981 acgcttttagt agcttaataa gatcttttaa gaaggagata tacatatggc gtagtagcga
2041 gacgttatca aagagtcatc gcgtttcaaa gttcgtatgg aaggttccgt taacggtcac
2101 gagttcgaaa tcgaaggtga aggtgaaggt cgtccgtacg aaggtacca gaccgctaaa
2161 ctgaaagtta ccaaaggtgg tccgctgccg ttcgctggg acatcctgtc cccgcagttc
2221 cagtacggtt ccaaagctta cgttaaacac ccggctgaca tcccggacta cctgaaactg
2281 tcttcccggg aaggtttcaa atgggaacgt gttatgaact tcgaagacgg tgggtgtgtt
2341 accgttacc caggactctc cctgcaagac ggtgagttca tctacaaagt taaactgcgt
2401 ggtaccaact tcccgtccga cggctccggt atgcagaaaa aaaccatggg ttgggaagct
2461 tccaccgaac gtatgtacc ggaagacggt gctctgaaag gtgaaatcaa aatgcgtctg
2521 aaactgaaag acggtggtca ctacgacgct gaagttaaaa ccacctacat ggctaaaaaa
2581 ccggttcagc tgccgggtgc ttacaaaacc gacatcaaac tggacatcac ctcccacaac
2641 gaagactaca ccatcgttga acagtacgaa cgtgctgaag gtcgtcactc caccggtgct
2701 taagatcca aactcgagta aggatctcca ggcatacaat aaaacgaaag gctcagtcga
2761 aagactgggc ctttctgttt atctgttgtt tctcggtgaa cgctctctac tagagtcaca
2821 ctggctcacc ttcgggtggg ctttctgcg tttataccta gggcgttcgg ctgcggcgag
2881 cggatcagc tcaactaaag gcggtataac ggttatccac agaatcaggg gataacgcag
2941 gaaagaacat gtgagcaaaa ggccagcaaa aggccaggaa ccgtaaaaag gccgcgttgc
3001 tggegttttt ccataggctc cccccctg acgagcatca caaaaatcga cgtcaagtc
3061 agaggtggcg aaaccgcaca ggactataaa gataccaggc gtttccccct ggaagctccc
3121 tegtgcgctc tctgttccg acctgcccgc ttaccggata cctgtcccct tttctccctt
3181 cgggaagcgt ggcgctttct catagctcac gctgtaggta tctcagttcg gtgtaggtcg
3241 ttcgctccaa gctgggctgt gtgcacgaac cccccgttca gcccgaccgc tgcgcttat
3301 ccggttaacta tegtcttgag tccaaccggg taagacacga cttatcg

```

//

C.14 LatInh Plasmid 2: cell type B (pJH5-66)

LOCUS pJH5-66 3998 bp DNA circular
DEFINITION

pJH5-52 with LuxRmod replaced with LasR. Used with pJH4-22 as Cell B type.

pLtetO-1_luxI
pFAB46_lasR

FEATURES	Location/Qualifiers
rep_origin	complement(3652..365) /gene="p15a"
misc_feature	complement(587..1381) /gene="KanR" /note="encodes_nptII_(aka_AphA,_neoR),_kan_and_neo_re"
CDS	2773..3492 /gene="lasR" /codon_start="0"
CDS	complement(1765..2382) /gene="luxI" /codon_start="0"
misc_feature	complement(1771..1803) /gene="LVA_deg_tag"
terminator	456..561 /gene="term_T0"
promoter	complement(2419..2492) /gene="pLtetO-1"
prot_bind	complement(2474..2492) /gene="tetO2"
prot_bind	complement(2449..2467) /gene="tetO2"
promoter	2700..2746 /gene="apFAB46" /note="BIOFAB_promoter"
terminator	complement(1570..1743) /gene="term_TSAL2"
terminator	1524..1569 /gene="term_RNAI"
misc_feature	2747..2772 /gene="5'-UTR_BBa_B0034" /note="Parts_Registry:_BBa_B0034"
terminator	3517..3596 /gene="term_rrnB" /note="Parts_Registry:_BBa_B0010"
terminator	3605..3645 /gene="term_T7" /note="Parts_Registry:_BBa_B0012"

BASE COUNT	995 a	1047 c	965 g	991 t		
ORIGIN						
1	tcagttccgg	gtaggcagtt	cgctccaagc	tggactgtat	gcacgaacce	cccgttcagt
61	ccgaccgctg	cgccttatec	ggtaactatc	gtcttgagtc	caaccgggaa	agacatgcaa
121	aagcaccact	ggcagcagcc	actggtaatt	gatttagagg	agttagtctt	gaagtcatgc
181	gccggttaag	gctaaactga	aaggacaagt	tttggtgact	gcgctctctc	aagccagtta
241	cctcggttca	aagagttggt	agctcagaga	accttcgaaa	aaccgccttg	caaggcggtt
301	ttttcgtttt	cagagcaaga	gattacgcgc	agacaaaac	gatctcaaga	agatcatctt
361	attaatcaga	taaaatattt	ctagatttca	gtgcaattta	tctcttcaa	tgtagcacct
421	gaagtcagcc	ccatacgata	taagttgtta	ctagtgcctg	gattctcacc	aataaaaaac
481	gccccggcgc	aaccgagcgt	tctgaacaaa	tccagatgga	gttctgaggt	cattactgga
541	tctatcaaca	ggagtccaag	cgagctctcg	aaccccagag	tcccgcctag	aagaactcgt
601	caagaaggcg	atagaaggcg	atgcgctgcg	aatcgggagc	ggcgataccg	taaagcacga
661	ggaagcggtc	agcccattcg	ccgccaagct	cttcagcaat	atcacgggta	gccaacgcta
721	tgtcctgata	gcggctccgc	acaccagcc	ggccacagtc	gatgaatcca	gaaaagcggc
781	cattttccac	catgatattc	ggcaagcagg	catcgccatg	ggtcacgacg	agatectcgc
841	cgtegggcat	gcgcgccttg	agcctggcga	acagttcggc	tggcgcgagc	ccctgatgct
901	cttcgtccag	atcctcctga	tcgacaagac	cggcttccat	ccgagtacgt	gctcgtctga
961	tgcgatgttt	cgcttggtgg	tcgaatgggc	aggtagccgg	atcaagcgta	tgcagccgcc
1021	gcattgcate	agccatgatg	gatactttct	cggcaggagc	aaggtgagat	gacaggagat
1081	cctgccccgg	cacttcgccc	aatagcagcc	agtccttcc	cgcttcagtg	acaacgtcga
1141	gcacagctgc	gcaaggaacg	cccgtcgtgg	ccagccacga	tagecgcgct	gcctcgtcct
1201	gcagttcatt	cagggcaccg	gacaggtcgg	tcttgacaaa	aagaaccggg	cgccccgtcg
1261	ctgacagccg	gaacacggcg	gcatcagagc	agccgattgt	ctgttgtgcc	cagtcatagc
1321	cgaatagcct	ctccacccaa	gcggccggag	aacctgcgtg	caatccatct	tgttcaatca
1381	tgcgaaaega	tctcctcct	gtctcttgat	cagatcatga	tcccctgcgc	catcagatcc
1441	ttggcggcaa	gaaagccate	cagtttactt	tgcagggcct	cccaacctta	ccagagggcg
1501	ccccagctgg	caattccgac	gtegatccgg	caaacaaacc	accgttggtta	gcgggtggttt
1561	ttttgtttgg	atcgacaate	ttcgtaagcg	tcatcaataa	gcgtaaaaaa	accgggcaat
1621	gccccggttt	ttaatgagaa	attttacctg	tcgtagccgc	caccatccgg	caaagaagca
1681	tacaaggctt	ttggcttata	gctacgtagc	gcattgcgtc	gcagcacaat	ccccgcaccg
1741	atcaagtctt	cgcgatgatt	attattatta	agctactaaa	gcgtagtttt	cgtegtttgc
1801	agcatttaag	actgcttttt	taaactgttc	attaataggc	atagacaata	caaccgattt
1861	agtatacct	aatacatgaa	tttctttgtc	tccaatacga	tgacaaggaa	ctttaatagc
1921	ctttaaaaat	cgctctattg	ctgttgatgt	tactgttaca	tattctgtaa	taccttgact
1981	aacagcgtgt	ttatataatg	cttcaaatag	tttcattgta	atttcactag	cagagttatt
2041	tatctttgag	ctatttttac	ctacagcaaa	acgacttaat	tcgactatat	taggatcttt
2101	gggagcactc	tgttgaccaa	gcaattcagg	aaaaacactt	ttcagcatat	aatcacctgt
2161	tgtaggtaat	aaacgccagc	atccacttac	attttcagta	tcatcacaag	cataaatata
2221	ttctgcattt	gagttatcat	actcactga	ttcaagggtta	tttctacaa	ctaagtecca
2281	ctcaagtctt	tgcttaaaca	cttgataacg	aagacttaga	atacctttat	actcctccga
2341	tggaattgcc	aaaaaatccg	atttttttat	cattatagtc	attttttttt	cctccttatt
2401	ttctccagga	agatcttcgg	tcagtgcgtc	ctgctgatgt	gctcagtatc	tctatcactg
2461	atagggatgt	caatctctat	cactgatagg	gagtcgacaa	aaataatgag	aatcaataga
2521	acttccgaga	agttcagccg	ctaataatcg	ccctgctcca	ttgtgcgccg	caataaaagt
2581	accggcatta	cgggtgcatt	ggcgcgccaa	atgcggccat	ctgtgggcaa	cctgtcgggt
2641	aagacccaaa	cttagtgtaa	taggtatcct	atgattattt	tttcatttga	tgccaaaaaa
2701	aaaagagtat	tgacttcgca	tctttttgta	cctataatag	attcattact	agagaaagag
2761	gagaaatact	agatggcctt	ggttgacggg	tttcttgagc	tggaacgctc	aagtggaaaa
2821	ttggagtgga	gcgccatcct	gcagaagatg	gcgagcgacc	ttggattctc	gaagatcctg

```

2881 ttcggcctgt tgcetaagga cagccaggac tacgagaacg ctttcatcgt cggcaactac
2941 ccggccgcct ggccgcgagca ttacgaccgg gctggctacg cgcgggtcga cccgacggtc
3001 agtcaactgta cccagagcgt actgccgatt ttctgggaac cgtccatcta ccagacgcga
3061 aagcagcaacg agttcttcga ggaagcctcg gccgccggcc tgggtgatgg gctgaccatg
3121 ccgctgcatg gtgctcgcgg cgaactcggc gcgctgagcc tcagcgtgga agcggaaaac
3181 cgggccgagg ccaaccgttt catggagtcg gtcttgccga cctgtggat gctcaaggac
3241 tacgcaactgc agagcgggtgc cggactggcc ttcgaacatc cggtcagcaa accggtggtt
3301 ctgaccagcc gggagaagga agtggtgcag tgggtgcgcca tcggcaagac cagttgggag
3361 atatcggtta tctgcaactg ctcggaagcc aatgtgaact tccatatggg aaatattcgg
3421 cggaaagtteg gtgtgacctc ccgccgcgta gcggccatta tggccgttaa tttgggtctt
3481 attactctct aaggatccaa actcgagtaa ggatctccag gcatcaaata aaacgaaagg
3541 ctcactcgaa agactgggcc tttcgtttta tctgtttggtt gtcggtgaac gctctctact
3601 agagtcaaac tggctcaact tcgggtgggc ctttctgcgt ttatacctag ggatatattc
3661 cgcttctctg ctcactgact cgctacgctc ggtegttcga ctgcggcgag cggaaatggc
3721 ttacgaacgg ggccggagatt tcttggaaga tgccaggaag ataacttaaca gggaagtgag
3781 agggcccgcg caaagccgtt tttccatagg ctccgcccc ctgacaagca tcacgaaatc
3841 tgacgctcaa atcagtggtg gcgaaaccgg acaggactat aaagatacca ggcgtttccc
3901 cctggcggct cctcgtgctg ctctcctggt cctgcctttc ggtttaccgg tgcattccg
3961 ctggttatggc cgcgtttgtc tcattccacg cctgacac

```

//

C.15 LatInh Plasmid 3: cell type A (pJH9-43)

LOCUS pJH9-43 5375 bp DNA circular
DEFINITION

pJH4-22 with luxR and lasI moved from pJH9-29 (Cell Type A).

pLtetO-1_lasI
pLuxI.tetR.LVAdegtag_mRFP1
pFAB46_luxR

FEATURES	Location/Qualifiers
rep_origin	complement(5018..325) /gene="ColE1"
misc_feature	complement(452..1111) /gene="CmR"
CDS	4135..4890 /gene="luxR" /codon_start="0"
CDS	3098..3775 /gene="mRFP1" /codon_start="0"
CDS	2410..3072 /gene="tetR_with_deg_tag" /codon_start="0"
misc_feature	3034..3066 /gene="LVA_deg_tag"

CDS	complement (1495..2103)							
	/gene=" lasI "							
	/codon_start=" 0 "							
terminator	331..436							
	/gene=" term_T0 "							
terminator	4906..5004							
	/gene=" term_rrnD_T1 "							
promoter	complement (2121..2183)							
	/gene=" pLtetO-1 "							
prot_bind	complement (2140..2158)							
	/gene=" tetO2 "							
prot_bind	complement (2165..2183)							
	/gene=" tetO2 "							
promoter	2316..2370							
	/gene=" pLuxI "							
prot_bind	2316..2334							
	/gene=" lux_box "							
promoter	4062..4108							
	/gene=" apFAB46 "							
	/note=" BIOFAB_promoter "							
misc_feature	2376..2409							
	/gene=" UTR_(rbs5000) "							
misc_feature	3078..3097							
	/gene=" RFP_RBS "							
terminator	complement (1300..1473)							
	/gene=" term_TSAL2 "							
terminator	1254..1299							
	/gene=" term_RNAI "							
misc_feature	4109..4134							
	/gene=" 5'-UTR_BBa_B0034 "							
	/note=" Parts_Registry:_BBa_B0034 "							
terminator	3800..3879							
	/gene=" term_rrnB "							
	/note=" Parts_Registry:_BBa_B0010 "							
terminator	3888..3928							
	/gene=" term_T7 "							
	/note=" Parts_Registry:_BBa_B0012 "							
BASE COUNT	1546	a	1236	c	1212	g	1381	t
ORIGIN								
1	ccactggcag	cagccactgg	taacaggatt	agcagagcga	ggtatgtagg	cggtgctaca		
61	gagttcttga	agtgggtggc	taactacggc	tacactagaa	ggacagtatt	tggtatctgc		
121	gctctgctga	agccagttac	cttcggaaaa	agagttggta	gctcttgatc	cggcaaaaa		
181	accaccgctg	gtagcgggtg	tttttttgtt	tgcaagcagc	agattacgcg	cagaaaaaaaa		
241	ggatctcaag	aagatccttt	gatcttttct	acgggggtctg	acgctcagtg	gaacgaaaac		
301	tcacgttaag	ggatthtggg	catgactagt	gcttgattc	tcaccaataa	aaaacgcccc		
361	gcggaaccg	agcgttctga	acaaatccag	atggagttct	gaggtcatta	ctggatctat		
421	caacaggagt	ccaagcgagc	tcgatatcaa	attacgcccc	gcctgcccac	tcctgcagtg		
481	actgthgtaa	ttcattaagc	attctgcccga	catggaagcc	atcaciaaacg	gcatgatgaa		
541	cctgaatcgc	cagcggcatc	agcaccttgt	cgcttgcgt	ataatatttg	cccatgggtga		
601	aaacgggggc	gaagaagttg	tccatattgg	ccacgtttaa	atcaaaaactg	gtgaaactca		

```

661 cccagggatt ggctgagacg aaaaacatat tctcaataaa cccttaggg aaataggcca
721 ggttttcacc gtaacacgcc acatcttgcg aatataatgt tagaaactgc cggaaatcgt
781 cgtgggtatc actccagagc gatgaaaacg tttcagtttg ctcattgaaa acgggtgtac
841 aagggtgaac actatcccat atcaccagct caccgtcttt cattgccata cgaaattccg
901 gatgagcatt catcaggcgg gcaagaatgt gaataaagge cggataaaac ttgtgcttat
961 ttttctttac ggtctttaaa aaggccgtaa tatccagctg aacggctctgg ttataggtac
1021 attgagcaac tgactgaaat gectcaaaat gttctttacg atgccattgg gatataatca
1081 cgggtggtata tccagtgtatt tttttctcca tgcgaaacga tectcatcct gtctcttgat
1141 cagatcatga tcccctgcgc catcagatcc ttggcggcaa gaaagccatc cagtttactt
1201 tgcagggcctt cccaacctta ccagagggeg cccagctgg caattccgac gtcgatccgg
1261 caaacaacc accgttggtg gcggttggtt ttttgtttg atcgacaatc ttcgtaagcg
1321 tcatcaataa gcgtaaaaaa accgggcaat gcccggtttt ttaatgagaa attttacctg
1381 tegtagecgc caccatccgg caaagaagca tacaaggctt ttggcttata gctacgtagc
1441 gcattgcgtc gcagcacaat cccggcaccg atcaagtctt cgcgatgatt attattatca
1501 tgaaacccgc agtcgctggt ccaccagcac tccccgtaa agcgcgatct gggtcttggc
1561 attgagttcg atgcgcaagg ccaccgcgcg ctcgatgccg atcttcaggt gcgaccgaa
1621 gcgcgatacg tccaggccgg cacggatcat catcttctcc acgectacgg tggttaccgt
1681 caccagcgtc tggatgtcgt tctgcaggct gtagegggccc agcgcgcgca tgcctccgg
1741 cgtacagtcg gaaaagccca gcgagccttt ctgtccagag ttgatggcga aacggctgag
1801 ttcccagatg tgcggcgagc aaggcgcttc cttgccgtgc agaagctccg ggaaggtggt
1861 cttcagcatg taggggccag tggatcagag aattgccag caaccgaaaa cctgggcttc
1921 aggagtatct tectggatca acatgtaata aggactgagt gcgtcataac catcgatttc
1981 catctcgtcg atgacactaa cgtcccagcc tttgcctcc ttgaacctt gagicgcaa
2041 cttgtgcate tgcgccagca gttttttatc gaactcttcg cgcgaccaa tttgtacgat
2101 catctagtat ttctctcttt tctctagtag tgetcagtat ctctatcact gatagggatg
2161 tcaatcteta tcaatgatag ggatttagct tecttagctc ctgaaaatct cgataactca
2221 aaaaatacgc ccggtagtga tcttatttca ttatggtgaa agttggaacc tcttacgtgc
2281 cgatcaacgt ctcatcttcg ccagatateg acgtcacctg taggatcgta caggtttacg
2341 caagaaaatg gtttggtata gtcgaataaa gatctaggaa aaagctcata taactagagt
2401 aagaggtcaa tgatgtctag attagataaa agtaaaagtga ttaacagcgc attagagctg
2461 cttaatgagg tcggaatcga aggtttaaaca acccgtaaac tgcgccagaa gctaggtgta
2521 gagcagccta cattgtattg gcatgtaaaa aataagcggg ctttgctcga cgccttagcc
2581 attgagatgt tagataggca ccatactcac ttttgccctt tagaagggga aagctggcaa
2641 gattttttac gtaataacgc taaaagtttt agatgtgctt tactaagtca tgcgatgga
2701 gcaaaagtac atttaggtac acggcctaca gaaaaacagt atgaaactct cgaaaatcaa
2761 ttagcctttt tatgccaaca aggtttttca ctagagaatg cattatatgc actcagcgtc
2821 gtggggcatt ttacttttagg ttgcgtattg gaagatcaag agcatcaagt cgctaaagaa
2881 gaaagggaaa cacctactac tgatagtatg ccgccattat tacgacaagc tatcgaatta
2941 tttgatcacc aaggtgcaga gccagccttc ttattcggcc ttgaattgat catatgcgga
3001 ttagaaaaac aacttaaagt tgaaagtggg tctgctgcaa acgacgaaa ctacgcttta
3061 gtagcttaat aagatctttt aagaaggaga tatacatatg gcgagtacg aagacgttat
3121 caaagagttc atgcgtttca aagttcgtat ggaaggttcc gttaacggtc acgagttcga
3181 aatcgaaggt gaaggtgaag gtcgtccgta cgaaggtacc cagaccgcta aactgaaagt
3241 taccaaaggt ggtccgctgc cgttcgcttg ggacatcctg tccccgcagt tccagtacgg
3301 ttccaaagct tacgttaaac acccgctga catcccggac tactgaaac tgtctctccc
3361 ggaaggtttc aaatgggaac gtgttatgaa ctctgaagac ggtggtggtg ttaccgttac
3421 ccaggactcc tccctgcaag acggtgagtt catctacaaa gttaaactgc gtggtaccaa
3481 ctccccgtcc gacggctccg ttatgcagaa aaaaaccatg ggttgggaag ctccaccga
3541 acgtatgtac ccggaagacg gtgctctgaa aggtgaaatc aaaatgcgct tgaaactgaa
3601 agacggtggt cactacgacg ctgaagttaa aaccacctac atggctaaaa aaccggttca

```

```

3661 gctgcccgggt gcttacaaaa cgcacatcaa actggacatc acctcccaca acgaagacta
3721 caccatcggt gaacagtaag aacgtgctga aggtcgtcac tccaccgggtg ctttaaggatc
3781 caaactegag taaggatctc caggcatcaa ataaaacgaa aggctcagtc gaaagactgg
3841 gcctttcggt ttatctggtg tttgtcgggtg aacgctctct actagagtca cactggctca
3901 ccttcgggtg ggcctttctg cgtttatacc tagggcggtc ggctgcggcg agcggtatca
3961 gctcactcaa aggcggtaat acggttatcc acagaatcag gggataacgc aggaaagaac
4021 atgtgagcaa aaggccagca aaaggccagg aaccgtaaaa aaaaaagagt attgacttcg
4081 catctttttg tacctataat agattcatta ctagagaaag aggagaaata ctagatgaaa
4141 aacataaatg ccgacgacac atacagaata attaataaaa ttaaagcttg tagaagcaat
4201 aatgatatta atcaatgctt atctgatatg actaaaatgg tacattgtga atattatta
4261 ctgcgatca tttatctca ttctatgggtt aaatctgata tttcaatcct agataattac
4321 cctaaaaaat ggaggcaata ttatgatgac gctaatttaa taaaatatga tcctatagta
4381 gattattcta actccaatca ttcaccaatt aattggaata tatttgaana caatgctgta
4441 aataaaaaat ctccaaatgt aattaaagaa gcgaaaacat caggtcttat cactgggttt
4501 agtttcccta ttcatacggc taacaatggc ttcggaatgc ttagttttgc acattcagaa
4561 aaagacaact atatagatag tttattttta catgcgtgta tgaacatacc attaatgtt
4621 ccttctctag ttgataatta tcgaaaaata aatatagcaa ataataaatc aaacaacgat
4681 ttaacccaaa gagaaaaaga atgttttagcg tgggcatgcg aaggaaaaag ctcttgggat
4741 atttcaaaaa tattaggttg cagtgagegt actgtcactt tccatttaac caatgcgcaa
4801 atgaaactca atacaacaaa ccgctgccaa agtatttcta aagcaatttt aacaggagca
4861 attgattgcc catactttta aaattaataa ggatctcagg tctcatgatg ggaactgcca
4921 gacatcaaat aaaacaaaag gctcagtcgg aagactgggc cttttgtttt atctgttgtt
4981 tgtegggtgaa cactctcccg ggcgctaggg tacgggtggc cgcgttgctg gcgtttttcc
5041 ataggctccg cccccctgac gagcatcaca aaaatcgac ctcaagtcag aggtggcgaa
5101 acccgacagg actataaaga taccagcgtt tccccctgg aagctccctc gtgcgctctc
5161 ctgttccgac cctgcccgtt accggatacc tgtecgctt tctccctcg ggaagegttg
5221 cgctttctca tagctcaagc tgtaggtatc tcagttcggg gtaggtcggt cgctccaagc
5281 tgggctgtgt gcacgaacce cccgttcagc ccgaccgctg cgccttatcc ggtaactatc
5341 gtcttgagtc caaccggta agacacgact tateg

```

//

C.16 LatInh Plasmid 3: cell type B (pJH9-44)

LOCUS pJH9-44 5348 bp DNA circular
DEFINITION

pJH4-22 with lasR and luxI moved from pJH9-30 (Cell Type B).

pLtetO-1_luxI
pLuxI.tetR_LVAdegtag_mRFP1
pFAB46.lasR

FEATURES

	Location/Qualifiers
rep_origin	complement(4991..325) /gene="ColE1"
misc_feature	complement(452..1111) /gene="CmR"
CDS	4144..4863

```

                /gene="lasR"
                /codon_start="0"
CDS            3107..3784
                /gene="mRFP1"
                /codon_start="0"
CDS            2419..3081
                /gene="tetR_with_deg_tag"
                /codon_start="0"
misc_feature   3043..3075
                /gene="LVA_deg_tag"
CDS            complement(1495..2112)
                /gene="luxI"
                /codon_start="0"
misc_feature   complement(1501..1533)
                /gene="LVA_deg_tag"
terminator     331..436
                /gene="term_T0"
terminator     4879..4977
                /gene="term_rrnD_T1"
promoter       complement(2130..2192)
                /gene="pLtetO-1"
prot_bind      complement(2149..2167)
                /gene="tetO2"
prot_bind      complement(2174..2192)
                /gene="tetO2"
promoter       2325..2379
                /gene="pLuxI"
prot_bind      2325..2343
                /gene="lux_box"
promoter       4071..4117
                /gene="apFAB46"
                /note="BIOFAB_promoter"
misc_feature   2385..2418
                /gene="UTR_(rbs5000)"
misc_feature   3087..3106
                /gene="RFP_RBS"
terminator     complement(1300..1473)
                /gene="term_TSAL2"
terminator     1254..1299
                /gene="term_RNAI"
misc_feature   4118..4143
                /gene="5'-UTR_BBa_B0034"
                /note="Parts_Registry:_BBa_B0034"
terminator     3809..3888
                /gene="term_rrnB"
                /note="Parts_Registry:_BBa_B0010"
terminator     3897..3937
                /gene="term_T7"
                /note="Parts_Registry:_BBa_B0012"
BASE COUNT    1462 a   1255 c   1251 g   1380 t

```

ORIGIN

```

1  ccaactggcag  cagccactgg  taacaggatt  agcagagcga  ggtatgtagg  cggtgctaca
61  gagttcttga  agtgggtggc  taactacggc  taaactagaa  ggacagtatt  tggatctgct
121  gctctgctga  agccagttac  cttcggaaaa  agagttggta  gctcttgatc  cggcaaaaaa
181  accaccgctg  gtagcgggtg  tttttttggt  tgcaagcagc  agattacgcg  cagaaaaaaaa
241  ggatctcaag  aagatccttt  gatcttttct  acgggggtct  acgctcagtg  gaacgaaaac
301  tcacgttaag  ggattttggt  catgactagt  gcttggattc  tcaccaataa  aaaacgcccc
361  gcggaaccg  agcgttctga  acaaatccag  atggagttct  gaggtcatta  ctggatctat
421  caacaggagt  ccaagegagc  tcgatatcaa  attacgcccc  gccctgccac  tcatcgcagt
481  actgttgtaa  ttcattaagc  attctgccga  catggaagcc  atcaciaaac  gcatgatgaa
541  cctgaategc  cageggcacc  agcaccttgt  cgccttgcgt  ataataattg  cccatgggtg
601  aaacgggggc  gaagaagttg  tccatattgg  ccacgtttaa  atcaaaactg  gtgaaactca
661  cccagggtat  ggctgagacg  aaaaacatat  tctcaataaa  ccctttaggg  aaataggcca
721  ggttttcacc  gtaaacgcgc  acatcttgcg  aatataatgt  tagaaactgc  cggaaatcgt
781  cgtgggtatc  actccagagc  gatgaaaacg  tttcagtttg  ctcatggaaa  acggtgtaac
841  aagggtgaac  actatcccat  atccaccgct  caccgtcttt  cattgccata  cgaaatccg
901  gatgagcatt  catcaggcgg  gcaagaatgt  gaataaaggc  cggataaaac  ttgtgcttat
961  ttttctttac  ggtcttttaa  aaggccgtaa  tatccagctg  aacggctctg  ttataggtac
1021  attgagcaac  tgaactgaa  gctcaaaaat  gttctttacg  atgccattgg  gatataatca
1081  cgggtgtata  tccagtgtat  tttttctcca  tgcgaaacga  tctcctcct  gtctcttgat
1141  cagatcatga  tcccctgcgc  catcagatcc  ttggcggcaa  gaaagccatc  cagtttactt
1201  tgcagggcct  cccaacctta  ccagaggcgg  ccccagctgg  caattccgac  gtcgatccgg
1261  caaacaacc  accgttggtg  gcggttggtt  ttttgtttgg  atcgacaate  ttcgtaagcg
1321  tcatcaataa  gcgtaaaaaa  accgggcaat  gcccggtttt  ttaatgagaa  attttacctg
1381  tegtagecgc  caccatccgg  caaagaagca  tacaaggctt  ttggcttata  gctacgtagc
1441  gcattgcgtc  gcagcacaat  cccggcaccg  atcaagtctt  cgcgatgatt  attattatta
1501  agctactaaa  gcgtagtgtt  cgtegtttgc  agcatttaag  actgcttttt  taaactgttc
1561  attaataggc  atagacaata  caaccgattt  agtatcact  aatacatgaa  tttctttgtc
1621  tccaatacga  tgacaaggaa  ctttaatac  ctttaaaaat  cgctctattg  ctggtgatgt
1681  tactgttaca  tattctgtaa  taccttgact  aacagcgtgt  ttatataatg  ctcaaatag
1741  tttcattgta  atttcactag  cagagttatt  tatctttgag  ctatttttac  ctacagcaaa
1801  acgacttaat  tcgactatat  taggatcttt  gggagcactc  tgttgaccaa  gcaattcagg
1861  aaaaacactt  ttcagcatat  aatcacctgt  tgtaggtaat  aaacgccagc  atccacttac
1921  attttcagta  tcatcacaag  cataaatata  ttctgcattt  gagttatcat  actcatctga
1981  ttcaaggtta  ttttctacaa  ctaagtccca  ctcaagtctt  tgcttaaaca  cttgataaac
2041  aagacttaga  atacctttat  actctccga  tggaattgcc  aaaaaatccg  atttttttat
2101  cattatagtc  atctagtatt  tctcctcttt  ctctagtagt  gctcagatc  tctatcactg
2161  atagggatgt  caatctctat  cactgatagg  gatttagctt  ccttagctcc  tgaaaatctc
2221  gataactcaa  aaaatacgcc  cggtagtgt  cttatttcat  tatggtgaaa  gttggaacct
2281  cttacgtgcc  gatcaacgtc  tcattttcgc  cagatatega  cgtcacctgt  aggatcgtac
2341  aggtttacgc  aagaaaatgg  tttgttatag  tcgaataaag  atctaggaaa  aagctcatat
2401  aactagagta  agaggtcaat  gatgtctaga  ttagataaaa  gtaaagtgat  taacagcgca
2461  ttagagctgc  ttaatgaggt  cggaatcgaa  ggtttaaaca  cccgtaaact  cgcccagaag
2521  ctagggtgta  agcagcctac  attgtattgg  catgtaaaaa  ataagcgggc  tttgctcgac
2581  gccttagcca  ttgagatggt  agataggcac  catactact  tttgccttt  agaaggggaa
2641  agctggcaag  attttttacg  taataacgct  aaaagtttta  gatgtgcttt  actaagtcac
2701  cgcgatggag  caaaagtaca  tttaggtaca  cggcctacag  aaaaacagta  tgaaactctc
2761  gaaaatcaat  tagccttttt  atgccaacaa  ggtttttcac  tagagaatgc  attatagca
2821  ctacgcgctg  tggggcattt  tactttaggt  tgcgtattgg  aagatcaaga  gcatcaagtc
2881  gctaaagaag  aaagggaaac  acctactact  gatagtatgc  cgccattatt  acgacaagct

```

2941 atcgaattat ttgatcacca aggtgcagag ccagccttct tattcggcct tgaattgatc
3001 atatgicggat tagaaaaaca acttaaagt gaaagtgggt ctgctgcaaa cgacgaaaac
3061 tacgcttttag tagcttaata agatctttta agaaggagat atacatatgg cgagtagcga
3121 agacgtttatc aaagagttca tgcgtttcaa agttcgtatg gaaggttccg ttaacggtea
3181 cgagttcgaa atcgaaggtg aaggtgaagg tcgtccgtac gaaggtaccg agaccgctaa
3241 actgaaagtt accaaaggtg gtccgctgcc gttcgttgg gacatcctgt ccccgagtt
3301 ccagtacggg tccaaagctt acgttaaaca cccggctgac atcccggact acctgaaact
3361 gtccttcccc gaaggtttca aatgggaacg tgttatgaac ttcgaagacg gtgggtgtgt
3421 taccgttacc caggactcct cctgcaaga cggtagtgc atctacaaag ttaaactgcg
3481 tggtaaccaac ttcccgtccg acgggtccgg tatgcagaaa aaaacctagg gttgggaagc
3541 ttccaccgaa cgtatgtacc cggaagacgg tgctctgaaa ggtgaaatca aaatgcgtct
3601 gaaactgaaa gacgggtggtc actacgacgc tgaagttaaa accacctaca tggctaaaaa
3661 accggttcag ctgccgggtg cttacaaaac cgacatcaaa ctggacatca cctcccacaa
3721 cgaagactac accatcgttg aacagtacga acgtgctgaa ggtcgtcact ccaccgggtc
3781 ttaaggatcc aaactcgagt aaggatctcc aggcacaaa taaaacgaaa ggctcagtcg
3841 aaagactggg cctttcgttt tatctgttgt ttgtcgggtga acgctctcta ctagagtcac
3901 actggctcac cttcgggtgg gcctttctgc gtttatacct agggcgcttcg gctgcggcga
3961 gcggtatcag ctcactcaaa ggcggtaata cggttatcca cagaatcagg ggataacgca
4021 ggaaagaaca tgtgagcaaa aggccagcaa aaggccagga accgtaaaaa aaaaagagta
4081 ttgacttccg atctttttgt acctataata gattcattac tagagaaaga ggagaaatac
4141 tagatggcct tggttgacgg tttcttgag ctggaacgct caagtggaaa attggagtgg
4201 agcgcacatc tgcagaagat ggcgagcgac cttggattct cgaagatcct gttcggcctg
4261 ttgcctaagg acagccagga ctacgagaac gccttcacg tcggcaacta cccggccgcc
4321 tggecgagc attacgaccg ggctggctac gcgcgggtcg acccgacggt cagtcactgt
4381 accgagagcg tactgccgat tttctgggaa ccgtccatct accgacgcg aaagcagcac
4441 gagttcttcg aggaagcctc ggccgcgggc ctggtgtatg ggctgacct gccgctgcat
4501 ggtgctcgcg gcgaactcgg cgcgctgagc ctcagcgtgg aagcggaaaa cggggccgag
4561 gccaaaccgtt tcatggagtc ggtcctgcc accctgtgga tgctcaagga ctacgcactg
4621 cagagcgggt ccggactggc cttcgaacat ccggtcagca aaccgggtgg tctgaccagc
4681 cgggagaagg aagtgttgca gtggtgcgcc atcggcaaga ccagttggga gatategggt
4741 atctgcaact gctcgggaagc caatgtgaac ttccatatgg gaaatatcgg gcggaagttc
4801 ggtgtgacct cccgccgcgt agcggccatt atggccgtta atttgggtct tattactctc
4861 taaggatctc aggtctcatg atgggaactg ccagacatca aataaaacaa aaggctcagt
4921 cggaagactg ggccctttgt tttatctgtt gtttgcggt gaacactctc cggggcgcta
4981 gggtacgggt ggccgcggtg ctggcgttt tccataggct ccgccccct gacgagcact
5041 acaaaaaatg acgctcaagt cagaggtggc gaaaccgac aggactataa agataccagg
5101 cgtttcccc tggaagctcc ctcgtgcgct ctctgttcc gacctgccg cttaccggat
5161 acctgtccgc ctttctccct tcgggaagcg tggecgcttc tcatagctca cgctgtaggt
5221 atctcagttc ggtgtaggtc gttcgtcca agctgggctg tgtgcacgaa cccccgttc
5281 agccccagc ctgcgcctta tccggttaact atcgtcttga gtccaaccg gtaagacagc
5341 acttatcg

//