5-2016

A Boolean Network Model of the L-Arabinose Operon

Andy Jenkins
Clemson University, leej@g.clemson.edu

A Boolean Network Model of the L-arabinose operon

A Thesis
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Master of Science
Mathematical Sciences

---

by
Andy Jenkins
May 2016

Accepted by:
Dr. Matthew Macauley, Committee Chair
Dr. Elena Dimitrova
Dr. Svetlana Poznanovikj

Abstract

The regulation of gene expression is essential for the maintenance of homeostasis within an organism. Thus, the ability to predict which genes are expressed and which are silenced based on the cellular environment is highly desired by molecular biologists. Mathematical models of gene regulatory networks have frequently been given in terms of systems of differential equations, which although useful for understanding the mechanisms of regulation, are not always as interpretable as discrete models when one wishes to analyze the global-level dynamics of the system. In particular, Boolean network models have been previously shown to be simple yet effective tools for modeling operons such as the lactose operon in *Escherichia coli*. In this thesis, we propose a Boolean model of a similar nature for the arabinose operon. While this operon is also used by *E. coli* to regulate sugar metabolism, it contains several unique biological features such as a positive inducible control mechanism that distinguish it from previously modeled gene networks. By treating the network model as a polynomial dynamical system, analysis of the system dynamics shows that our model accurately captures the biological behavior of the operon and also provides insight into interactions within the network.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

A core theme throughout molecular biology is the central dogma, which explains how information flows from genotype (the genetic code) to the phenotype (the observable physical characteristics of an organism) [C$^+$70]. Genetic information encoded in the nucleotides of the DNA strand undergoes transcription by RNA polymerase enzymes to produce a messenger RNA (mRNA) strand. This mRNA is then translated by ribosomes into an amino acid sequence known as a polypeptide, or protein. The functions of these proteins then lead to the phenotype. Clearly this process requires extensive regulation, as the desire for presence or absence of a protein, as well as its concentration, are dependent on the current state of the organism and its environmental conditions [VCS11].

Regulation of gene expression can occur at any of the steps during the flow of information, including transcription, post-transcription, translation, and post-translation modifications [LBZ$^+$00]. The regulation of transcription in prokaryotic organisms has been particularly well-studied, and most of the early understanding of genetic regulatory systems came from the study of operons [JM61]. An *operon* is a collection of contiguous genes with related functions that are all transcribed onto a single mRNA strand, along with two adjacent control sequences that regulate their expression. The genes whose protein products perform some coordinated function are known as structural genes. The two control sequences are a promoter, a region of DNA to which RNA polymerase binds to initiate transcription of the structural genes, and an operator, a sequence of nucleotides located between the promoter and structural genes whose status determines whether transcription will occur. The state of the operator and its method of controlling gene expression can be categorized into one of four types: the transcription of the structural genes can be regulated by positive (an activator protein

binds to the operator to initiate transcription) or negative (a repressor protein bound to the operator prevents transcription) control, and the mechanism of regulation can be inducible (addition of a metabolite initiates transcription) or repressible (addition of a metabolite prevents transcription) [EB68].

The first described and most extensively studied operon is the lactose (*lac*) operon in *E. coli* [JPSM05]. The structural genes of the *lac* operon produce enzymes that participate in the catabolism of the sugar lactose, which the cell can use as an alternative energy source to the more preferred sugar glucose. Thus transcription of the system of genes should occur only when lactose is present and glucose is not available. This transcriptional control is accomplished by the binding of a repressor protein to the operator, which physically blocks RNA polymerase from binding to the promoter region, and which can only be removed by the presence of lactose. The second method of control occurs due to catabolite repression, where the presence of glucose inhibits the synthesis of the enzymes necessary for lactose metabolism.

The *lac* operon is the prototypical example of a negative inducible operon, and for many years it was assumed that this was the only mechanism by which these gene regulatory networks were regulated. Later discoveries proved this hypothesis false, as the tryptophan (*trp*) [BKL$^+$75] and arginine (*arg*) [LHA92] operons exhibit negative repressible control mechanisms, and the arabinose (*ara*) operon utilizes both positive and negative inducible control mechanisms [GS71].

Many proposed models of the interactions between the elements of the *lac* operon exist, where the dynamics of the system are frequently modeled as a system of ordinary differential equations (ODEs) [SM08]. While these models are very useful for understanding mechanisms and obtaining quantitative information about the state of the system at a given time, their inherent complexity can make qualitative interpretations of the solutions a daunting task. Furthermore, these solutions are dependent on specific initial conditions, parameter values, and rate constants, many of which are experimentally determined and can require extensive knowledge of the reaction kinetics occurring within the system in order to make an accurate prediction [DB08]. An alternative modeling approach under a discrete framework uses Boolean networks (BNs), which in many cases provide a more intuitive description of the biological system. In this case, each component of the network is assigned a value of 0 or 1 at each time step, usually indicating absence or presence (relative to some specified threshold) of a biological product at the current time step. More recent models of the *lac* operon have found this modeling framework to be effective both in capturing the key qualitative

features of the network and in accurately predicting the behavior of the operon, in the sense that the resulting state of the network agrees with the state of the operon for each given set of initial conditions [VCS11].

Another well-studied operon that frequently appears in the biological literature is the arabinose (*ara*) operon. Similar to the *lac* operon, the function of the *ara* operon is the metabolism of the sugar arabinose in the absence of glucose. However, the mechanism by which the network is regulated is significantly different and also more complex than that of the *lac* operon, as the *ara* operon exhibits both positive and negative inducible control [MHS86]. Although continuous models such as systems of ODEs have been formulated [Yil12], no logical model of the *ara* operon has been proposed.

In this thesis, we present a Boolean network model of the arabinose operon and analyze the system to show that it is a valid interpretation of how the operon functions biologically. The organization is as follows. In Chapter 2 we give a mathematical treatment of Boolean networks, describe how such models are analyzed, and discuss how a model can be reduced in order to capture the core qualitative behavior. In Chapter 3 we review the work done by Stigler and Veliz-Cuba on the *lac* operon as a motivating example for the use of this modeling framework with gene regulatory networks. Our novel Boolean network model of the *ara* operon is presented in Chapter 4, along with a discussion of the network dynamics. Finally, we end with some remarks on possible future projects in Chapter 5.

Chapter 2

Boolean Network Models

2.1 Preliminaries

In this section we give a brief mathematical background of Boolean networks. Throughout, let $\mathbb{F}_2 = \{0, 1\}$ denote the finite field of order 2.

Definition 2.1.1. A *Boolean network*(BN) $F = (f_1, f_2, \ldots, f_n)$ on n variables $x_1, x_2, \ldots, x_n$ is a sequence of Boolean functions $f_i : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$.

The Boolean functions in Definition 2.1.1 that govern the dynamical evolution of the network are given as logical functions of the model variables. That is, we define the functions f_i in terms of the x_i using the logical operators AND ($\wedge$), OR ($\vee$), and NOT ($\_$). We can then apply these functions to update the network variables at each timestep to obtain the new state of F. Frequently, the updates are performed synchronously across the variables, although it is possible to update asynchronously. In fact, an asynchronous update schedule is more "natural" in the biological sense, as particular molecules or structures are changing states at varying rates. However, synchronous updates still capture the qualitative behavior of the biological system, so we choose in this thesis to consider the synchronous case.

Definition 2.1.2. The *wiring diagram* of a Boolean network F is the directed graph with vertex set $V = \{1, 2, \ldots, n\}$ and edges (i, j) if function f_j depends on variable x_i.

The structure of the connections within the network (or *network topology*) depicted in the wiring diagram provides useful biological information in the form of directed cycles known as *feedback*

loops. A given directed edge (i, j) has a circle at its head if variable x_i has an inhibitory interaction with x_j. This corresponds to the presence of NOT x_i, or $\overline{x_i}$, in the Boolean function f_j. Otherwise the edge has an arrow at its head indicating a positive interaction.

The state transitions of the network are referred to as the *network dynamics*. In order to evaluate the dynamics of the system F, we need to evaluate the functions f_i over all possible variable inputs. We can then view the dynamics as a directed graph known as the *state space of* F, where each node in the graph represents a particular state of the network, where the edges are of the form $x \longrightarrow f(x)$ for all $x \in \mathbb{F}_2^n$.

Definition 2.1.3. Directed cycles in the state space are called *limit cycles*. If the length of the limit cycle is 1, then it is called a *fixed point*.

In biological applications, a fixed point is equivalent to a *steady state* of the system. From an applied standpoint, we are interested in determining whether the biological system reaches a fixed point, enters into a longer limit cycle, or exhibits some other behavior given initial conditions on the system. However, as the state space grows exponentially with the number of variables in the system, determining the state space via brute force calculations is not computationally feasible for any reasonably sized network. In the next section we discuss an alternative method of determining the fixed points of a network via computational algebra. First, we require a few algebraic definitions.

Definition 2.1.4. Let K be a field and consider $K[x_1, \ldots, x_n]$, the ring of polynomials in n variables with coefficients in K. Let $\mathcal{F} \subset K[x_1, \ldots, x_n]$. We define the *ideal generated by* $\mathcal{F}$ to be the set

$$\langle \mathcal{F} \rangle = \{p_1 f_1 + \ldots + p_r f_r \mid f_1, \ldots, f_r \in \mathcal{F}, \;\; p_1, \ldots, p_r \in K[x_1, \ldots, x_n]\}.$$

Definition 2.1.5. Consider the set of monomials

$$M = \{x_1^{i_1} x_2^{i_2} \cdots x_m^{i_m} \mid i_1, i_2, \ldots, i_m \in \mathbb{N}\}$$

of $K[x_1, \ldots, x_n]$. A *monomial order* on M is a total order $\prec$ such that

(i) the constant monomial 1 is the smallest monomial.

(ii) it is multiplicative. That is, if $x_1^{i_1} \cdots x_m^{i_m} \prec x_1^{j_1} \cdots x_m^{j_m}$ then $x_1^{i_1+k_1} \cdots x_m^{i_m+k_m} \prec x_1^{j_1+k_1} \cdots x_m^{j_m+k_m}$.

Definition 2.1.6. Fix a monomial order $\prec$. Every polynomial $p \in K[x_1, \ldots, x_n]$ has a unique *initial monomial*, denoted by $\text{in}_\prec(f)$, defined as the largest monomial $x^a = x_1^{a_1} \cdots x_m^{a_m}$ (with respect to $\prec$) in f with a non-zero coefficient. The *initial ideal* $\text{in}_\prec(I)$ is the ideal generated by the initial monomials of all $f \in I$.

Definition 2.1.7. A finite subset $\mathcal{G}$ of an ideal I is called a *Gröbner basis* with respect to a monomial order $\prec$ if

$$\text{in}_\prec(I) = \langle \text{in}_\prec(g) \mid g \in \mathcal{G} \rangle.$$

Suppose, in addition, that the following conditions hold:

(i) For each $g \in \mathcal{G}$, the coefficient of the initial monomial of g is 1.

(ii) The set of initial monomials minimally generates $\text{in}_\prec(I)$.

(iii) No trailing term of any $g \in \mathcal{G}$ is in $\text{in}_\prec(I)$.

Then we say that $\mathcal{G}$ is a *reduced Gröbner basis.*

It can be shown that for a fixed monomial order, any ideal $I \subset K[x_1, \ldots, x_n]$ has a unique reduced Gröbner basis which can be computed using the *Buchberger algorithm* (see [Stu96], [CLO92], for a more detailed exposition). Our application of Gröbner bases will be in solving systems of polynomial equations. The *variety* of a set of polynomials $\mathcal{F}$ is the set of all common zeros of the polynomials. Since the variety does not change when $\mathcal{F}$ is replaced by another set of polynomials which generate the same ideal, we can in particular compute the variety of the reduced Gröbner basis $\mathcal{G}$ of $\mathcal{F}$ to find the solution set to the system $\{f_i = 0 \mid f_i \in \mathcal{F}\}$. Therefore, in our Boolean network context, if we can convert our Boolean functions to polynomial functions over the finite field $\mathbb{F}_2$, we can then compute the variety of the corresponding reduced Gröbner basis $\mathcal{G}$ to find the fixed points. In practice, computational algebra systems are used to compute $\mathcal{G}$, and the solution to the resulting system is "obvious," as our polynomials are over $\mathbb{F}_2$.

2.2 Analyzing a model

Suppose we are given a Boolean network model $F = (f_1, \ldots, f_n)$. We convert f_i to polynomials p_i in $\mathbb{F}_2[x_1, \ldots, x_n]$ using the following rules to convert the logical operators AND, OR, and NOT:

- $x_i \wedge x_j = x_i x_j$
- $x_i \vee x_j = x_i + x_j + x_i x_j$
- $\overline{x_i} = x_i + 1$

Solving for the fixed points $\{f_i = x_i \mid i = 1, \ldots, n\}$ in the Boolean network is equivalent to solving the system of polynomials $\mathcal{F} = \{p_i + x_i = 0\}$. To accomplish this we perform the following algorithm for each combination of parameter values:

1. Define $I = \langle p_i + x_i \rangle$, where $i = 1, \ldots, n$, to be the ideal generated by our polynomials.
2. Compute a (reduced) Gröbner basis $\mathcal{G} = \{g_j\}$ of I using the computer algebra system Sage (code that accomplishes this is given in Appendix A).
3. Solve the system of polynomials $\{g_j = 0 \mid j = 1, \ldots, n\}$ to obtain the solution set $S \subseteq \mathbb{F}_2^n$.

Since S is also the solution set to $\mathcal{F}$, each element $x \in S$ corresponds to a fixed point of the network.

2.3 Reduction of Boolean models

A question frequently of interest is to determine what the "essential" components are of a given network, in the sense that the dynamics of the entire network can be captured by only examining the qualitative behavior of this subnetwork of essential components. In order to find this subnetwork, we would like to delete those nodes and edges that are nonfunctional, or do not directly function in the regulation. This can be determined logically by simplifying the Boolean expressions in the logical functions. Below is the method used in [VCS11] to reduce a given Boolean network and its associated wiring diagram.

1. Simplify Boolean expressions using Boolean algebra to delete nonfunctional variables.
2. Remove edges not corresponding to Boolean expressions to delete nonfunctional edges.
3. Remove vertices x_i whose associated Boolean function f_{x_i} does not depend on x_i.
   (a) If the vertex y depends on x_i, replace $f_y(x_1, \ldots, x_i, \ldots, x_n)$ by $f_y(x_1, \ldots, f_{x_i}, \ldots, x_n)$.
   (b) Replace edges $(y, x_i), (x_i, z)$ by (y, z).

Chapter 3

The *lac* operon

3.1 Biological background

In this section, we give a biological description of the *lac* operon found in *E. coli*. The majority of this description is adapted from information in the book [Pie12].

The proteins produced by the structural genes (*lacZ*, *lacY*, *lacA*) of this system carry out the metabolism of the lactose sugar, which is used as an energy source by the cell. Lactose enters the cell via the transporter protein β-galactoside permease (LacY), where it can be cleaved into glucose and its stereoisomer galactose by the enzyme β-galactosidase (LacZ). This enzyme is also responsible for converting lactose to allolactose. Finally, β-galactoside transacetylase (LacA) transfers an acetyl group from acetyl-CoA to β-galactoside.

The expression of these structural genes is regulated by the repressor protein product of the *lacI* gene. In the absence of lactose, this repressor protein is tightly bound to the operator site of the operon. This prevents RNA polymerase from binding to the promoter and therefore it cannot transcribe the structural genes, so the operon is "off". If extracellular lactose is available, diffusion of the sugar can occur at low concentrations. The few available molecules of β-galactosidase can convert the lactose to its isomer, allolactose. This isomer binds to the repressor protein, causing it to undergo a conformational change and the protein subsequently dissociates from the operator. Thus the binding of the RNA polymerase to the promoter is no longer blocked, the structural genes are transcribed, and the operon is "on".

Glucose also plays a role in regulating the transcription of the structural genes via catabolite

repression. In addition to the inactivation of the repressor, a substrate bound protein known as the cyclic AMP catabolite activator protein complex (or cAMP-CAP complex) must bind to the promoter region to cause the DNA to undergo further conformational changes to enhance the binding of RNA polymerase to the promoter. The presence of glucose inhibits the production of cAMP, thereby preventing transcription of the *lac* operon.

Finally, in the absence of glucose there exists a certain range of "medium" extracellular lactose concentrations such that the operon's behavior exhibits *bistability*. That is, if a particular population of cells is exposed to a growth medium containing an extracellular lactose concentration in this range, we can expect to find some cells with the *lac* operon being actively transcribed and other cells with the operon repressed. Whether the *lac* operon is on or off under medium lactose levels depends on whether the cell developed in a lactose-rich or lactose-starved environment.

3.2 A Boolean model

This Boolean model of the lac operon is comprised of the variables and functions representing the presence or absence of the molecules involved in the gene regulatory network. We take as parameters the presence or absence of extracellular glucose (G_e) and extracellular lactose (L_e, L_{em}). Note that intracellular lactose, extracellular lactose, allolactose, and repressor protein concentrations have been split into two variables (no subscript and m subscript) in order to represent three possible states for those variables:

$$\text{Concentration of } X = \begin{cases} \text{low} & X_m = X = 0 \\ \text{medium} & X_m = 1, X = 0 \\ \text{high} & X_m = X = 1 \end{cases}$$

Splitting these variables in this manner will allow our model to capture the bistability present in the network when extracellular lactose is at a medium concentration.

3.2.1 Boolean variables

The Boolean variables are labeled as follows:

- M = *lac* mRNA

- P = *lac* permease
- B = *lac* β-galactosidase
- C = cAMP-CAP complex
- R = repressor protein
- A = allolactose
- L = lactose
- G = glucose

where the subscript m denotes medium concentration and e denote extracellular concentrations.

3.2.2 Derivation of Boolean functions

The derivation of the Boolean functions is as follows:

- For the structural genes' mRNA to be transcribed, we need the presence of the cAMP-CAP complex as well as the absence of the repressor protein. Thus the Boolean function is $f_M = C \wedge \overline{R} \wedge \overline{R_m}$.
- For the permease protein to be produced, there must be mRNA present that can be translated. Thus the Boolean function is $f_P = M$.
- For the β-galactosidase enzyme to be produced, there must be mRNA present that can be translated. Thus the Boolean function is $f_B = M$.
- For the cAMP-CAP complex to form, high concentrations of cAMP are necessary, which only occurs in the absence of glucose. Thus the Boolean function is $f_C = \overline{G_e}$.
- The repressor protein will be active in the absence of allolactose. Thus the Boolean function is $f_R = \overline{A} \wedge \overline{A_m}$.
- The repressor will be active at a medium level or higher in the absence of allolactose, or if the repressor was active at a high level. Thus the Boolean function is $f_{R_m} = (\overline{A} \wedge \overline{A_m}) \vee R$.
- To have high concentrations of allolactose, we need intracellular lactose to be present and the enzyme β-galactosidase to be present. Thus the Boolean function is $f_A = L \wedge B$.

- To have medium concentrations of allolactose, we need either medium or high concentrations of intracellular lactose so that the basal number of β-galactosidase molecules can convert the lactose. Thus the Boolean function is $f_{A_m} = L \vee L_m$.

- To have high concentrations of intracellular lactose, we need both the membrane permease protein and extracellular lactose to be present and also extracellular glucose to be absent. Thus the Boolean function is $f_L = P \wedge L_e \wedge \overline{G_e}$.

- To have medium concentrations of intracellular lactose, we need extracellular lactose to be available in either high concentrations so that it can readily diffuse across the membrane without the assistance of the permease protein, or we require the presence of the permease with at least medium concentrations of extracellular lactose. We also require extracelluluar glucose to be absent. Thus the Boolean function is $f_{L_m} = ((L_{em} \wedge P) \vee L_e) \wedge \overline{G_e}$.

3.2.3 Wiring Diagram

The wiring diagram of the model is given below in Figure 3.1. We denote parameters by square nodes and variables by circular nodes. Edges between nodes with an arrow indicate positive interactions and circles indicate negative interactions. The shaded region represents intracellular space. Note that we have displayed L, L_m as a single node, as well as L_e, L_{em}, R, R_m, and A, A_m.

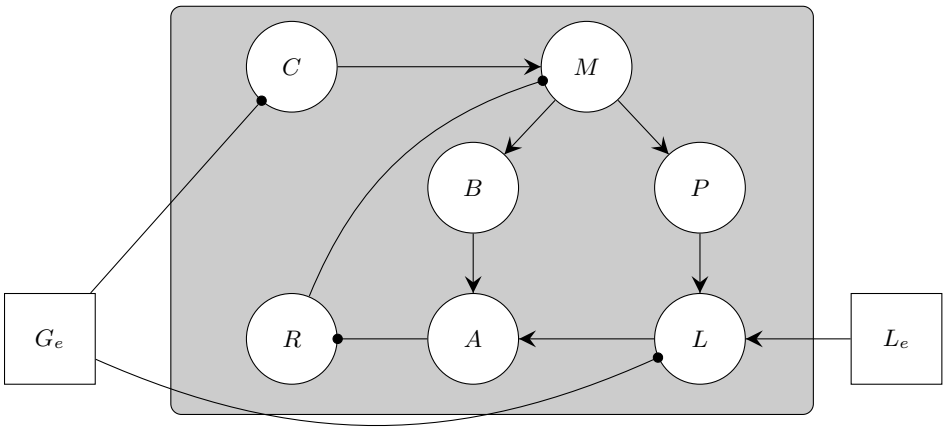

Figure 3.1: The wiring diagram of a Boolean model of the *lac* operon.

3.3 Network dynamics

We can analyze the dynamics of our Boolean network model by specifying initial conditions on the parameters and then evaluating the Boolean functions at each time step to determine if the system eventually reaches a steady state in the form of a fixed point of the state space. This is equivalent to solving the system $\{f_{x_i} = x_i \mid i = 1, \ldots, 10\}$ where we have renamed our Boolean variables as follows:

$$(M, B, R, A, L, P, C, R_m, A_m, L_m) = (x_1, x_2, x_3, x_4, x_5, x_6, x_7.x_8, x_9, x_{10})$$

To perform this analysis, we first convert our logical functions to a system of polynomial equations in $\mathbb{F}_2[x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}]$ using the methods described in Chapter 2. By working in this algebraic setting, we will be able to use computational algebra to assist in our network analysis. The resulting system of polynomials F is

$$\begin{cases} f_M = C \wedge \overline{R} \wedge \overline{R_m} = M \\ f_B = M = B \\ f_R = \overline{A} \wedge \overline{A_m} = R \\ f_A = L \wedge B = A \\ f_L = P \wedge L_e \wedge \overline{G_e} = L \\ f_P = M = P \\ f_C = \overline{G_e} = C \\ f_{R_m} = (\overline{A} \wedge \overline{A_m}) \vee R = R_m \\ f_{A_m} = L \vee L_m = A_m \\ f_{L_m} = ((L_{em} \wedge P) \vee L_e) \wedge \overline{G_e} = L_m \end{cases} \iff \begin{cases} x_1 + x_7(x_3+1)(x_8+1) = 0 \\ x_2 + x_1 = 0 \\ x_3 + (x_4+1)(x_9+1) = 0 \\ x_4 + x_2x_5 = 0 \\ x_5 + x_6L_e(G_e+1) = 0 \\ x_6 + x_1 = 0 \\ x_7 + G_e + 1 = 0 \\ x_8 + (x_4+1)(x_9+1) + x_3 + x_3(x_4+1)(x_9+1) = 0 \\ x_9 + x_5 + x_{10} + x_5x_{10} = 0 \\ x_{10} + (x_6L_{em} + L_e + x_6L_{em}L_e)(1+G_e) = 0 \end{cases}$$

For each of the 6 parameter combinations $(G_e, L_{em}, L_e) \in \mathbb{F}_2^3$, where we do not consider the results with $L_{em} = 0, L_e = 1$ (since this case cannot occur) we must solve F using the algorithm described in Chapter 2, where the resulting solutions will be the fixed points of the network. The results obtained for each parameter combination and the biological interpretation are given in Table

3.1. Observe that the operon being ON corresponds to a solution where $M = P = B = 1$ and being OFF corresponds to a solution where $M = P = B = 0$.

Parameters $x = (G_e, L_{em}, L_e)$	Fixed point(s) $(M, P, B, C, R, R_m, A, A_m, L, L_m)$	Operon ON or OFF ?
(0,0,0)	$(0,0,0,1,1,1,0,0,0,0)$	OFF
(0,1,0)	$(0,0,0,1,1,1,0,0,0,0)$	OFF
	$(1,1,1,1,0,0,0,1,0,1)$	ON
(0,1,1)	$(1,1,1,1,0,0,1,1,1,1)$	ON
(1,0,0)	$(0,0,0,0,1,1,0,0,0,0)$	OFF
(1,1,0)	$(0,0,0,0,1,1,0,0,0,0)$	OFF
(1,1,1)	$(0,0,0,0,1,1,0,0,0,0)$	OFF

Table 3.1: Fixed points of the *lac* operon Boolean network model for each choice of parameters.

From the table we see that the presence of extracellular glucose causes the model to predict the operon to be OFF, regardless of the other parameter values. If extracellular glucose is absent, the model predicts the operon to either be OFF, bistable, or ON when extracelluluar lactose is at low, medium, or high concentrations, respectively. Thus the model is consistent with biological observations.

3.4 A reduced model

We now use the method outlined in Chapter 2 to compute a reduced model F' involving only the variables M, L, and L_m. The new Boolean functions are as follows:

- For mRNA to be transcribed, we require the presence of the cAMP-CAP complex, which will be present if extracellular glucose is absent. Also, we require the repressor protein to be absent, which occurs if intracellular lactose is present. Thus the Boolean function is $f_M = \overline{G_e} \wedge (L \vee L_m)$.

- In order for intracellular lactose to be at a high concentration, permease protein must be present (which requires mRNA to be transcribed) and also extracellular lactose must be present. Furthermore, extracelluluar glucose must be absent. Thus the Boolean function is $f_L = M \wedge L_e \wedge \overline{G_e}$.

- In order for intracellular lactose to be at a medium concentration we need extracellular lactose to be present at a medium concentration and permease protein (which requires transcribed mRNA) or we need a high concentration of extracellular lactose that can diffuse into the cell.

Furthermore, we require the absence of extracellular glucose. Thus the Boolean function is $f_{L_m} = ((L_{em} \wedge M) \vee L_e) \wedge \overline{G_e}$.

The associated wiring diagram of the reduced model is shown in Figure 3.2. Note that the variables L, L_m and L_e, L_{em} have been grouped into single nodes.

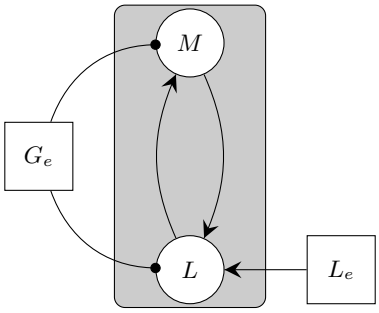

Figure 3.2: The wiring diagram of the reduced Boolean model of the *lac* operon.

To find the fixed points, we rename the variables

$$(M, L, L_m) = (x_1, x_2, x_3)$$

and convert the logical functions to polynomials in $\mathbb{F}_2[x_1, x_2, x_3]$ as seen below:

$$\begin{cases} f_M = \overline{G_e} \wedge (L \vee L_m) = M \\ f_L = M \wedge L_e \wedge \overline{G_e} = L \\ f_{L_m} = ((L_{em} \wedge M) \vee L_e) \wedge \overline{G_e} = L_m \end{cases} \iff \begin{cases} (G_e + 1)(x_2 + x_3 + x_2x_3) + x_1 = 0 \\ x_2 + x_1L_e(G_e + 1) = 0 \\ x_3 + (x_1L_{em} + L_e + x_1L_{em}L_e)(G_e + 1) = 0 \end{cases}$$

If we solve this system for the 6 parameter combinations $(G_e, L_{em}, L_e) \in \mathbb{F}_2^3$, where we do not consider the results with $L_{em} = 0, L_e = 1$ (since this case cannot occur), we obtain the results shown in Table 3.2. Note that the operon being ON corresponds to a solution where $M = 0$ and the operon is OFF if $M = 1$.

Parameters $x = (G_e, L_{em}, L_e)$	Fixed point(s) (M, L, L_m)	Operon ON or OFF ?
(0,0,0)	$(0, 0, 0)$	OFF
(0,1,0)	$(0, 0, 0)$	OFF
	$(1, 0, 1)$	ON
(0,1,1)	$(1, 1, 1)$	ON
(1,0,0)	$(0, 0, 0)$	OFF
(1,1,0)	$(0, 0, 0)$	OFF
(1,1,1)	$(0, 0, 0)$	OFF

Table 3.2: Fixed points of the *lac* operon Boolean network model for each choice of parameters.

From the table we see that we obtain precisely the same qualitative results with the reduced model as with the full model. That is, the reduced model predicts the operon to be ON only when extracellular glucose is absent and extracellular lactose is present in high concentrations, and also the reduced model exhibits bistability when extracellular glucose is absent and extracellular lactose is at a medium concentration.

Chapter 4

The *ara* operon

4.1 Biological background

In this section we give a biological treatment of the arabinose operon by describing the molecules and structures included in our model and examining how they interact with one another. The majority of this information is adapted from the extensive work done by the Schleif lab [Sch00].

In the bacterium *E. coli*, the five-carbon sugar L-arabinose can be metabolized to use as a carbon and energy source. The three structural genes involved in the metabolism are *araBAD*, whose transcription is controlled by the corresponding promoter p_{BAD}. The isomerase AraA converts L-arabinose to L-ribulose, the kinase AraB phosphorylates L-ribulose to L-ribulose-phosphate, and finally the epimerase AraD converts L-ribulose-phosphate to D-xylulose-phosphate, which then enters the pentose phosphate pathway. The transport of arabinose into the cell is controlled by two different transport systems, both located upstream from the *ara* operon. The *araE* gene produces AraE, a membrane bound protein that functions as a transporter in a low affinity transport system, and the *araFGH* genes produce three corresponding proteins that together form a high affinity transport system known as an ATP-binding cassette.

The expression of these structural genes is regulated by the protein AraC, which is unusual in the sense that it can function as either a repressor or an activator depending on the intracellular concentrations of arabinose. The cell maintains a small amount (approx. 20 molecules) of AraC at all times, which binds to two sites, I_1 and O_2, of the DNA strand that causes a DNA loop structure (Figure 4.1). This loop acts as a *repressor* of transcription of both the *araBAD* genes and the *araC*

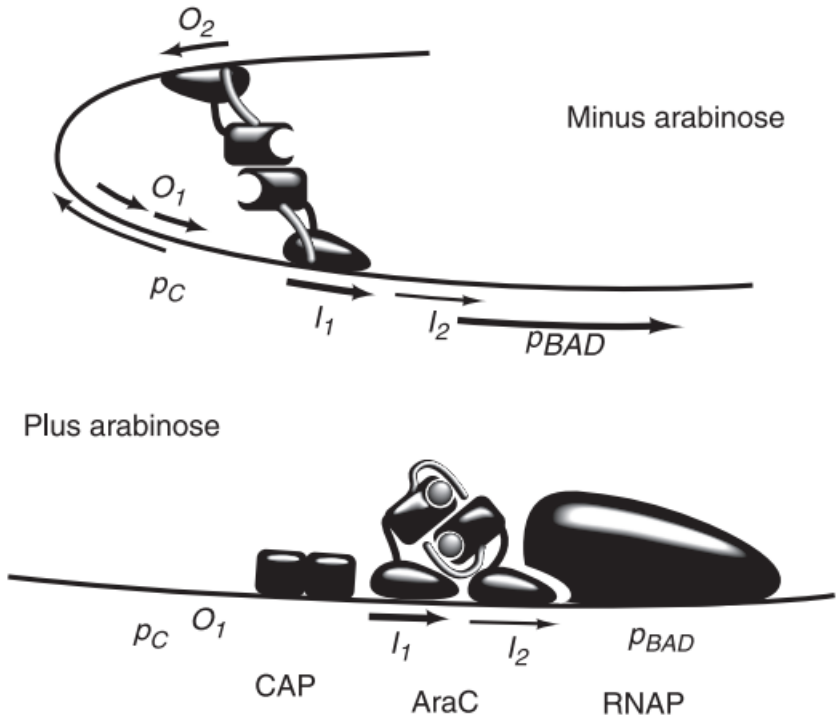

Figure 4.1: The ara operon in the absence and presence of arabinose.

gene by physically blocking RNA polymerase from attaching to either promoter sequence. Note that this method of repression is quite different than the method used in the *lac* operon, but the end result is the same. If extracellular arabinose is present, some molecules can be transported into the cell via passive transport. Once intracellular arabinose is available, these molecules will bind to the AraC protein and cause it to undergo a conformational change, which leads to the AraC protein dissociating from the O_2 operator site and subsequently binding to the I_2 site. In this arabinose-bound form, AraC now functions as an *activator* and induces the binding of RNA polymerase to the p_{BAD}, p_E, p_{FGH}, and p_C promoter regions.

Glucose also plays a role in regulating the transcription of the structural genes via catabolite repression. In addition to the arabinose-bound AraC protein, a second activator known as the cyclic AMP catabolite activator protein complex (or cAMP-CAP complex) must bind to the promoter region to cause the DNA to undergo further conformational changes to allow the binding of RNA polymerase. As in the *lac* operon, the presence of glucose inhibits the production of cAMP, thereby preventing transcription of the *ara* operon.

4.2 A Boolean model

Our Boolean model for the arabinose operon is comprised of variables and functions representing the presence or absence of molecules and structures involved in the gene regulatory network. We take as parameters the presence or absence of extracellular glucose (G_e), extracellular arabinose (A_e), and AraC protein that is not bound to arabinose ($AraC\_$). Note that we have split intracellular arabinose into two variables, A_m and A, representing low and high intracellular arabinose concentrations respectively, in order to maintain the Boolean structure while accurately capturing the biological mechanism. Also, two variables M_S and M_T representing mRNA are used, as the structural genes and transport genes are transcribed onto separate mRNA strands. Finally, we also distinguish between AraC protein that is unbound to arabinose (acting as a repressor) and bound to arabinose (acting as an activator) with the parameter $AraC\_$ and variable $AraC_+$ respectively.

4.2.1 Boolean variables

The Boolean variables are labeled as follows:

- $M_S = ara_{BAD}$ mRNA
- $M_T = ara_{EFGH}$ mRNA
- T = transport proteins (AraEFGH proteins)
- A = intracellular arabinose
- C = cAMP-CAP protein complex
- L = DNA loop
- Ara_+ = arabinose-bound AraC protein

where the subscript m denotes medium concentration and e denotes extracellular concentrations.

4.2.2 Derivation of Boolean functions

The derivation of the Boolean functions is as follows:

- For the structural genes' mRNA to be transcribed, we need the presence of the cAMP-CAP protein complex and the arabinose-bound AraC protein, as well as the absence of the DNA loop structure. Thus the Boolean function is $f_{M_S} = C \wedge Ara_+ \wedge \overline{L}$.

- For the transport genes' mRNA to be transcribed, we need the presence of the cAMP-CAP protein complex and the arabinose-bound AraC protein. Thus the Boolean function is $f_{M_T} = C \wedge Ara_+$.

- For the transport proteins to be produced, we need the presence of the transport genes' mRNA. Thus the Boolean function is $f_T = M_T$.

- For intracellular arabinose to be at a low concentration (or higher), we require the presence of either extracellular arabinose at a high concentration or a medium concentration of extracellular arabinose and the transport protein, and the absence of extracellular glucose. Thus the Boolean function is $f_{A_m} = ((A_{em} \wedge T) \vee A_e) \wedge \overline{G_e}$.

- For intracellular arabinose to be at a high concentration, we require the presence of extracellular arabinose and transport proteins, as well as the absence of extracellular glucose. Thus the Boolean function is $f_A = A_e \wedge T \wedge \overline{G_e}$.

- For the cAMP-CAP protein complex to be produced we require the absence of external glucose. Thus the Boolean function is $f_C = \overline{G_e}$.

- For the DNA loop to be formed, we require the presence of the AraC protein, but for the protein to not be bound to arabinose. Thus the Boolean function is $f_L = AraC_- \wedge \overline{AraC_+}$.

- For the arabinose-bound form of the AraC protein to be formed, we require the presence of the AraC protein and either low or high concentrations of intracellular arabinose. Thus the Boolean function is $f_{Ara_+} = Ara_- \wedge (A_m \vee A)$.

4.2.3 Wiring Diagram

The wiring diagram of our model is given below in Figure 4.2. We denote parameters by square nodes and variables by circular nodes. Edges between nodes with an arrow indicate positive interactions and circles indicate negative interactions. The shaded region represents intracellular space. Note that we have displayed A, A_m as a single node as well as A_e, A_{em}.

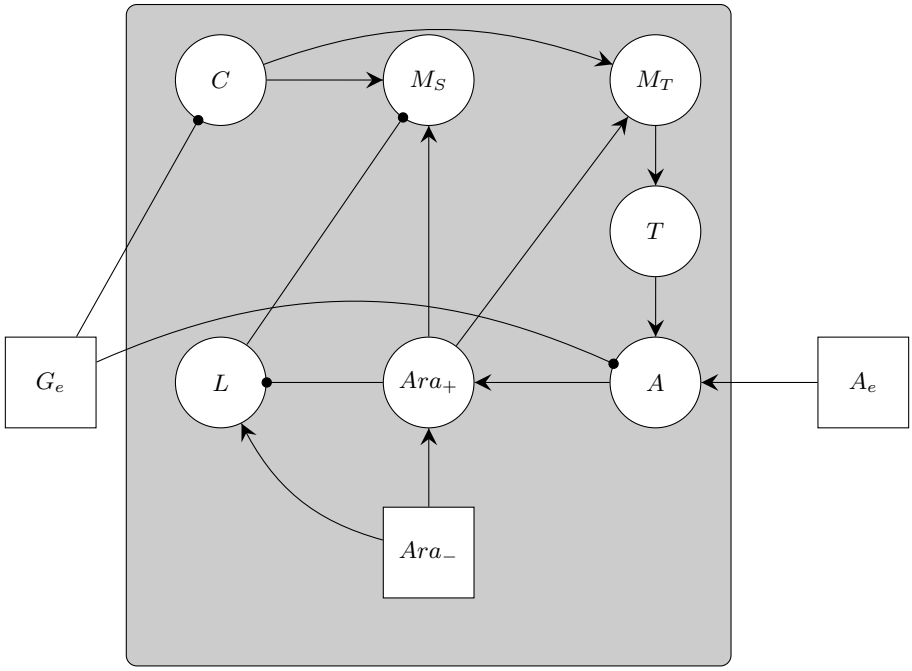

Figure 4.2: The wiring diagram of our proposed Boolean model of the *ara* operon.

4.3 Network dynamics

We can analyze the dynamics of our Boolean network model by specifying initial conditions on the parameters and then evaluating the Boolean functions at each time step to determine if the system eventually reaches a steady state in the form of a fixed point of the state space. This is equivalent to solving the system $\{f_{x_i} = x_i \mid i = 1, \ldots, 8\}$ where we have renamed our Boolean variables as follows:

$$(A_m, A, Ara_+, C, L, M_S, M_T, T) = (x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8).$$

To perform this analysis, we first convert our logical functions to a system of polynomial equations in $\mathbb{F}_2[x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8]$ using the methods described in Chapter 2. By working in this algebraic setting, we will be able to use computational algebra to assist in our network analysis.

The resulting system of polynomials F is

$$\begin{cases} f_{A_m} = ((A_{em} \wedge T) \vee A_e) \wedge \overline{G_e} = A_m \\ f_A = A_e \wedge T \wedge \overline{G_e} = A \\ f_{Ara_+} = (A_m \vee A) \wedge Ara_- = Ara_+ \\ f_C = \overline{G_e} = C \\ f_L = \overline{Ara_+} \wedge Ara_- = L \\ f_{M_S} = Ara_+ \wedge C \wedge \overline{L} = M_S \\ f_{M_T} = Ara_+ \wedge C = M_T \\ f_T = M_T = T \end{cases} \iff \begin{cases} x_1 + (A_{em}x_8 + A_e + A_{em}A_e x_8)(G_e + 1) = 0 \\ x_2 + x_8 A_e (G_e + 1) = 0 \\ x_3 + (x_1 + x_2 + x_1 x_2) Ara_- = 0 \\ x_4 + G_e + 1 = 0 \\ x_5 + (x_3 + 1) Ara_- = 0 \\ x_6 + x_3 x_4 (x_5 + 1) = 0 \\ x_7 + x_3 x_4 = 0 \\ x_8 + x_7 = 0 \end{cases}$$

For each of the 12 parameter combinations $(A_e, A_{em}, G_e, Ara_-) \in \mathbb{F}_2^4$, where we do not consider the cases with $A_{em} = 0, A_e = 1$ (since this case cannot occur), we must solve F using the algorithm described in Chapter 2, where the resulting solutions will be the fixed points of the network. The results obtained for each parameter combination and the biological interpretation are given in Table 4.1. Observe that the operon being ON corresponds to a solution where $M_S = M_T = 1$ and being OFF corresponds to a solution where $M_S = M_T = 0$.

Parameters $x = (A_e, A_{em}, G_e, Ara_-)$	Fixed point(s) $(A_m, A, Ara_+, C, L, M_S, M_T, T)$	Operon ON or OFF ?
(0,0,0,0)	$(0,0,0,1,0,0,0,0)$	OFF
(0,0,0,1)	$(0,0,0,1,1,0,0,0)$	OFF
(0,0,1,0)	$(0,0,0,0,0,0,0,0)$	OFF
(0,0,1,1)	$(0,0,0,0,1,0,0,0)$	OFF
(0,1,0,0)	$(0,0,0,0,0,0,0,0)$	OFF
(0,1,0,1)	$(0,0,0,1,1,0,0,0)$	OFF
	$(1,0,1,1,0,1,1,1)$	ON
(0,1,1,0)	$(0,0,0,0,1,0,0,0)$	OFF
(0,1,1,1)	$(0,0,0,0,1,0,0,0)$	OFF
(1,1,0,0)	$(1,0,0,1,0,0,0,0)$	OFF
(1,1,0,1)	$(1,1,1,1,0,1,1,1)$	ON
(1,1,1,0)	$(0,0,0,0,0,0,0,0)$	OFF
(1,1,1,1)	$(0,0,0,0,1,0,0,0)$	OFF

Table 4.1: Fixed points of our *ara* operon Boolean network model for each choice of parameters.

From the table we see that the presence of extracellular glucose causes the model to predict the operon to be OFF, regardless of the other parameter values. Similarly, the model predicts the

operon to be OFF in the absence of AraC protein or any level of extracellular arabinose. The only case in which the model predicts the operon to be exclusively ON is in the presence of high levels of extracellular arabinose and AraC protein and in the absence of extracellular glucose. Finally, if we have a medium concentration of extracellular arabinose and AraC protein, but extracellular glucose is absent, we observe bistability.

4.4 A reduced model

We now use the method outlined in Chapter 2 to compute a reduced model F' involving only the variables M_S, A, A_e, G_e, and $Ara\_$. The new corresponding Boolean functions are as follows:

- For mRNA to be transcribed we need the presence of the cAMP-CAP protein complex which requires absence of glucose. We also need absence of the loop which requires presence of arabinose-bound AraC protein which in turn requires presence of arabinose in at least medium concentration. Finally, we require the presence of arabinose-bound AraC protein as an activator which requires both the unbound AraC protein and intracellular arabinose in at least medium concentration. This simplifies to give the Boolean function $f_M = \overline{G_e} \wedge Ara\_ \wedge (A \vee A_m)$.

- For intracellular arabinose to be at a high concentration, we require the presence of extracellular arabinose at high concentration and the absence of glucose. We also require the presence of the transport proteins which requires the transcription of the transport protein mRNA which requires the presence of the cAMP-CRP protein complex which requires the absence of glucose. This simplifies to give the Boolean function $f_A = \overline{G_e} \wedge A_e \wedge M$.

- For intracellular arabinose to be at a medium concentration, we require the absence of glucose and either a high concentration of extracellular arabinose that can diffuse across the membrane or a medium concentration of extracellular arabinose and the transport proteins. For the transport protein to be present we also require transcription of the mRNA which in turn needs the presence of the cAMP-CAP protein complex. This simplifies to give the Boolean function $f_{A_m} = \overline{G_e} \wedge ((A_{em} \wedge M) \vee A_e)$.

The associated wiring diagram of the reduced model is shown in Figure 4.3.

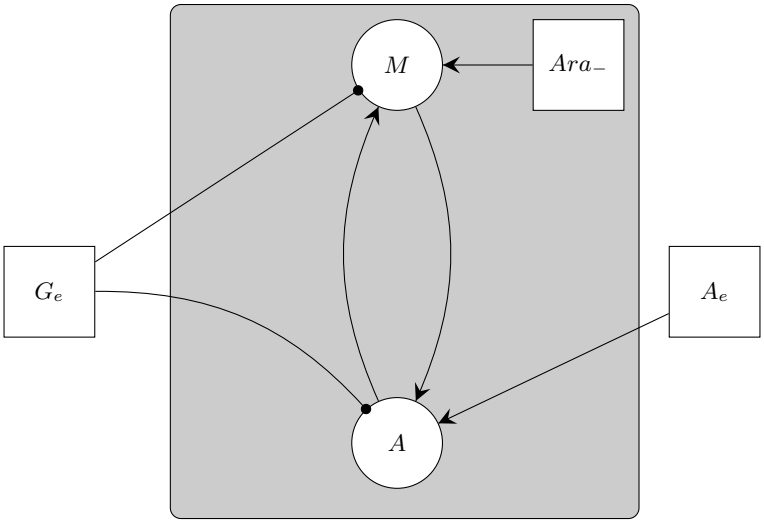

Figure 4.3: The wiring diagram of the reduced Boolean model of the *ara* operon.

To find the fixed points, we rename the variables

$$(M, A, A_m) = (x_1, x_2, x_3)$$

and convert the logical functions to polynomials in $\mathbb{F}_2[x_1, x_2, x_3]$ as seen below

$$\begin{cases} f_M = \overline{G_e} \wedge Ara_- \wedge (A \vee A_m) = M \\ f_A = \overline{G_e} \wedge A_e \wedge M = A \\ f_{A_m} = \overline{G_e} \wedge ((A_{em} \wedge M) \vee A_e) = A_m \end{cases} \iff \begin{cases} x_1 + (x_2 + x_3 + x_2x_3)(G_e + 1)Ara_- = 0 \\ x_2 + A_e x_1(G_e + 1) = 0 \\ x_3 + (A_{em}x_1 + A_e + A_{em}A_e x_1)(G_e + 1) = 0 \end{cases}$$

If we solve this system for 12 parameter combinations $(A_e, A_{em}G_e, Ara_-) \in \mathbb{F}_2^4$, where we have excluded the cases where $A_e = 1$ and $A_{em} = 0$, we obtain the results shown in Table 4.2. Note that the operon being ON corresponds to a solution where $M = 1$ and the operon is OFF if $M = 0$.

Parameters $x = (A_e, A_{em}, G_e, Ara_-)$	Fixed point(s) (M, A, A_m)	Operon ON or OFF ?
(0,0,0,0)	$(0, 0, 0)$	OFF
(0,0,0,1)	$(0, 0, 0)$	OFF
(0,0,1,0)	$(0, 0, 0)$	OFF
(0,0,1,1)	$(0, 0, 0)$	OFF
(0,1,0,0)	$(0, 0, 0)$	OFF
(0,1,0,1)	$(0, 0, 0)$	OFF
	$(1, 0, 1)$	ON
(0,1,1,0)	$(0, 0, 0)$	OFF
(0,1,1,1)	$(0, 0, 0)$	OFF
(1,1,0,0)	$(0, 0, 1)$	OFF
(1,1,0,1)	$(1, 1, 1)$	ON
(1,1,1,0)	$(0, 0, 0)$	OFF
(1,1,1,1)	$(0, 0, 0)$	OFF

Table 4.2: Fixed points of our *ara* operon Boolean network model for each choice of parameters.

From the table, we see that we obtain precisely the same qualitative results with the reduced model as with the full model. The reduced model predicts that the operon will be OFF if extracellular glucose is present or if AraC protein is absent. If extracelluluar glucose is absent, AraC protein is present, and extracellular arabinose is at low, medium, or high concentration, then the model predicts the operon to be OFF, bistable, and ON, respectively.

Chapter 5

Conclusions

Mathematical modeling of biological systems has been popular for decades, and the modeling of gene regulatory networks is no exception. However, until recently, most formulations were continuous models that could obscure biological insight beneath technical details. With the advent of Boolean modeling techniques came clearer qualitative interpretations of the network dynamics of transcriptional gene expression, providing both an increased interest in these modeling techniques from the biological community and a fruitful new area of research for mathematicians.

In this thesis, we have extended the work done by Stigler and Veliz-Cuba in their Boolean network model of the *lac* operon by providing a model for the *ara* operon. This model incorporates the unique features of the arabinose operon, (such as the use of both positive and negative inducible control mechanisms and the use of DNA looping), and still exhibits the expected biological behavior for every combination of parameters of extracellular environmental conditions. The reduced model also exhibits the same dynamics, and shows that the core components of the network are the *araBAD* mRNA and arabinose. Furthermore, both models manage to capture the bistability of the network when arabinose is at a medium concentration.

Based on the success of the Boolean framework in modeling inducible operons, in the future it would be sensible to apply these techniques to negative repressible operons such as the tryptophan and arginine operons. Negative repressible operons provide a unique challenge in modeling, as many of them use a mechanism known as attenuation in addition to the usual repressor/activator binding mechanism. In the process of attenuation, transcription is interrupted prematurely before the RNA polyermase finishes transcribing the structural genes. This stoppage occurs due to differential folding

of the DNA which is dependent on the concentration of a regulatory molecule. In the case of the *trp* operon, this molecule is tryptophan itself, leading to the additional complication that events occurring during translation are affecting the transcription process. Furthermore, more complex regulatory networks such as the glutamine synthetase (*gln*) operon could benefit from this Boolean network approach [UNBM83]. The *gln* operon has several added difficulties. Multiple regulatory genes (*glnG, glnL*) are present within a single operon, where *glnG* controls expression of the structural gene directly and *glnL* regulates the protein product of *glnG*. There also exist additional regulatory proteins which are both activators and repressors. Finally, the operon contains three promoters instead of the usual single promoter.

Future work could also focus on other mechanisms of genetic regulation. Most Boolean network modeling of gene regulatory networks has focused on operons, but operons are only commonly found in prokaryotic organisms. Eukaryotic organisms possess other common methods of controlling gene expression which could benefit from these qualitative modeling techniques [BKCC03]. For example, RNA interference (or RNA silencing) involves small RNA molecules that can interfere with transcription and translation in a variety of ways, but currently only continuous models exist and they fail to explain certain key features of the silencing behavior [BMA03]. Perhaps Boolean models could not only provide a better interpretation of the regulatory behavior, but in addition could be used to predict biological function in some cases where the mechanism or regulatory molecule is unknown.

Finally, it is still necessary to find an algebraic method of detecting and identifying limit cycles of size greater than one in the state space of a Boolean network. The current use of Gröbner bases allows us to find fixed points in the state space, but we have no guarantee that other network behaviors are not possible without directly observing the state space diagram. In the case of the *lac* and *ara* operon, the size of the network allows us to enumerate every initial state and compute the trajectory taken in the state space to verify the absence of limit cycles that are not fixed points, but this is clearly not a feasible method for larger, more complex networks. Therefore it is desirable to develop an algorithm that will allow the use of computational algebra to find these limit cycles.

Appendices

Appendix A Sage code

A.1 Refined Model Code

The code below defines the polynomial ring $\mathbb{F}_2[x_1, \ldots, x_8]$, specifies the parameter values for $(A_e, A_{em}, G_e, Ara_-)$, computes the ideal I corresponding to our system of polynomial equations, and then computes the Groebner basis B corresponding to I.

```
P.<x1,x2,x3,x4,x5,x6,x7,x8> = PolynomialRing(GF(2),8,order='lex');
Ae = 0; Aem = 0; Ge = 0; ara = 1;
I = ideal(x1+(Aem*x8+Ae+Aem*Ae*x8)*(Ge+1), x2+x8*Ae*(Ge+1), x3+(x1+x2+x1*x2)*ara,
x4+Ge+1, x5+(x3+1)*ara, x6+x3*x4*(x5+1), x7+x3*x4, x8+x7);
B = I.groebner.basis();
```

We have included three sample outputs corresponding to the parameter combinations $(0, 0, 0, 1)$, $(1, 1, 0, 1)$, and $(0, 1, 0, 1)$. The returned Grobner bases correspond to the operon being OFF, ON, and bistable respectively.

1. `B = [x1, x2, x3, x4 + 1, x5 + 1, x6, x7, x8]`
2. `B = [x1 + 1, x2 + 1, x3 + 1, x4 + 1, x5, x6 + 1, x7 + 1, x8 + 1]`
3. `B = [x1 + x8, x2, x3 + x8, x4 + 1, x5 + x8 + 1, x6 + x8, x7 + x8]`

A.2 Reduced Model Code

The code below defines the polynomial ring $\mathbb{F}_2[x_1, x_2, x_3]$, specifies the parameter values for $(A_e, A_{em}, G_e, Ara_-)$, computes the ideal I corresponding to our system of polynomial equations, and then computes the Groebner basis B corresponding to I.

```
P.<x1,x2,x3> = PolynomialRing(GF(2),3,order='lex');
Ae = 0; Aem = 0; Ge = 0; ara = 1;
I = ideal(x1+(Ge+1)*ara*(x2+x3+x2*x3),x2+(Ge+1)*Ae*x1,x3+(Aem*x1+Ae+Aem*Ae*x1)*(Ge+1));
B = I.groebner.basis();
```

We have included three sample outputs corresponding to the parameter combinations $(0,0,0,1)$, $(1,1,0,1)$, and $(0,1,0,1)$. The returned Grobner bases correspond to the operon being OFF, ON, and bistable respectively.

1. `B = [x1, x2, x3]`
2. `B = [x1 + 1, x2 + 1, x3 + 1]`
3. `B = [x1 + x3, x2]`

Bibliography

[BKCC03] William J Blake, Mads Kærn, Charles R Cantor, and James J Collins. Noise in eukaryotic gene expression. *Nature*, 422(6932):633–637, 2003.

[BKL+75] Kevin Bertrand, Laurence Korn, Frank Lee, Terry Platt, Catherine L Squires, Craig Squires, and Charles Yanofsky. New features of the regulation of the tryptophan operon. *Science*, 189(4196):22–26, 1975.

[BMA03] Carl T Bergstrom, Erin McKittrick, and Rustom Antia. Mathematical models of rna silencing: unidirectional amplification limits accidental self-directed reactions. *Proceedings of the National Academy of Sciences*, 100(20):11511–11516, 2003.

[C+70] Francis Crick et al. Central dogma of molecular biology. *Nature*, 227(5258):561–563, 1970.

[CLO92] David Cox, John Little, and Donal O'shea. *Ideals, varieties, and algorithms*, volume 3. Springer, 1992.

[DB08] Maria Davidich and Stefan Bornholdt. The transition from differential equations to boolean networks: a case study in simplifying a regulatory network model. *Journal of Theoretical Biology*, 255(3):269–277, 2008.

[EB68] WOLFGANG EpSTEIN and Jonathan R Beckwith. Regulation of gene expression. *Annual review of biochemistry*, 37(1):411–436, 1968.

[GS71] Jack Greenblatt and ROBERT Schleif. Arabinose c protein: regulation of the arabinose operon in vitro. *Nat New Biol*, 233(40):166–170, 1971.

[JM61] François Jacob and Jacques Monod. On the regulation of gene activity. In *Cold Spring Harbor Symposia on Quantitative Biology*, volume 26, pages 193–211. Cold Spring Harbor Laboratory Press, 1961.

[JPSM05] François Jacob, David Perrin, Carmen Sánchez, and Jacques Monod. L'opéron: groupe de gènes à expression coordonnée par un opérateur [cr acad. sci. paris 250 (1960) 1727–1729]. *Comptes rendus biologies*, 328(6):514–520, 2005.

[LBZ+00] Harvey Lodish, Arnold Berk, S Lawrence Zipursky, Paul Matsudaira, David Baltimore, James Darnell, et al. *Molecular cell biology*, volume 4. WH Freeman New York, 2000.

[LHA92] Chung-Dar Lu, John E Houghton, and Ahmed T Abdelal. Characterization of the arginine repressor from salmonella typhimurium and its interactions with the carab operator. *Journal of molecular biology*, 225(1):11–24, 1992.

[MHS86] Katherine Martin, Li Huo, and Robert F Schleif. The dna loop model for ara repression: Arac protein occupies the proposed loop sites in vivo and repression-negative mutations lie in these same sites. *Proceedings of the National Academy of Sciences*, 83(11):3654–3658, 1986.

[Pie12] Benjamin A Pierce. *Genetics: A conceptual approach*. Macmillan, 2012.

[Sch00] Robert Schleif. Regulation of the l-arabinose operon of escherichia coli. *Trends in Genetics*, 16(12):559–565, 2000.

[SM08] Moisés Santillán and Michael C Mackey. Quantitative approaches to the study of bistability in the lac operon of escherichia coli. *Journal of The Royal Society Interface*, 5(Suppl 1):S29–S39, 2008.

[Stu96] Bernd Sturmfels. *Gröbner bases and convex polytopes*, volume 8. American Mathematical Soc., 1996.

[UNBM83] Shizue Ueno-Nishio, Keith C Backman, and Boris Magasanik. Regulation at the glnl-operator-promoter of the complex glnalg operon of escherichia coli. *Journal of bacteriology*, 153(3):1247–1251, 1983.

[VCS11] Alan Veliz-Cuba and Brandilyn Stigler. Boolean models can explain bistability in the lac operon. *Journal of Computational Biology*, 18(6):783–794, 2011.

[Yil12] Necmettin Yildirim. Mathematical modeling of the low and high affinity arabinose transport systems in escherichia coli. *Molecular BioSystems*, 8(4):1319–1324, 2012.