



5-2005

Coupling Analysis, Simulation, and Experimentation in Natural and Engineered Biological Systems at the Molecular Scale

Derek W. Austin

University of Tennessee - Knoxville

Recommended Citation

Austin, Derek W, "Coupling Analysis, Simulation, and Experimentation in Natural and Engineered Biological Systems at the Molecular Scale." PhD diss., University of Tennessee, 2005.
https://trace.tennessee.edu/utk_graddiss/665

This Dissertation is brought to you for free and open access by the Graduate School at Trace: Tennessee Research and Creative Exchange. It has been accepted for inclusion in Doctoral Dissertations by an authorized administrator of Trace: Tennessee Research and Creative Exchange. For more information, please contact trace@utk.edu.

To the Graduate Council:

I am submitting herewith a dissertation written by Derek W. Austin entitled "Coupling Analysis, Simulation, and Experimentation in Natural and Engineered Biological Systems at the Molecular Scale." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Electrical Engineering.

Michael L. Simpson, Major Professor

We have read this dissertation and recommend its acceptance:

Benjamin Blalock, Chris Cox, Syed Islam, Greg Peterson

Accepted for the Council:

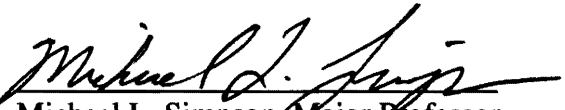
Dixie L. Thompson

Vice Provost and Dean of the Graduate School

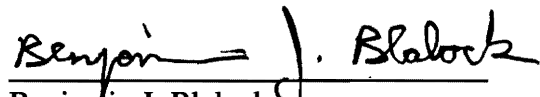
(Original signatures are on file with official student records.)

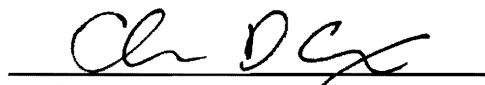
To the Graduate Council:


I am submitting herewith a dissertation written by Derek William Austin entitled "Coupling Analysis, Simulation, and Experimentation in Natural and Engineered Biological Systems at the Molecular Scale." I have examined the final paper copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Electrical Engineering.



Michael L. Simpson, Major Professor

We have read this dissertation
and recommend its acceptance:

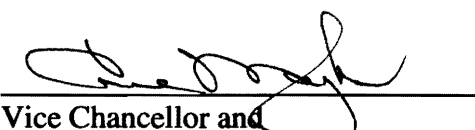

Benjamin J. Blalock


Chris D. Cox


Syed K. Islam


Greg D. Peterson

Accepted for the Council:


Vice Chancellor and
Dean of Graduate Studies

**Coupling Analysis, Simulation, and Experimentation in
Natural and Engineered Biological Systems
at the Molecular Scale**

A Dissertation
Presented for the
Doctor of Philosophy
Degree
The University of Tennessee, Knoxville

Derek William Austin

May 2005

Thesis
2005b
.A88

Copyright © 2005 by Derek William Austin

All rights reserved

Dedicated to Claire

with all my love

Acknowledgments

There are many people and organizations to whom I am grateful for making this dissertation research a success. I cannot thank enough my advisor, Dr. Michael Simpson, and my committee for your guidance and for the resources that you provided at both the University of Tennessee and Oak Ridge National Laboratory (ORNL). I would like to acknowledge the Center for Environmental Biology and the Biology Microscopy Laboratory at the University of Tennessee, where all of the experimental components of this research were performed. I am also particularly grateful to the Center for Nanophase Materials Sciences at ORNL, the Defense Advanced Research Projects Agency, the Keck Futures Initiative, and the National Science Foundation for the funding that they have provided to support me and this research.

This work would not have been made possible without the assistance of Michael Allen, John Wilgus, Chris Cox, James McCollum, John Dunlap, Roy Dar, and Laura Morris-Edwards. Michael and John Wilgus, thank you for the endless supply of cells and plasmids that you constructed. To Chris and James, I thank you for your assistance with the stochastic simulations. John Dunlap, thank you for teaching me everything I know about confocal microscopy. Roy, thanks for hanging in there with me as we manually tracked cells for hours. And to Laura, thank you for editing this manuscript.

Lastly, I want to thank my family and friends. You have been there for me through all the ups and downs, and I would not have made it this far without your eternal love and support.

Abstract

Cellular functions are controlled by genetic regulatory networks called gene circuits. Recently, there has been much interest in how gene circuits deal with or even exploit stochastic fluctuations in molecular species within the cellular environment. Through a coupling of analysis and simulation with experimentation, this dissertation work furthers the understanding of gene circuit noise behavior and makes significant contributions to the analytical and experimental tools that are currently available for the study and design of natural and synthetic gene circuits. In this study, models are developed for unregulated and autoregulated gene circuits. Results from the analysis are compared to computer simulations and experimental results. Exact stochastic simulations show that the derived analytical expressions are valid even for populations as low as 10 molecules, despite linear approximations made by the analysis. The experimental portion of this work presents a novel method for acquiring *in vivo* measurements of real-time gene expression. The techniques developed here are used to report the very first measurements of frequency content in gene circuit noise and verify theoretical predictions that negatively autoregulated gene circuits shift their noise spectra up to higher frequency. Through measured shifts in noise spectra, these frequency measurements can also reveal subtle and condition-dependent regulatory pathways. Measured noise spectra may also permit *in vivo* estimation of gene circuit kinetic rate parameters.

Table of Contents

Chapter	Page
1. Introduction to Systems Biology	1
1.1 Modeling Genetic Circuits.....	2
1.1.1 Mathematical Models.....	3
1.1.2 Exact Stochastic Simulation	7
1.2 Literature Review.....	8
1.2.1 Engineered Gene Circuits	9
1.2.2 Stochasticity in Genetic Circuits.....	14
1.3 Problem Statement.....	20
1.4 Scope of Dissertation	21
2. Primer on Molecular Biology	23
2.1 Cellular Components	23
2.1.1 Genes.....	24
2.1.2 RNA	26
2.1.3 Ribosomes and Proteins.....	26
2.2 Genetic Processes.....	27
2.2.1 Transcription	27
2.2.2 Translation	29
2.2.3 Gene Regulation.....	30
2.2.4 Replication	31
2.2.5 Transformation.....	34
2.3 Antibiotics.....	35
3. Frequency-Domain Analysis	36
3.1 Transfer Functions	36
3.1.1 Bode Plots	40
3.1.2 Effects of Negative Feedback	43
3.2 Random Signals	45
3.2.1 Autocorrelation	46
3.2.2 Energy Spectral Density	47
4. Gene Circuit Analysis and Simulation.....	51
4.1 Modeling and Noise Analysis	52
4.1.1 Unregulated Gene Circuit	52
4.1.2 Autoregulated Gene Circuit	63
4.2 Simulation Results	70
4.2.1 Unregulated Gene Circuit	72
4.2.2 Autoregulated Gene Circuit	75
4.3 Discussion.....	79

5.	<i>In Vivo</i> Measurements of Gene Expression	83
5.1	Sample Preparation	83
5.2	Data Acquisition	87
5.3	Image and Data Processing.....	94
6.	Analysis of <i>In Vivo</i> Noise Measurements.....	102
6.1	Experiments and Conditions	102
6.2	Autocorrelations.....	104
6.2.1	Temperature Effects.....	105
6.2.2	Protein Half-life	109
6.2.3	Negative Feedback.....	112
6.3	Noise Bandwidth.....	115
6.4	Gene Circuit Models	117
6.5	Estimation of Model Parameters.....	120
7.	Conclusions and Future Work	122
	References.....	126
	Appendices.....	137
	Vita.....	155

List of Tables

Table		Page
4.1	ODEs and steady-state equations for the unregulated gene circuit.....	55
4.2	Parameters used to model the unregulated gene circuit.....	62
4.3	Parameters used to model the autoregulated gene circuit.....	69
4.4	Biospreadsheet parameters used for the unregulated gene circuit.....	73
4.5	Comparison of steady-state populations for the unregulated gene circuit.....	74
4.6	Comparison of steady-state populations for the autoregulated gene circuit.....	76
4.7	Biospreadsheet parameters used to study the effects of time delays in the autoregulated gene circuit.....	77
5.1	Settings used for laser confocal microscopy.....	90
6.1	List of experiments and conditions.....	103

List of Figures

Figure	Page
1.1 Model of an autoregulated gene.....	4
1.2 Toggle switches	10
1.3 A genetic logic gate	11
1.4 Structure and output response of the repressilator	13
1.5 Increase in noise bandwidth due to negative feedback	17
1.6 Effects of feedback on protein variation	18
2.1 Components of a prokaryotic cell	25
2.2 Double-helix structure of DNA	25
2.3 Expression of a gene	28
2.4 Transcription of DNA to synthesize mRNA.....	28
2.5 Translation of mRNA to synthesize proteins.....	30
2.6 Regulation of the <i>lac</i> genes in <i>E. coli</i>	32
2.7 Replication of DNA	32
3.1 Block diagram of a system with input signal, $x(t)$, impulse response $h(t)$, and output, $y(t)$	37
3.2 Common-emitter inverting amplifier	39
3.3 Frequency-domain model of common-emitter amplifier at high frequency.....	39
3.4 High-frequency Bode plot of the transfer function for the common-emitter inverting amplifier with cutoff frequency f_c	41
3.5 Classical negative feedback system	44
3.6 Common-emitter amplifier with negative feedback	44
3.7 Gain-bandwidth tradeoff with addition of negative feedback	45
3.8 Sampling of two random signals and their autocorrelation functions	47
3.9 Energy spectral densities of the random signals in Figure 3.8	49
4.1 Unregulated gene circuit model with delays and protein dimerization	54
4.2 Electrical circuit equivalent for the unregulated gene circuit	57
4.3 Frequency-dependent functions for the biochemical processes in the unregulated gene circuit.....	59
4.4 Total noise ESD in the steady-state dimer population, S_d , and calculated contributions to S_d by each of the noise sources, $S_{j,k}$, for the unregulated gene circuit.....	62
4.5 Autoregulated gene circuit model with reversible operator-dimer binding.....	64
4.6 Electrical circuit equivalent for the autoregulated gene circuit	65
4.7 Frequency-dependent functions for the biochemical processes in the autoregulated gene circuit.....	67
4.8 Total noise ESD in the steady-state dimer population, S_d , and calculated contributions to S_d by each of the noise sources, $S_{j,k}$, for the autoregulated gene circuit.....	69

4.9	Simulated dimer population versus time at sampling rate $f_s = 0.1$ Hz.....	71
4.10	Representative noise ESD from ESS results created by appending spectrums from different sampling rates, f_s , to remove aliasing artifacts.....	72
4.11	Comparison of dimer noise ESD between the analytical model and ESS for the unregulated gene circuit.....	74
4.12	Comparison of dimer noise ESD between the analytical model and ESS for the autoregulated gene circuit.....	76
4.13	Overshooting of dimer steady-state population in time domain as total delay time is increased from 50 to 800 s for the autoregulated gene circuit.....	78
4.14	Peaking in the dimer noise ESD for the autoregulated gene circuit as total delay time is increased from 50 to 800 s.....	79
4.15	Bandwidth comparison of the unregulated and autoregulated gene circuits.....	80
4.16	Bode plot of calculated loop transmission, $T(f)$, for the autoregulated gene circuit when total delay time is increased from 50 to 800 s.....	81
5.1	Plasmids constructed for <i>in vivo</i> measurements of gene expression.....	85
5.2	Biological sample ready for microscopy.....	87
5.3	Primary components of a laser confocal microscope.....	88
5.4	Leica TCS SP2 laser confocal microscope.....	90
5.5	Before (a) and after (b) images for a 7-hour experiment using <i>E. coli</i> cells with pGFP-asv at 26°C.....	91
5.6	Stack of cropped images acquired for a single time sample.....	93
5.7	Tracking a single cell through selected frames of a movie.....	96
5.8	Measured cell area and exponential growth for a single cell.....	97
5.9	Selection of thresholds used to segment cells from their image background.....	98
5.10	Image segmentation used to determine total cell area per image.....	99
5.11	Noise measured for a single cell.....	100
5.12	Noise in GFP concentrations for the 8 labeled cells of Figure 5.5.....	100
6.1	Individual cell ($\Phi_i(\tau)$) and composite ($\Phi_c(\tau)$) normalized ACFs of GFP noise for pGFP-asv at 22°C.....	106
6.2	Cell growth versus temperature for pGFP-asv cells.....	107
6.3	Individual cell ($\Phi_i(\tau)$) and composite ($\Phi_c(\tau)$) normalized ACFs of GFP noise for pGFP-asv at different temperatures.....	108
6.4	Individual cell ($\Phi_i(\tau)$) and composite ($\Phi_c(\tau)$) normalized ACFs of noise for GFP half-life variants at 25-26°C.....	109
6.5	Fluorescence and growth comparison of pGFP-asv and pGFP-WT.....	111
6.6	ATc induction curve for pTetR-GFP.....	113
6.7	Individual cell ($\Phi_i(\tau)$) and composite ($\Phi_c(\tau)$) normalized ACFs of noise for an unregulated (ASV22), control (ASV26+ATc), and autoregulated (TetR26) gene circuit.....	114
6.8	Bandwidth of gene circuit noise found from half-correlation times.....	116
6.9	Simple models for (a) variant pGFP and (b) pTetR-GFP gene circuits.....	117
6.10	Extrinsic and intrinsic contributions to GFP noise.....	119
6.11	Individual cell ($\Phi_i(\tau)$), composite ($\Phi_c(\tau)$), and theoretical ($\Phi_{theory}(\tau)$) normalized ACFs of GFP noise for the ASV22 experiment.....	120

List of Abbreviations

ACF	autocorrelation function
ATc	anhydrotetracycline
CME	chemical master equation
Di	dimerized protein
DNA	deoxyribonucleic acid
ESD	energy spectral density
ESS	exact stochastic simulation
FD	frequency-domain
GFP	green fluorescent protein
mRNA	messenger RNA
ODE	ordinary differential equation
P	protein
RNA	ribonucleic acid
RNAP	RNA polymerase
WT	wild-type

Nomenclature

b	burst rate
δ	dilution rate from cell growth
f	frequency
γ_p	protein decay rate
γ_r	mRNA decay rate
λ	half-life
T_d	cell volume doubling time
τ_p	translation delay time
τ_r	transcription delay time

Chapter 1

Introduction to Systems Biology

Biological systems perform many complex functions that give cells the ability to sense, communicate, navigate, or even fabricate nanoscale materials [1-4]. All of these advanced behaviors are controlled by genetic circuits and gene regulatory networks. Understanding the structure, function, and dynamics of gene networks is the primary aim of systems biology. This interdisciplinary field includes biologists, chemists, computer scientists, engineers, mathematicians, and physicists, who are all working together to develop a shared language that describes genetic systems. Research in systems biology also promotes a strong coupling between analysis, simulation, and experimentation [5]. Analysis and simulation help define experiments that should be conducted; likewise, experimental results provide feedback that may lead to model refinements or reveal new functions [6].

The potential of systems biology research is limitless. Development of the necessary analytical, computational, and experimental tools may make it possible to predict the behavior of connected genetic circuits [7]. An ensemble of such tools could lead to

breakthroughs with significant applications in gene therapy and medicine. For example, genes responsible for the development of a genetic disorder or a viral disease could be turned off (or on) by engineered regulatory networks. The ability to design genetic networks could also lead to realization of biomolecular computers that process information within cells and produce desired cellular behavior [8, 9]. All of these advancements will accompany improvements in computation such as distributed computing and grid-based simulation [10]. In the pharmaceutical industry, modeling and simulation may one day successfully predict the side effects of drugs before clinical trials are even initiated [11]. The medical field is not the only area that will benefit from progress made in systems biology research. On the contrary, a deeper understanding of genetic networks found in biological systems may also guide the development of future architectures for electronic circuits, parallel computing, control theory, and systems design, just to name a few.

1.1 Modeling Genetic Circuits

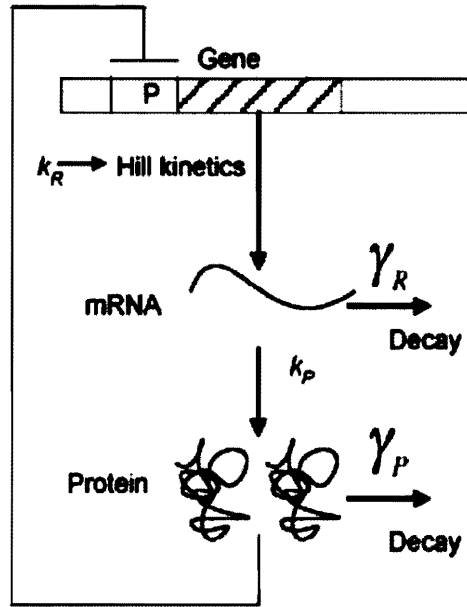
The functions encoded by genes and genetic networks are carried out by chemical reactions. These chemical reactions describe processes such as the production, binding, and degradation of molecules within the cell. Modeling a gene circuit begins by applying biological knowledge to list all of the involved chemical reactions. This step can prove difficult because every gene and protein in the network might not be known or characterized. Next, the reactions within the network are translated into mathematical representations. At this point, the model can be analyzed or simulated. Analysis

provides insight to system response and how adjustment of circuit parameters affects the overall behavior. On the other hand, simulation rapidly calculates system behavior as a function of time and may quickly uncover interesting phenomena that are not revealed by analysis alone. These two methods of modeling are complementary and should be used in conjunction with one another.

1.1.1 Mathematical Models

Ordinary differential equations (ODEs) are a common tool for analyzing dynamic systems in science and engineering. To understand how ODEs are applied to genetic circuits, consider the extensively studied case in which a gene is negatively regulated by the protein that it produces. In the absence of repressor protein (P), the RNA polymerase ($RNAP$) can bind to the promoter (Pro) and transcribe mRNA (R). The mRNA can then be translated by ribosomes (Rib) to create protein. The mRNA is also degraded and decays (noted below as $*$) with a particular half-life, λ . The created protein, which typically degrades much slower than mRNA, can then repress gene expression by binding to the operator (Op) and blocking RNAP. Figure 1.1 shows a model for the biological processes described here. Note that in this figure P does not stand for protein but rather the promoter region adjacent to the gene. A partial list of the chemical reactions involved in this circuit are given below, along with their corresponding reaction rates, r_i , derived from the law of mass action:





Source: Simpson, et al., *Proc. Natl. Acad. Sci. USA* **100**, pp. 4551-4556, 2003.

Figure 1.1. Model of an autoregulated gene.



where k_i is the rate constant for the reaction, and the terms in brackets represent molar concentrations of the chemical species. Finally, the rate equations can be rearranged to describe the changes in the molar concentrations of each species as a function of time.

The equations for mRNA and protein can be simplified to [12]:

$$\frac{d[R]}{dt} = h(P) - k_3[R], \quad (1.7)$$

$$\frac{d[P]}{dt} = k_7[R] - k_6[P], \quad (1.8)$$

where it has been assumed that reactions 1.4 and 1.5 are relatively fast compared to the other reactions, and thus, are at quasi-steady state (i.e., $d[Op]/dt = 0$). The reactions for the production of *RNAP* and *Rib* (not listed above) are usually considered to be in equilibrium so that $[RNAP]$ and $[Rib]$ are constant. Therefore, $[RNAP]$ has been absorbed into the term $h(P)$ and likewise $[Rib]$ is included in k_7 . In Figure 1.1, the parameter k_R symbolizes $h(P)$, γ_R represents k_3 , and k_P and γ_P correspond to k_7 and k_6 , respectively. The function $h(P)$ captures the behavior of how the production of mRNA, or transcription, is controlled by the protein population. This regulation function models the switching transition between two states (operator bound or unbound) and is typically described by the sigmoidal Hill expression [12]:

$$h(P) = \frac{k_{max}}{1 + \left(\frac{P}{k_d}\right)^n}, \quad (1.9)$$

where k_{max} is the maximum reaction rate, k_d is the protein concentration at which the reaction rate equals half of k_{max} , and n is known as the Hill coefficient, which is positive for repressive feedback and negative for inductive feedback.

To accurately model molecular interactions within cells, fluctuations (i.e., noise) in both reaction rates and populations of chemical species must be taken into account. A

popular way to accomplish this is to append an additive white noise source, $\eta(t)$, to each of the ODEs [13]. Equations 1.7 and 1.8 become:

$$\frac{d[R]}{dt} = h(P) - k_3[R] + \eta(t), \quad (1.10)$$

$$\frac{d[P]}{dt} = k_7[R] - k_6[P] + \eta(t). \quad (1.11)$$

These stochastic differential equations are referred to as Langevin equations. Despite the linearizations and approximations associated with this technique, careful application of Langevin analysis has provided many insights into gene circuit behavior [14-16]. Results of these analyses are discussed in more detail in the literature review of Section 1.2.

Implicit in the ODE models described to this point is the fact that they are continuous and deterministic. For dynamic systems with large numbers of molecules, the error between discrete and continuous behavior can be safely overlooked. However, cellular systems involving just a few, discrete molecules and events can cause fluctuations in gene expression yielding nondeterministic behavior [17]. Conventional deterministic models may not predict these probabilistic outcomes [18]. A standard for explicitly treating discrete stochastic behavior is the Chemical Master Equation (CME). The CME describes how the probability of any state in a system evolves over time as a result of the chemical reactions that are allowed to occur. For M chemical reactions with initial condition (\mathbf{X}_0, t_0) , the CME can be formulated as [13]:

$$\frac{\partial}{\partial t} P(\mathbf{X}, t | \mathbf{X}_0, t_0) = \sum_{j=1}^M [a_j(\mathbf{X} - \mathbf{v}_j) P(\mathbf{X} - \mathbf{v}_j, t | \mathbf{X}_0, t_0) - a_j(\mathbf{X}) P(\mathbf{X}, t | \mathbf{X}_0, t_0)], \quad (1.12)$$

where \mathbf{X} is a vector that indicates the number of molecules of each chemical species, a_j is the propensity function (similar to probability) that a reaction will occur, and \mathbf{v}_j is the change in the number of molecules due to the occurrence of a reaction. By definition, the CME is a Markov chain since the current state of the system simply depends on the previous state. It has been shown that as the number of molecules becomes larger, the CME asymptotically approaches the Langevin equation with the noise term, $\eta(t)$, set appropriately [13]. However, satisfying this condition may not be possible for some genetic circuits due to the small number of involved molecules.

1.1.2 Exact Stochastic Simulation

Given the large number of chemical species and possible reactions in even the simplest gene networks, analytical and numeric solutions to the CME are difficult to derive. As an alternative approach to this problem, Gillespie developed an algorithm for simulating coupled chemical reactions called the first reaction method [19], also known as exact stochastic simulation (ESS). Gillespie's algorithm guarantees that the resulting distribution of \mathbf{X} (Eq. 1.12) at time t will approach the distribution implied by the CME when enough simulations have been performed. The Gillespie algorithm works by randomly generating a time interval τ for when the next reaction will take place and determines which reaction is most likely to occur based on relative probabilities and the current state of the molecular populations in the system. After the selected reaction occurs, the state of the system is updated and the time is incremented by τ . This procedure is repeated over and over until the total simulation time elapses.

Certainly the greatest advantage of ESS is that it can be used to simulate any genetic circuit described by chemical reactions. The Gillespie algorithm also produces inherent noise that accurately reflects the stochastic behavior of the biochemical processes. This property gives ESS the ability to reveal bifurcation or switching states that might otherwise go undetected by deterministic ODE models [18]. The results are accurate even at low molecular populations. Unfortunately, execution of the Gillespie algorithm is computationally demanding and time consuming for realistic genetic networks with many reactions. To address this issue, several improvements have been made to the ESS algorithm to increase its efficiency. Gibson and Bruck introduced dependency graphs that can greatly reduce the number of calculations to be performed [20]. Gillespie later developed an enhancement called τ -leaping, which combines multiple reactions into a single step [21]. Although these techniques can reduce simulation time considerably, they sometimes come at the cost of loss in accuracy. Therefore, caution must be exercised when applying these techniques if one is investigating stochastic effects and system performance of genetic circuits.

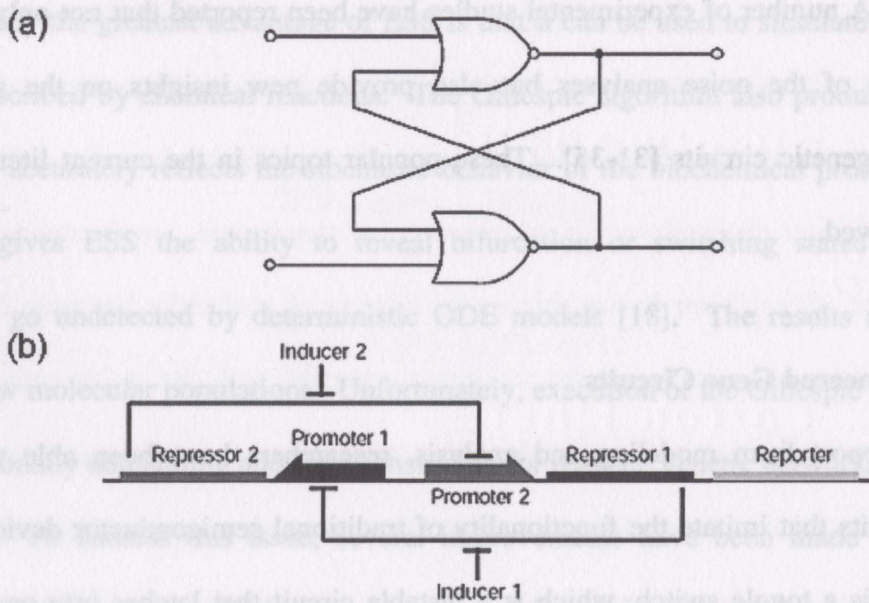
1.2 Literature Review

In several cases, mathematical models and simulations have guided the design of synthetic gene circuits that mimic silicon-based electronic devices such as toggle switches, logic gates, and oscillators [22-25]. Inspired by the ability of cells to operate precisely even when laden with noisy internal components, there has been a growing appreciation for analyzing and modeling the stochastic properties of gene networks [14-

16, 26-30]. A number of experimental studies have been reported that not only confirm some results of the noise analyses but also provide new insights on the stochastic behavior of genetic circuits [31-35]. These popular topics in the current literature are briefly reviewed.

1.2.1 Engineered Gene Circuits

With support from modeling and analysis, researchers have been able to design genetic circuits that imitate the functionality of traditional semiconductor devices. One such device is a toggle switch, which is a bistable circuit that latches into one state or another depending upon a given input stimulus. Because the system remains in its state even after the input has been removed, the toggle switch is a 1-bit memory that remembers a stimulus event. In electrical engineering, an example of such a device is the RS flip-flop, shown in Figure 1.2(a). Gardner *et al.* constructed a genetic toggle switch in *Escherichia coli* bacteria cells by using the mutual repression of two genes to achieve bistability [23]. Figure 1.2(b) illustrates the design of this genetic latch circuit. If the input stimulus is inducer 1, then transcription of repressor 2 is blocked. This scenario latches the system in its high state because repressor 1 and the observable reporter (green fluorescent protein, GFP) continue to be expressed. In a similar manner, the system can be switched to its low state by presenting inducer 2 as the input. Using ODE analysis, Gardner *et al.* revealed how the bistable region of operation was affected by circuit parameters: correct operation of the genetic toggle switch depended on strong promoters, effective transcriptional repression, and relatively equal synthesis and decay rates for the

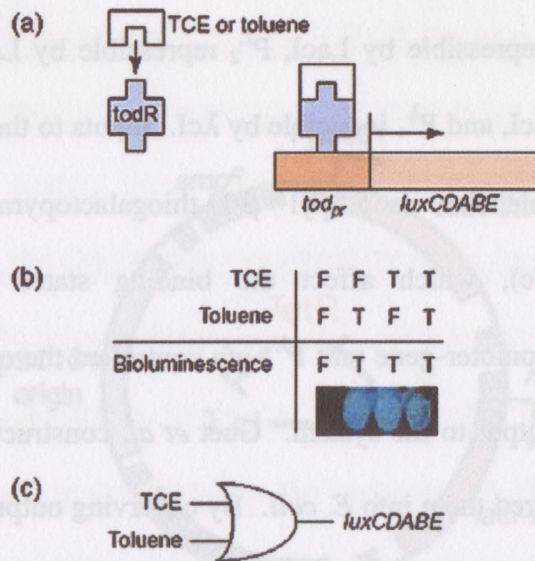


Source: Gardner, et al., *Nature* **403**, pp. 339-342, 2000.

Figure 1.2. Toggle switches. (a) Electronic RS flip-flop consisting of two NOR gates and (b) a genetic toggle switch constructed from two mutually repressive genes.

proteins of repressor 1 and repressor 2. The mathematical predictions were validated by experimental evidence. *E. coli* cells, transformed with engineered plasmids (self-replicating DNA molecules) containing promoters and genes with appropriate characteristics, could be switched between two states using suitable chemical inducers.

A couple of approaches for designing logic gates in cellular systems have been successfully implemented. First, an example of a rational design is presented for a logical OR gate that was realized by using a promoter that responds to two different inducers. Simpson *et al.* employed this method in whole cells using trichloroethylene (TCE) and toluene to activate the *tod* promoter (Figure 1.3(a)), which up-regulated the



Source: Simpson, et al., *Trends Biotech* **19**, pp. 317-323, 2001.

Figure 1.3. A genetic logic gate. (a) Interaction of TCE or toluene with the *tod* promoter to induce expression of the *luxCDABE* genes and produce bioluminescence, (b) the logical truth table describing the input-output relationships, and (c) schematic diagram of the resulting OR gate.

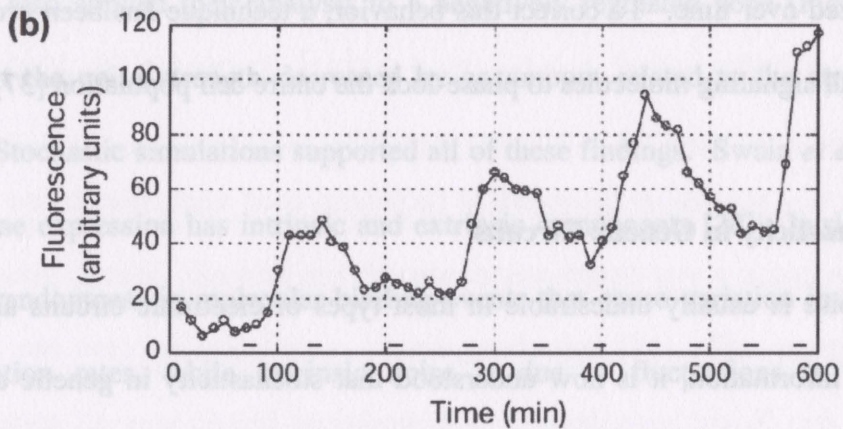
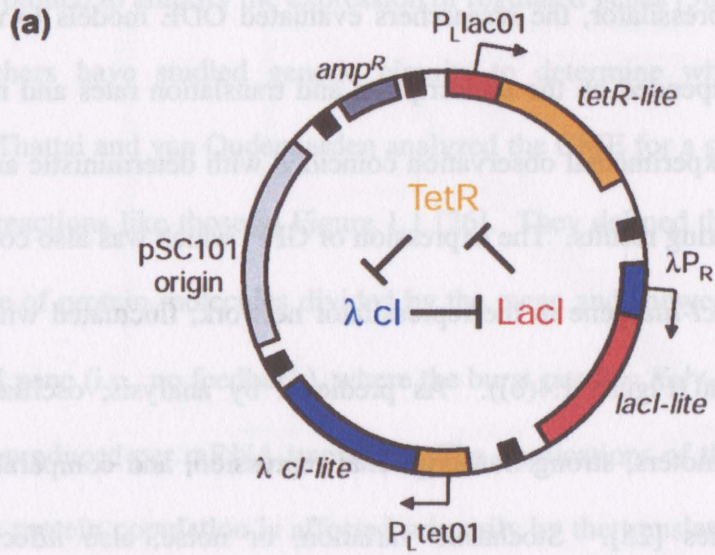
expression of the *luxCDABE* genes, resulting in production of bioluminescence [24].

Figure 1.3(b) shows that if one or both of the inducer molecules were present (True), then the output genes were expressed (True) and the cells produced light. Otherwise, if both inducer molecules were absent (False), then the output was not expressed (False). Figure 1.3(c) shows a schematic diagram of this logical OR gate with inputs and output.

As an alternative to rational design, Guet *et al.* used combinatorial methods to generate a library of logic circuits by shuffling the connectivity of genetic networks [36]. The genetic circuits were integrated into plasmids made of promoter-gene units constrained to the structure $P_i - lacI - P_j - \lambda cI - P_k - tetR$, where P_i , P_j , and P_k were one

of five promoters: P^{L_1} repressible by LacI, P^{L_2} repressible by LacI, P^T repressible by TetR, P^λ repressible by λ cI, and P^λ_+ inducible by λ cI. Inputs to the genetic circuits were the small inducer molecules isopropyl β -D-thiogalactopyranoside (IPTG) and anhydrotetracycline (ATc), which affect the binding states of LacI and TetR, respectively. A fourth promoter-gene unit P^λ_{-gfp} controlled the expression of GFP and acted as an observable output to the system. Guet *et al.* constructed 30 of the possible 125 plasmids and transferred them into *E. coli*. By observing output fluorescence for the four different input conditions, with or without IPTG and with or without ATc, the researchers discovered genetic logic circuits that included NAND, NOR, and NOT IF gates. Analysis of the experimental data also showed that the 30 characterized networks simplified to just 13 distinct connectivity diagrams, or circuit topologies [36]. These results indicate that, in addition to being a useful tool for creating genetic networks whose underlying working mechanisms are unknown, combinatorial techniques can also contribute to the developing knowledge of gene circuit function and structure.

Clocks are used in many man-made and living systems to coordinate the timing of events. Using a closed-loop cascade of transcriptional repressor circuits, Elowitz and Leibler implemented an artificial clock in *E. coli* nicknamed the repressilator [25]. The design of the repressilator is shown in Figure 1.4(a). The behavior of this genetic circuit is analogous to an electronic ring oscillator. Expression of the *tetR-lite* gene produces TetR that represses the P_{LacO-1} promoter and shuts down the production of λ cI. Repression of λ cI frees the λP_R promoter and permits expression of the *lacI-lite* gene. Finally, the production of LacI represses the P_{LacO-1} promoter, which stops the expression



Source: Elowitz, et al., *Nature* **403**, pp. 335-338, 2000.

Figure 1.4. Structure and output response of the repressilator. (a) Plasmid structure of a genetic oscillator using three repressive genes and (b) measured oscillations in output fluorescence of GFP from a single cell.

of TetR, allowing the production of λ C_I to go back up, and so on. To guide the construction of the repressilator, the researchers evaluated ODE models to realize that oscillatory behavior depended on the transcription and translation rates and mRNA and protein decay rates. Experimental observation coincided with deterministic analysis and showed several fascinating results. The expression of GFP, which was also controlled by the promoter for the *λ cl-lite* gene in the repressilator network, fluctuated with a typical period of 150 minutes (Figure 1.4(b)). As predicted by analysis, oscillations were favored by strong promoters, strong transcriptional repression, and comparable mRNA and protein decay rates [25]. Stochastic variation, or noise, also affected circuit performance and caused the phase of oscillators in the cell population to become desynchronized over time. To correct this behavior, a technique has been proposed that employs small signaling molecules to phase-lock the entire cell population [37].

1.2.2 Stochasticity in Genetic Circuits

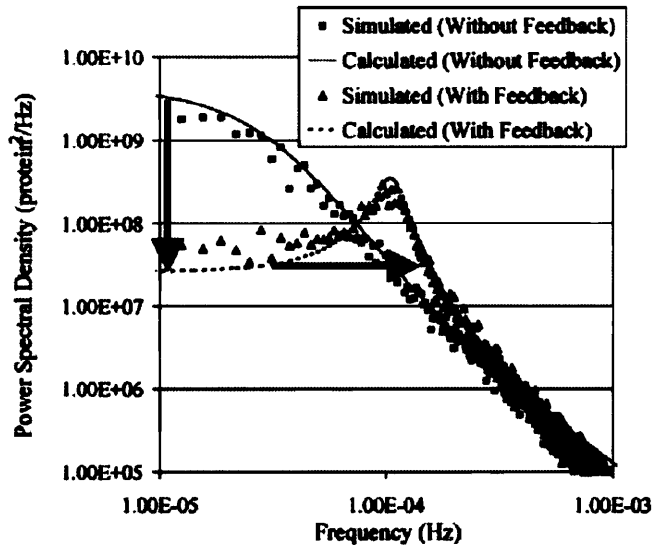
While noise is usually undesirable in most types of electronic circuits and systems that process information, it is now understood that stochasticity in genetic circuits can actually play an important functional role in network behavior and decision-making processes [29]. For example, stochastic fluctuations in molecular population largely control the lysis-lysogeny decision in λ -phage infected cells [18]. A similar functional role was proposed for noise in bacterial quorum sensing systems, where the inherent biomolecular noise creates a redundancy, and a quorum is sensed even if there is

destructive interference in cell-cell signaling [15]. It has also been proposed that noise could be exploited to amplify the expression of regulated genes [38].

Researchers have studied genetic circuits to determine where sources of noise originate. Thattai and van Oudenaarden analyzed the CME for a single gene circuit with first-order reactions like those in Figure 1.1 [26]. They defined the noise strength, v , as the variance of protein molecules divided by the mean and showed that $v \approx 1 + b$ for an unregulated gene (i.e., no feedback), where the burst rate $b = K_p/\gamma_R$ is the average number of proteins produced per mRNA transcript. The implications of their results are that the variation in protein population is affected primarily by the translation step and that noise strength is greater when the protein level is approaching steady-state [26]. These researchers also applied their analysis to a negatively regulated gene (Figure 1.1) and showed that the noise strength decreased by an amount related to the strength of the feedback. Stochastic simulations supported all of these findings. Swain *et al.* argue that noise in gene expression has intrinsic and extrinsic components [28]. Intrinsic noise is caused by randomness in molecular binding events that cause variation in transcription and translation rates, while extrinsic noise is due to fluctuations in the cellular environment, the cell cycle, regulatory proteins, and populations of molecules such as RNAP and ribosomes. Using CME analysis, Swain *et al.* defined noise strength as the standard deviation of protein molecules divided by the mean and showed that the main source of intrinsic noise is transcription when the burst rate is greater than 2 [28]. Disagreement between the findings of these two research groups is likely due to interpretations of their results. While the translation process alone makes a contribution

to noise in the protein population, translation appears to amplify the transcription noise at higher burst rates.

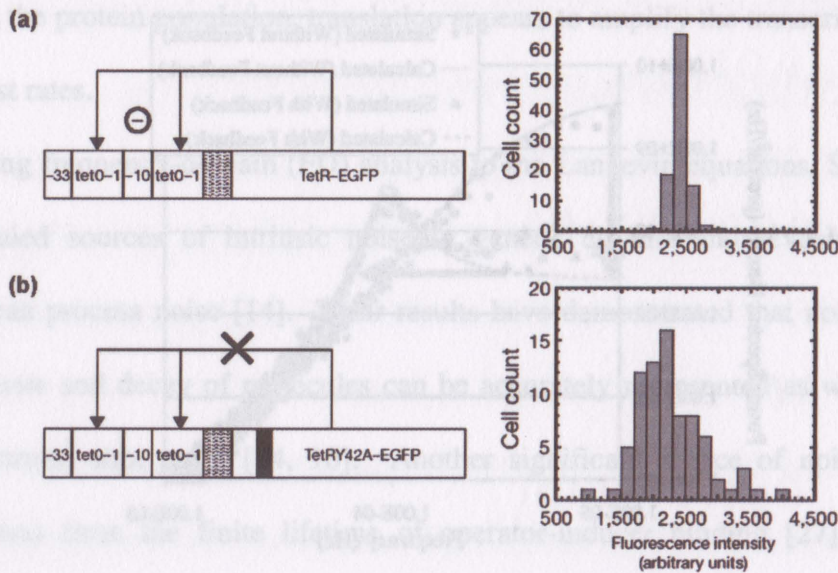
Applying frequency-domain (FD) analysis to the Langevin equations, Simpson *et al.* have modeled sources of intrinsic noise in genetic circuits and revealed how gene networks can process noise [14]. Their results have demonstrated that noise associated with synthesis and decay of molecules can be accurately represented as wideband (i.e., white spectrum) shot noise [14, 16]. Another significant source of noise in genetic circuits arises from the finite lifetime of operator-inducer binding [27]. Using FD analysis, Simpson *et al.* showed that this operator noise is band-limited and contributes to the noise in mRNA synthesis [16]. To understand how genetic circuits can process inherent noise, researchers have analyzed the noise of output protein in terms of its power spectral density (PSD), which describes how noise is distributed across the frequency spectrum. Cox *et al.* used FD analysis to show how reversible chemical reactions (e.g., dimerization of protein) either whiten or band-limit the noise PSD depending on the rates of the forward and reverse reactions [15]. A significant prediction of the analysis by Simpson *et al.* was that gene circuits with negative feedback cause a reduction in the noise power at certain frequencies and shift noise up to higher frequencies (i.e., increase noise bandwidth) as shown in Figure 1.5 [14]. Samoilov *et al.* have demonstrated that some genetic networks can act as low-pass and band-pass filters [39]. Results from both of these research groups suggest how some genetic networks may have evolved for the purpose of filtering or reducing noise as it propagates through cascaded gene circuits.



Source: Simpson, et al., *Proc. Natl. Acad. Sci. USA* **100**, pp. 4551-4556, 2003.

Figure 1.5. Increase in noise bandwidth due to negative feedback.

Several experiments have been conducted to determine the origins and properties of noise in genetic circuits. Becskei and Serrano demonstrated how a negatively regulated gene circuit can decrease variation in protein production [31]. The genetic constructs shown in Figure 1.6 were cloned into plasmids and transformed into *E. coli*. The output protein was TetR fused with GFP, which was observed in hundreds of cells under a fluorescence microscope. The fluorescence histogram in Figure 1.6(a) shows the small variance seen in GFP for the regulated gene. The genetic construct was then mutated so that the TetR-GFP protein lost its affinity for the $P_{\text{LetO-1}}$ promoter, disabling the feedback mechanism. As expected, the production of GFP by the unregulated gene exhibited higher levels of noise, or variation, as shown in Figure 1.6(b).



Source: Becskei, et al., *Nature* **405**, pp. 590-593, 2000.

Figure 1.6. Effects of feedback on protein variation. (a) Negatively autoregulated gene with histogram showing small variation in GFP and (b) increased variation, or noise, when the feedback is removed.

Experiments by Ozbudak *et al.* supported the theoretical predictions by Thattai and van Oudenaarden that translation is the strongest contributor to intrinsic noise [32]. By inserting a single *gfp* gene into the chromosome of *Bacillus subtilis* behind the tightly regulated promoter P_{spac} , Ozbudak *et al.* modulated the gene's transcriptional efficiency by controlling the concentration of inducer, IPTG. To modulate the translational efficiency, point mutations were created in the *gfp* gene to alter the ribosomal binding site. Flow cytometry was then used to measure the fluorescence of thousands of cells expressing GFP. With noise strength defined as variance over mean, these researchers

showed that noise strength was indeed strongly dependent on translational efficiency and that transcriptional efficiency had a very weak effect on the noise strength [32].

A clever experiment designed by Elowitz *et al.* facilitated measurements of intrinsic and extrinsic noise in gene expression [33]. This research group created strains of *E. coli* with a chromosome containing a gene for cyan (CFP) and yellow (YFP) fluorescent protein. Both of these genes had the same type of promoter, which could be induced with IPTG to vary the transcriptional efficiency. When intrinsic noise was low, the expression of CFP and YFP were equal, causing the cells to appear yellow. As intrinsic noise increased, the expression of the two genes became uncorrelated and cells appeared red or green due to the increased production of one protein over the other. Although translation effects were not studied here by Elowitz *et al.*, their results also showed that intrinsic noise increased strongly as the transcriptional efficiency decreased [33].

To study stochastic gene expression in eukaryotic cells, Blake *et al.* performed experiments similar to those of Ozbudak *et al.* using *Saccharomyces cerevisiae* (yeast) [34]. They created two separate genetic circuits using two different promoters that regulated the expression of GFP. Both genetic circuits showed similar behavior: low noise strength (defined here as variance over mean) at low transcriptional efficiency, a strong increase in noise at 20-40% transcriptional efficiency, and then a gradual decrease to a low noise state at 100% induction [34]. By modifying the *gfp* gene, Blake *et al.* also modulated the translational efficiency of the gene. Their results showed that increases in translation efficiency caused only slight increases in the noise strength, in contrast to the results by Ozbudak *et al.* for prokaryotic cells. Using the technique of Elowitz *et al.*,

Raser and O'Shea measured intrinsic and extrinsic noise in yeast. They found that intrinsic noise strength was gene-specific and not dependent on the absolute rate of expression [35]. Thus, their measurements and model assert that noise in prokaryotic and eukaryotic gene expression is not that dissimilar. Instead, they proposed a model in which differences in noise are due to relative rates of promoter activation and the rate of the subsequent transcription process that follows. Clearly, much more work needs to be done to sort out the noise contributions made to protein production by the many individual cellular processes.

1.3 Problem Statement

Innovations in technology over the past decade have accelerated genomics research to yield entire DNA sequences of species including bacteria, rice, and even humans [40-42]. There are now ongoing efforts to maintain databases (Ref. [43], for example) that list identified genes along with their function, if known, and any observed interactions with larger genetic networks. Despite all the progress made in DNA sequencing and gene identification, the functions of many genes remain unknown. For example, nearly 40% of the protein-coding genes in *E. coli* have no attributed function [40]. Microarray experiments and studies of cellular response to input perturbations have aided researchers in deducing underlying interconnections of genes in biochemical pathways [6, 44, 45]. However, the structure and function of many genetic networks have yet to be determined and a wiring diagram alone is not enough to understand the properties and behavior of an entire system. From an engineering perspective, biological systems are the products of

fundamental gene circuits that are connected together to form large elaborate networks [46]. Thus, understanding the complex behavior of genetic networks requires a thorough knowledge of their components (DNA, RNA, proteins, etc.) and how these components interact with each other. Ultimately, an understanding of the structure-function relationships in genetic networks may lead to the development of new engineered systems that mimic the robustness, adaptiveness, and fault-tolerance seen in cellular systems [47, 48]. To realize this great challenge, more engineering tools and methods are needed for modeling, simulating, and experimenting with genetic networks in biological systems.

1.4 Scope of Dissertation

By coupling analysis and simulation with experimentation, this dissertation research contributes to the understanding of gene circuit behavior by developing models for genetic networks and demonstrating an experimental technique that reveals information about underlying genetic processes using the spectral content of gene circuit noise. Chapter 2 provides the reader with a short primer on some fundamental concepts in molecular biology, while Chapter 3 briefly reviews some techniques in frequency-domain analysis of electrical circuits and random signals. Next, Chapter 4 describes the development of models for regulated and unregulated gene circuits in prokaryotic cells, the frequency-domain noise analyses, and results from computer simulations. Chapter 5 explains experimental methods used to obtain *in vivo* noise measurements of gene expression in bacteria cells. Then, Chapter 6 presents spectral analysis of experimental

data to elucidate information about the structure and behavior of the genetic circuits used in this work. Finally, concluding remarks are given in Chapter 7 along with suggestions for future possible work.

Chapter 2

Primer on Molecular Biology

Before modeling and analyzing gene circuits, it is helpful to review some fundamental concepts in molecular biology. This chapter describes many of the cellular components and genetic processes that are encountered numerous times in this dissertation work. Covering details of the biology here should help the reader to better understand the simplifications and rationales of the models formulated in Chapter 4 and the experimental designs developed in Chapter 5.

2.1 Cellular Components

A cell is a small unit of living matter enclosed in a plasma membrane. Cells can be classified into two types: prokaryotes and eukaryotes. Prokaryotic cells, such as bacteria, are single-celled organisms that lack a nucleus. In contrast, eukaryotic cells have a nucleus and can be single-celled (e.g., yeast) or multicellular (e.g., plants and animals). While the two types of cells share many similarities, the focus of attention hereafter is on prokaryotic cells, which are used in the work presented in subsequent

chapters of this dissertation. As shown in Figure 2.1, prokaryotic cells have a single cytoplasmic compartment just a few micrometers in length that contains all of their biomolecules.

2.1.1 Genes

The instruction sets for all cellular processes are genes, which are stored in the form of double-stranded molecules known as deoxyribonucleic acid (DNA). This familiar double-helix structure, shown in Figure 2.2, resembles a spiral ladder constructed from linked nucleotides [49]. Each nucleotide is composed of a 5-carbon sugar (deoxyribose) attached to a phosphate group and a nitrogen-containing sidegroup, known as a base. These bases can be of four different types: adenine, cytosine, guanine, and thymine, which correspond to four distinct nucleotides referred to as A, C, G, and T, respectively (Figure 2.2). Since the formation of base pairs occurs such that A bonds only to T and C bonds only to G, the two twisted strands of DNA are said to be complementary to each other. A group of ordered nucleotide pairs along the DNA comprises a gene, and the sequence of the nucleotides defines the function of the gene. A typical bacterial cell may contain roughly 1000-5000 genes on its chromosome, whose uncoiled length can be as long as a millimeter. These genes are passed down to progeny cells and may evolve over time.

While the primary genes required for cell growth are stored on chromosomal DNA, cells may also carry genes on plasmids. Plasmids are double-stranded circular loops of DNA [50], typically denoted with a lower-case “p” followed by an abbreviated name of

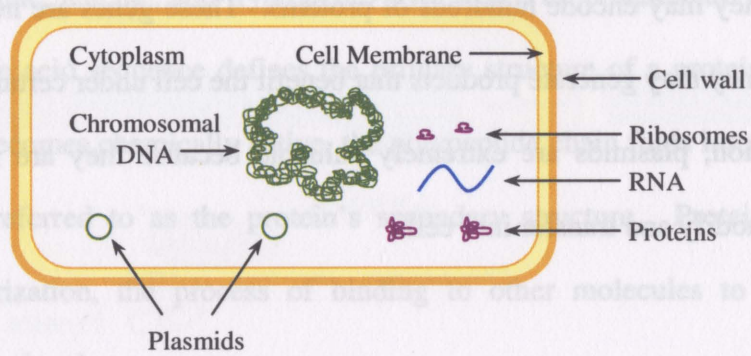
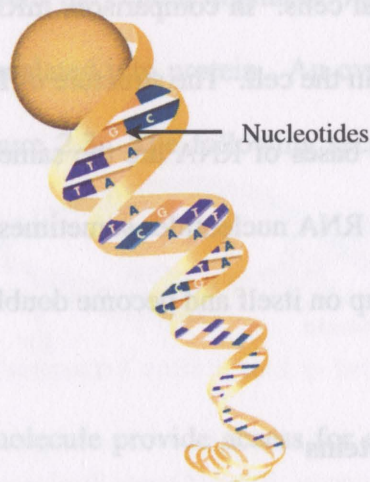


Figure 2.1. Components of a prokaryotic cell.



Source: U.S. Department of Energy Human Genome Program

Figure 2.2. Double-helix structure of DNA.

the derivative (e.g., pBR322). Plasmids may carry enough genes to encode just a few proteins or they may encode hundreds of proteins. These genes are not essential to cell growth, but they may generate products that benefit the cell under certain conditions. For experimentation, plasmids are extremely valuable because they are relatively easy to genetically modify and transfer into cells.

2.1.2 RNA

Ribonucleic acid (RNA) can be formed from DNA by replacing deoxyribose with the sugar ribose. Named with respect to their functions, the three types of RNA are messenger RNA (mRNA), ribosomal RNA (rRNA), and transfer RNA (tRNA). All three types of RNA are made by an enzyme, or catalytic protein, called RNA polymerase (RNAP). Due to their high rate of synthesis and stability, rRNA and tRNA make up 95% of the total RNA in bacterial cells. In comparison, mRNA has a shorter lifetime as it is degraded more rapidly within the cell. The structure of RNA is similar to single-stranded DNA [51]. The nucleotide bases of RNA are the same as DNA except that thymine is replaced by uracil (U). The RNA nucleotides sometimes link (A bonds to U and C bonds to G) causing RNA to fold up on itself and become double-stranded.

2.1.3 Ribosomes and Proteins

Most DNA sequences in bacteria are dedicated to genes encoding proteins, and it is proteins that do most of the work of a cell. Large biomolecules called ribosomes, composed of subunits of proteins and RNA, are responsible for protein synthesis [52].

Proteins consist of polypeptide chains of amino acids. There exist 20 different amino acids (e.g., glutamine, leucine, and tryptophan) with which all proteins are created. The order of the amino acid sequence defines the primary structure of a protein. Typically before a protein becomes chemically active, the polypeptide chain folds into a distinctive shape, which is referred to as the protein's secondary structure. Proteins may also undergo multimerization, the process of binding to other molecules to form larger functional macromolecules.

2.2 Genetic Processes

Cells function as biochemical factories that continuously process and convert molecules such as amino acids and sugars. For cells to perform, genes must be expressed; that is, the cell's genetic instructions must be read in order to synthesize needed proteins. This concept forms the central dogma of biology: information in DNA is transcribed into RNA to be translated into protein. An overview of the processes for gene expression is shown in Figure 2.3. The following sections describe transcription and translation in further detail.

2.2.1 Transcription

The grooves along a DNA molecule provide access for enzymes that transcribe the encoded genes. The transcription process begins when RNAP binds to a nucleotide sequence at the beginning of the gene called the promoter region. The RNAP separates the double-helix and then moves along a single strand of DNA, as shown in Figure 2.4.

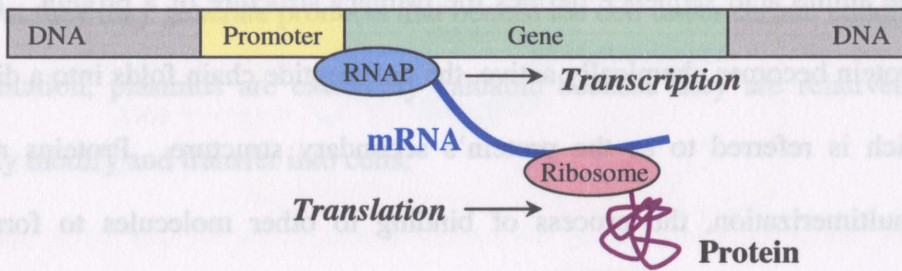
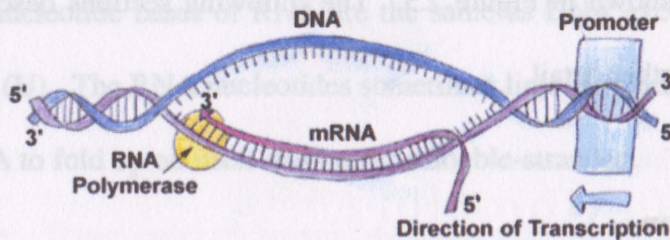


Figure 2.3. Expression of a gene.



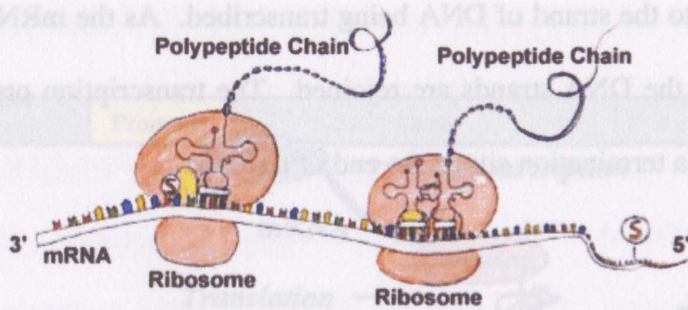
Source: <http://library.thinkquest.org>

Figure 2.4. Transcription of DNA to synthesize mRNA.

While it advances, the RNAP assembles individual nucleotides into a strand of RNA that is complementary to the strand of DNA being transcribed. As the mRNA is constructed, it peels away, and the DNA strands are rejoined. The transcription process stops when the RNAP reaches a termination site at the end of the gene.

2.2.2 Translation

Translation is the process in which protein molecules are assembled. In prokaryotic cells, ribosomes can bind to the mRNA and begin the process of translation as soon as a strand of mRNA extends from the RNAP. In contrast, eukaryotic cells must transport mRNA out of their nucleus, where their DNA is stored, before the mRNA can be translated. Ribosomes translate the mRNA's sequence of nucleotides into polypeptide chains of linked amino acids, as shown in Figure 2.5. First, a ribosome binds to mRNA at a translation initiation region. The ribosome then moves along the mRNA and reads one codon at a time. A codon is a set of three nucleotides, which represent a word of a symbolic genetic code. With just a few exceptions, this genetic code is universal to all species. Each code word (i.e., codon) specifies which amino acid the ribosome should append to the growing polypeptide chain. For example, the codon UGG corresponds to the amino acid tryptophan. A designated amino acid is brought to the ribosome by a tRNA that has the codon's complementary nucleotide sequence (anticodon). As suggested by Figure 2.5, multiple ribosomes may bind and translate the same mRNA once the previously bound ribosome has cleared the initiation region. In addition, an mRNA molecule may be polycistronic, meaning that it contains multiple translation



Source: <http://library.thinkquest.org>

Figure 2.5. Translation of mRNA to synthesize proteins.

initiation sites and encodes more than one protein. A ribosome continues to assemble a polypeptide chain until it reaches a stop codon, a sequence that does not encode for an amino acid. Finally, the protein is released from the ribosome. The protein may then undergo structural changes, such as folding, before reaching its final functional form.

2.2.3 Gene Regulation

For cells to function properly, they must make proteins at the right times in response to physiological and environmental conditions. Controlling the expression of genes is commonly referred to as gene regulation. Deemed to be a strategy for conserving resources within the cell, regulation is typically enforced at the transcriptional stage of gene expression [53]. Positive regulation, or induction, occurs when an inducer molecule

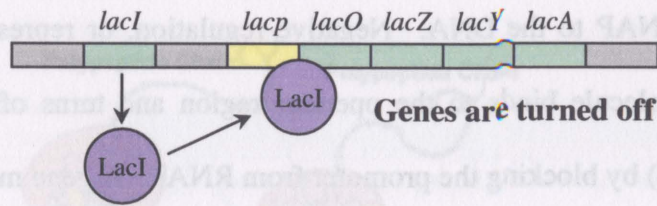
binds to an operator (part of the gene's promoter region) and enables gene expression by recruiting RNAP to the DNA. Negative regulation, or repression, takes place when a repressor molecule binds to the operator region and turns off the gene (i.e., prevents transcription) by blocking the promoter from RNAP. A gene may be autoregulated by its own protein product, or it may be regulated by several other proteins, which can result in the formation of elaborate genetic regulatory networks.

The *lac* genes in *E. coli* are part of a well-studied example of gene regulation [53]. These genes encode the enzymes for utilization of the sugar lactose. In the presence of lactose the *lac* genes are expressed because allolactose (converted from lactose) is an inducer for the *lac* promoter, *lacp* in Figure 2.6. Conversely, the *lac* genes (*lacO*, *lacZ*, *lacY*, *lacA*) are not expressed if lactose is unavailable. In the absence of lactose, the product of the *lacI* gene represses the *lac* genes by binding to one of the three *lacO* operators (o_1 , o_2 , o_3). This binding blocks RNAP, which in turn inhibits transcription of the remaining *lac* genes. However, when lactose is present, the inducer (allolactose) binds up free repressor molecules. This changes the conformation of the repressor so that it no longer binds to the operator, allowing RNAP to bind to the promoter (*lacp*) and transcribe the *lacO*, *lacZ*, *lacY*, and *lacA* genes.

2.2.4 Replication

Cell reproduction demands the replication of DNA. Replication begins with separation of the DNA strands using proteins called helicases. The point of separation is called the replication fork, shown in Figure 2.7. The DNA chromosomes of most

Absence of Lactose:



Presence of Lactose:

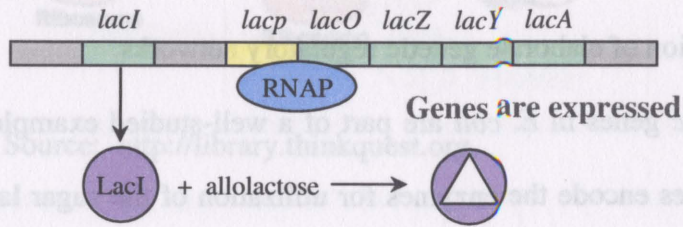
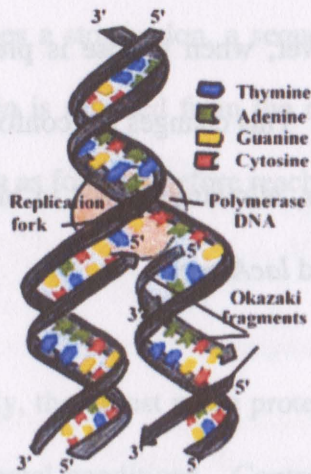


Figure 2.6. Regulation of the *lac* genes in *E. coli*.



Source: <http://library.thinkquest.org>

Figure 2.7. Replication of DNA.

bacteria are circular, but they are coiled up many times over to fit within the confines of the cell (Figure 2.1). Topoisomerases undo DNA coiling ahead of the replication fork, breaking the strands if necessary to uncoil them. Helix-destabilizing proteins keep the two single strands of DNA apart. DNA polymerases start at the chromosome's origin of replication, and then two replication forks move in opposite directions. The polymerases use each separated strand of DNA as a template to join (polymerize) deoxynucleotide complements together (Figure 2.7). Accessory proteins help keep the polymerase on the DNA strand and also perform editing of the base pairs (A-T, C-G). Replication is completed when the two replication forks meet at the other side of the circular chromosome, leaving the cell with two identical double-helix strands of DNA. This replication process is semiconservative in that each new chromosome contains one of the original single strands of DNA. After chromosome replication is completed, then the cell can divide [54]. During cell division, one DNA chromosome is passed to each of the daughter cells.

As a cell grows, any plasmids that it contains replicate autonomously. Plasmids have at least one origin of replication and regenerate just like chromosomal DNA. Often, plasmids encode just one of the proteins needed for initiating their own replication. Then the plasmids borrow helicases and polymerases from the host cell. The average number of a particular plasmid in a newborn cell is referred to as the copy number. Relaxed plasmids have high copy numbers, and stringent plasmids have low copy numbers. There exist feedback mechanisms that regulate plasmid copy number within the cell [55]. For example, at high plasmid concentrations proteins can bind to plasmids, blocking their

replication. At the time of cell division, plasmids are divided among the daughter cells. Although plasmids are not usually distributed equally among progeny cells, partitioning systems ensure that at least one copy of the plasmid segregates into each daughter cell.

2.2.5 Transformation

Transformation occurs when cells take up free DNA directly from their environment. This type of gene exchange was the first to be discovered in bacteria. Transformation is often the best way to reintroduce experimentally altered DNA into cells. DNA gets bound to specific receptors on the cell surface, and the bound DNA is broken into smaller pieces by endonucleases. Transformation is almost always single-strand uptake [53]. One of the DNA strands is degraded, while the other complementary strand is transported into the cell.

Cells that are capable of taking up DNA are said to be competent. Experiments show that competent bacteria take up DNA with relatively high efficiency [53]. Much less is known about the gene products that make cells permeable to DNA. Some types of bacteria will take up DNA from any source, while other types of bacteria will only take up DNA from their own species having specific uptake sequences. Most types of cells are not naturally transformable. However, some cells can be made competent by electroporation, brief electric shocks that submit the cells to strong electric fields, or with certain chemical treatments, such as calcium ion induction [56, 57].

2.3 Antibiotics

Antibiotics are chemicals that impede genetic processes and the growth of cells [58]. These substances can prevent DNA replication or change its structure. To stop replication, antibiotics usually halt polymerization by binding to DNA. When bound to DNA, antibiotics may also block RNA production. Structural changes can occur when antibiotics terminate DNA chains by mimicking deoxynucleotides; for example, Mitomycin-C cross-links guanine bases [59]. Antibiotics can also prevent translation of protein by inhibiting binding of ribosomes and tRNA. While all the possible forms of antibiotic interference seem to have negative connotations, the effects of antibiotics can indeed be advantageous to a cell when preventing the expression of harmful genes.

Chapter 3

Frequency-Domain Analysis

In general, a system is a group of interconnected components that generates one or more outputs by processing or transforming input signals. Linear systems possess the property of superposition: if an input consists of a scaled sum of several inputs, then the output is the scaled sum of the individual responses to each single input. A system whose characteristics do not change over time is referred to as time-invariant. Many systems encountered in nature can be modeled as linear time-invariant (LTI) systems [60], whether it be at all times or only under certain operating conditions. Since LTI systems are so common, many analytical techniques have been developed for studying their behavior. This chapter reviews frequency-domain (FD) analysis applicable to all LTI systems.

3.1 Transfer Functions

A block diagram for a system with one input and output is shown in Figure 3.1. If the input signal is described by the function $x(t)$, then the output, $y(t)$, is found as

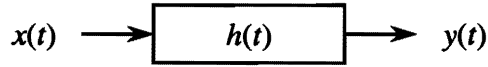


Figure 3.1. Block diagram of a system with input signal, $x(t)$, impulse response $h(t)$, and output, $y(t)$.

$$y(t) = \int_{-\infty}^{\infty} x(\tau)h(t - \tau)d\tau, \quad (3.1)$$

where $h(t)$ represents the impulse response of the system [60]. The solution of the convolution integral in Eq. 3.1 is sometimes nontrivial, and it is often more easily solved in the frequency domain. Conversion from time domain to the frequency domain occurs through the Fourier transform:

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-i2\pi ft} dt, \quad (3.2)$$

where f represents frequency (Hz) and i symbolizes $\sqrt{-1}$. In the frequency domain, Eq. 3.1 becomes [60]

$$Y(f) = X(f) \cdot H(f), \quad (3.3)$$

which means that the FD output response of a system is simply the product of the Fourier transforms of the input signal and the system's impulse response. The time-domain output is then given by

$$y(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} Y(f) e^{i2\pi ft} df, \quad (3.4)$$

which defines the inverse Fourier transform. The term $H(f)$ in Eq. 3.3 is called the transfer function. A transfer function is defined as the ratio of the FD output to the FD input, that is, $H(f) = Y(f) / X(f)$. Note that a transfer function depends on what is defined as the input and output, and thus, a single system can have many transfer functions. Standard notation for a transfer function polynomial is

$$H(f) = C \frac{\prod_{m=1}^m \left(1 + i \frac{f}{z_m}\right)}{\prod_{n=1}^n \left(1 + i \frac{f}{p_n}\right)}, \quad (3.5)$$

where C is a constant and z_m and p_n are the zeros and poles of the system, respectively.

To help illustrate the application of FD analysis, consider the common-emitter inverting amplifier shown in Figure 3.2. This system receives a small input voltage signal, $v_i(t)$, and generates an amplified output voltage signal, $v_o(t)$. Derivation of the transfer function for this circuit begins by drawing a small-signal model and then applying the Fourier transform to each of the components. Figure 3.3 shows a high-frequency model for the amplifier circuit, after Fourier transform, that substitutes base-collector and base-emitter capacitance with Miller capacitance, C_M , and ignores any base resistance in the bipolar junction transistor [61]. In Figure 3.3, r_{π} is the small-signal

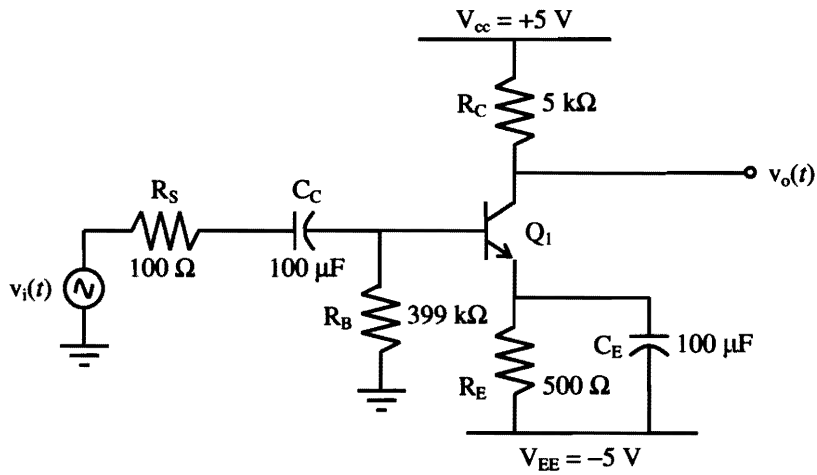


Figure 3.2. Common-emitter inverting amplifier.

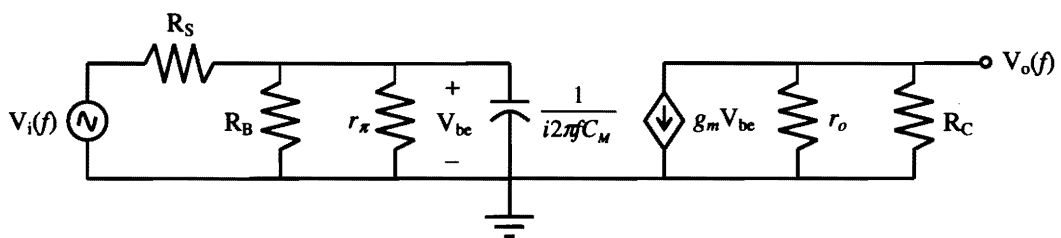


Figure 3.3. Frequency-domain model of common-emitter amplifier at high frequency.

base-emitter resistance and r_o is the output resistance of the transistor [61]. At high frequency, capacitors C_C and C_E are modeled as a short circuit (i.e., no impedance). Nodal analysis to find $H(f) = V_o(f) / V_i(f)$ yields a single-pole transfer function:

$$H(f) = \frac{A_o}{\left(1 + i \frac{f}{f_c}\right)}, \quad (3.6)$$

where the passband voltage gain, A_o , is given by

$$A_o = -g_m (R_C \parallel r_o) \left(\frac{R_B \parallel r_\pi}{R_S + R_B \parallel r_\pi} \right), \quad (3.7)$$

$f_c = 1/[2\pi C_M(R_S \parallel R_B \parallel r_\pi)]$, and g_m is the transconductance of the transistor [61].

3.1.1 Bode Plots

A Bode plot is a graphical tool for visualizing the frequency response, $H(f)$, of a system [62]. A Bode diagram consists of two plots. The first plot displays the magnitude of $H(f)$ in decibels (dB) versus frequency. The magnitude of $H(f)$, $|H(f)|$, from Eq. 3.6 can be expressed in dB as

$$|H(f)|_{dB} = 20 \cdot \log(|A_o|) - 20 \cdot \log\left(\frac{\sqrt{f^2 + f_c^2}}{f_c}\right). \quad (3.8)$$

For $f \ll f_c$, the magnitude of $H(f)$ is practically constant as the first term on the right-hand side of Eq. 3.8 dominates the expression. When $f = f_c$, the magnitude of $H(f)$ drops 3 dB below its maximum of $20 \cdot \log(|A_o|)$. For $f \gg f_c$, the magnitude of $H(f)$ then decreases 20 dB per decade increase in frequency, as shown by Figure 3.4(a). As a rule, the slope of a

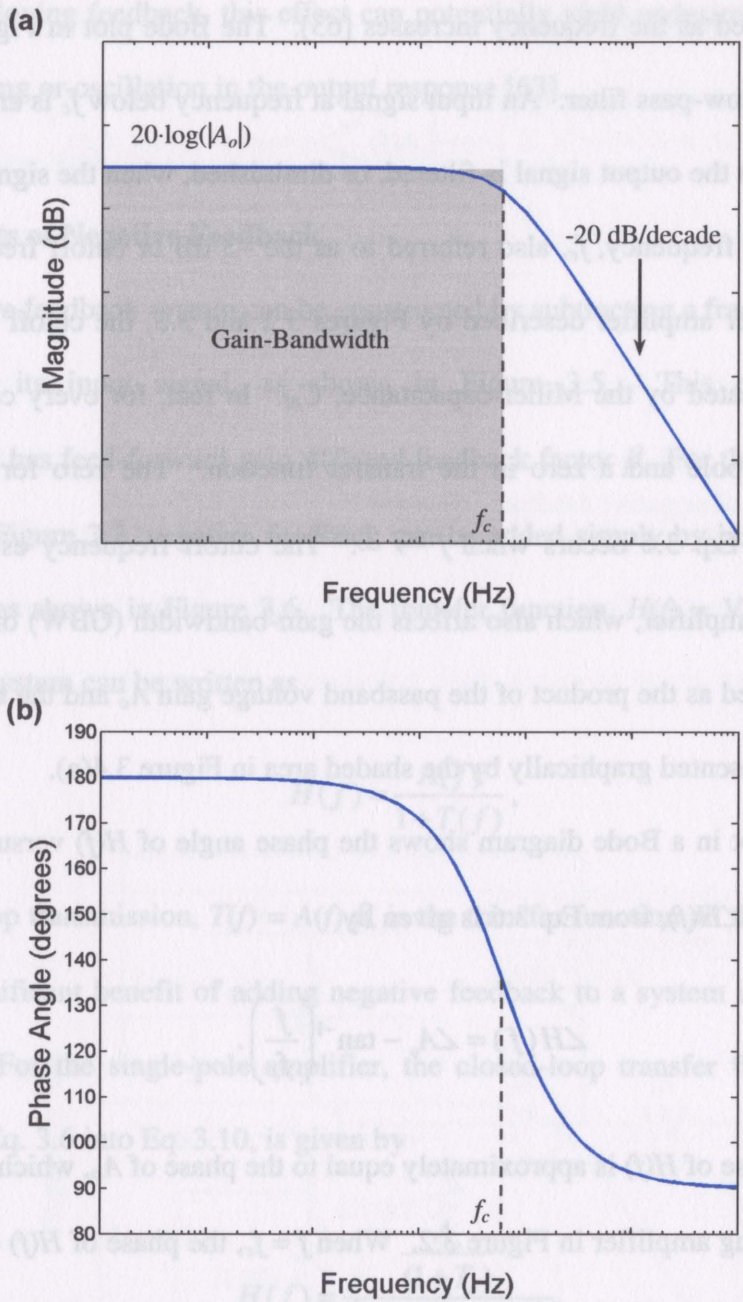


Figure 3.4. High-frequency Bode plot of the transfer function for the common-emitter inverting amplifier with cutoff frequency f_c . (a) Magnitude, and (b) phase angle of the transfer function.

magnitude Bode plot decreases (increases) an additional 20 dB/decade for each pole (zero) that is crossed as the frequency increases [63]. The Bode plot in Figure 3.4(a) is characteristic of a low-pass filter. An input signal at frequency below f_c is amplified by a gain of A_o , but then the output signal is filtered, or diminished, when the signal frequency exceeds the corner frequency, f_c , also referred to as the -3 dB or cutoff frequency. For the common-emitter amplifier described by Figures 3.2 and 3.3, the cutoff frequency is set by the pole created by the Miller capacitance, C_M . In fact, for every capacitor in a circuit, there is a pole and a zero in the transfer function. The zero for the transfer function given by Eq. 3.6 occurs when $f \rightarrow \infty$. The cutoff frequency establishes the bandwidth of the amplifier, which also affects the gain-bandwidth (GBW) of the system. The GBW is defined as the product of the passband voltage gain A_o and the bandwidth of the amplifier, represented graphically by the shaded area in Figure 3.4(a).

The second plot in a Bode diagram shows the phase angle of $H(f)$ versus frequency. The phase of $H(f)$, $\angle H(f)$, from Eq. 3.6 is given by

$$\angle H(f) = \angle A_o - \tan^{-1} \left(\frac{f}{f_c} \right). \quad (3.9)$$

For $f \ll f_c$, the phase of $H(f)$ is approximately equal to the phase of A_o , which is 180° (Eq. 3.7) for the inverting amplifier in Figure 3.2. When $f = f_c$, the phase of $H(f)$ decreases by 45 degrees, as shown in Figure 3.4(b). Then, for $f \gg f_c$, the phase is decreased by 90 degrees. In general, the phase angle of a Bode plot decreases (increases) an additional 90 degrees for each left-half plane pole (zero) that is crossed as the frequency increases [63]. Back in the time domain, these changes in phase angle correspond to time delays in the

output signal: as phase angle decreases, the output signal lags further behind in time. For systems employing feedback, this effect can potentially yield undesirable behavior, such as overshooting or oscillation in the output response [63].

3.1.2 Effects of Negative Feedback

A negative feedback system can be constructed by subtracting a fraction of a system's output from its input signal, as shown in Figure 3.5. This classical feedback configuration has feed-forward gain $A(f)$ and feedback factor β . For the common-emitter amplifier in Figure 3.2, negative feedback can be added simply by inserting a feedback resistor, R_F , as shown in Figure 3.6. The transfer function, $H(f) = V_o(f) / V_i(f)$, of this closed-loop system can be written as

$$H(f) = \frac{A(f)}{1+T(f)}, \quad (3.10)$$

where the loop transmission, $T(f) = A(f) \cdot \beta$, is the transfer function all the way around the loop. A significant benefit of adding negative feedback to a system is the extension of bandwidth. For the single-pole amplifier, the closed-loop transfer function, found by substituting Eq. 3.6 into Eq. 3.10, is given by

$$H(f) = \frac{\frac{A_o}{(1+T_o)}}{\left(1+i\frac{f}{(1+T_o)f_c}\right)}, \quad (3.11)$$

with $T_o = A_o \cdot \beta$. Consequently, the magnitude of the passband gain is reduced by a factor of $1+|T_o|$, and the effective cutoff frequency (i.e., bandwidth) is increased by $1+|T_o|$.

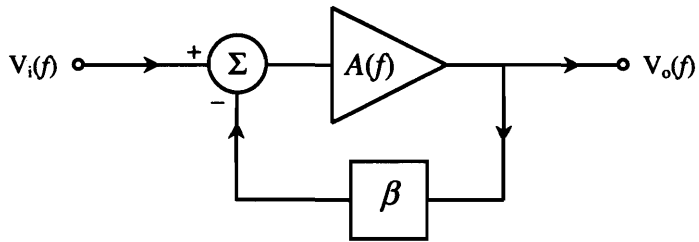


Figure 3.5. Classical negative feedback system.

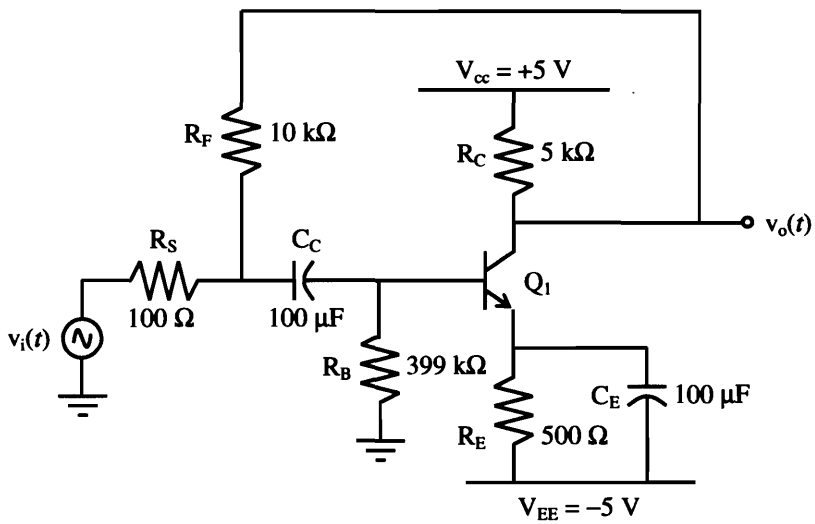


Figure 3.6. Common-emitter amplifier with negative feedback.

These two combined effects, illustrated by the Bode plot in Figure 3.7, are known as the gain-bandwidth tradeoff.

3.2 Random Signals

Signals that cannot be precisely described by an equation or predicted at any given time are called random signals. Random signals are found in all real systems. For example, shot noise is present in the current of semiconductor diodes, due to stochastic injection of carriers (electrons and holes) across the *pn*-junction [64]. Although the exact value of these fluctuations in a diode's current is nondeterministic, some properties of the noise can be described exactly. One of the best ways to analyze random signals as they propagate through systems is by examining their autocorrelation and energy spectral density.

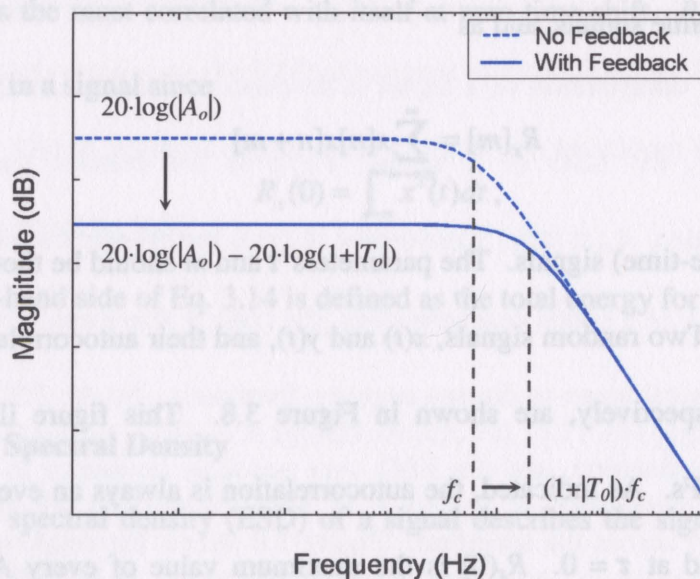


Figure 3.7. Gain-bandwidth tradeoff with addition of negative feedback.

3.2.1 Autocorrelation

Correlation between two different signals is a measurement of how much they are related to each other. The two signals are said to be positively correlated when both of the signals tend to move in the same direction over a long period of time. If one signal moves upward over a long duration of time while the other moves downward, then the two signals are negatively correlated. The two signals are said to be uncorrelated if, over a long period of time, they move together roughly half of the time and move in opposite directions the remainder of the time.

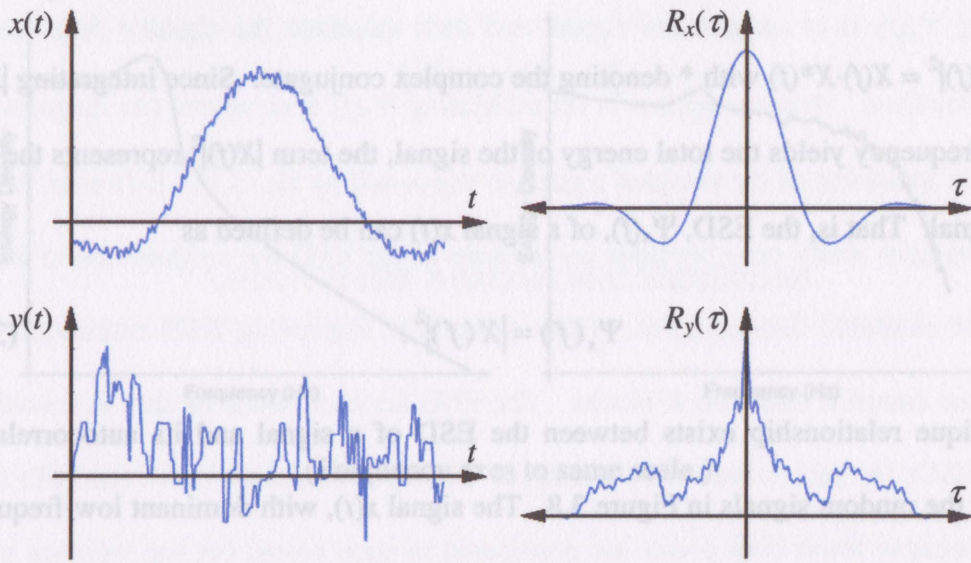
Autocorrelation describes how a signal is correlated with *itself* over time and is one of the best ways to describe a random signal. The autocorrelation function (ACF), $R_x(\tau)$, for a signal $x(t)$ can be defined as [65]

$$R_x(\tau) = \int_{-\infty}^{\infty} x(t)x(t+\tau)dt \quad (3.12)$$

for real continuous-time signals, and as

$$R_x[m] = \sum_{n=-\infty}^{\infty} x[n]x[n+m] \quad (3.13)$$

for sampled (discrete-time) signals. The parameters τ and m should be thought of as time shifts or time lags. Two random signals, $x(t)$ and $y(t)$, and their autocorrelation functions, $R_x(\tau)$ and $R_y(\tau)$, respectively, are shown in Figure 3.8. This figure illustrates some properties of all ACFs. As indicated, the autocorrelation is always an even function and is always maximized at $\tau = 0$. $R_x(0)$ is the maximum value of every ACF because a



(Time axes to same scale.)

Figure 3.8. Sampling of two random signals and their autocorrelation functions.

signal is always the most correlated with itself at zero time shift. $R_x(0)$ is also equal to the total energy in a signal since

$$R_x(0) = \int_{-\infty}^{\infty} x^2(t) dt, \quad (3.14)$$

where the right-hand side of Eq. 3.14 is defined as the total energy for a signal $x(t)$.

3.2.2 Energy Spectral Density

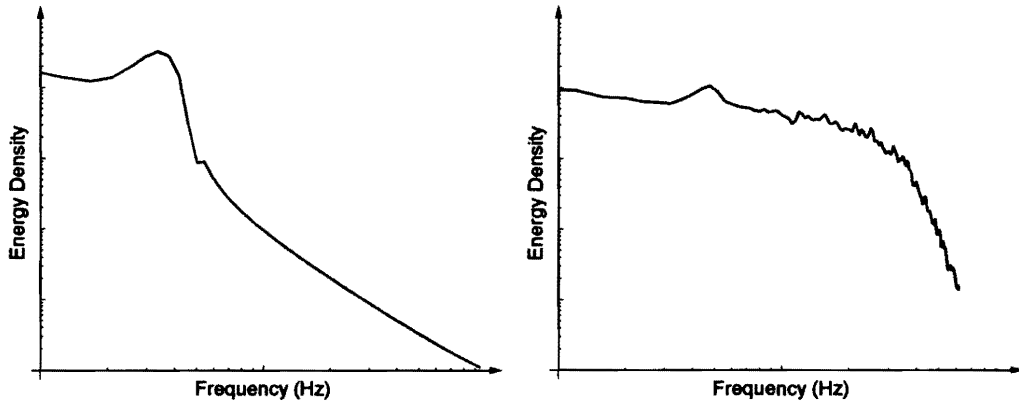
The energy spectral density (ESD) of a signal describes the signal's distribution of energy as a function of frequency. From Parseval's theorem, the relationship between a signal's total energy, E_x , and its Fourier transform is given by

$$E_x = \int_{-\infty}^{\infty} |X(f)|^2 df, \quad (3.15)$$

where $|X(f)|^2 = X(f) \cdot X^*(f)$ with $*$ denoting the complex conjugate. Since integrating $|X(f)|^2$ over all frequency yields the total energy of the signal, the term $|X(f)|^2$ represents the ESD of the signal. That is, the ESD, $\Psi_x(f)$, of a signal $x(t)$ can be defined as

$$\Psi_x(f) = |X(f)|^2. \quad (3.16)$$

A unique relationship exists between the ESD of a signal and its autocorrelation. Consider the random signals in Figure 3.8. The signal $x(t)$, with dominant low-frequency components, varies slowly over time. The corresponding autocorrelation function, $R_x(\tau)$, has a relatively broad spread about $\tau = 0$, as the signal is highly correlated with itself over long durations of time. In comparison, the rapidly changing signal $y(t)$ has stronger high-frequency components, and the spread of its ACF about $\tau = 0$ is noticeably narrower because $y(t)$ becomes less correlated with itself after fewer time lags. Hence, information about the frequency composition of a signal is revealed in the features of its ACF. In mathematical terms, the ESD of a signal, $\Psi_x(f)$, is the Fourier transform of its autocorrelation function, $R_x(\tau)$ [65]. Conversely, $R_x(\tau)$ is the inverse Fourier transform of $\Psi_x(f)$. Since the ACF of a signal is always even, the ESD is also even. For real systems, it doesn't make physical sense to discuss negative frequency. So the ESD over positive frequency is usually doubled, and the ESD over negative frequency is set to zero to preserve total energy across the spectrum. Figure 3.9 shows the single-sided ESDs for the random signals in Figure 3.8. In agreement with the signal descriptions given above,



(Frequency axes to same scale.)

Figure 3.9. Energy spectral densities of the random signals in Figure 3.8.

most of the energy in $x(t)$ is found at lower frequencies, while the energy in $y(t)$ is distributed across higher frequencies.

The effect a system has on the ESD of a signal is determined by the system's transfer function, $H(f)$. Given an input signal with $\Psi_x(f)$, it can be shown that the ESD of the output, $\Psi_y(f)$, is given by [65]

$$\Psi_y(f) = |H(f)|^2 \Psi_x(f). \quad (3.17)$$

Besides determining the ESD of a system output, Eq. 3.17 has significant application. For example, if the transfer function of a system is completely unknown, then by measuring the ESD of the input and output signals, one can deduce $H(f)$. This is one approach that can be used for system identification.

There are several different ways to measure the ESD of a signal [66]. One method of estimating $\Psi_x(f)$ is to sample the signal, and then calculate the signal's ACF and its Fourier transform. Another approach for estimating $\Psi_x(f)$ is to square the magnitude of the Fourier transform of the sampled signal, as suggested by Eq. 3.16. However, each of these techniques really only provides an estimate of the ESD for a signal due to a finite number of available data samples. One method for improving ESD estimation is to window the sampled data into K blocks. The ESD for each block of data is found, and then the ESDs for each window are averaged. The spectral resolution of the ESD is not as great because fewer data points are considered in each block, but the variance of the ESD estimate is decreased.

Chapter 4

Gene Circuit Analysis and Simulation

In this chapter, analysis and simulation are mutually applied towards the development of models for two fundamental building blocks in genetic regulatory networks: the unregulated and autoregulated gene circuit. The analyses presented below focus primarily on the behavior of gene circuit noise. There are several motivations behind this. For one, there is considerable interest in understanding how genetic systems are able to function properly, even in environments with discrete numbers of molecules, given all of their noise generating components. Secondly, frequency-domain (FD) noise analysis not only gives structural detail about a system but also reveals information through application of transfer functions about how the circuit processes its inherent noise [14, 15]. Finally, current technologies now provide means for measuring noise in gene expression [31-33], which facilitates the coupling of analysis and simulation with experimentation in order to assess the accuracy of developed models and make model refinements as needed.

4.1 Modeling and Noise Analysis

Modeling a gene circuit begins by applying biological knowledge to list all of the biochemical reactions considered integral to the circuit. To assist FD analysis, the models are converted into electrical circuits to make them more familiar to electrical engineers. Next, all of the identified noise sources are added to the circuit, and then FD transfer functions are derived for each of the noise sources. Finally, the contributions from each noise source (i.e., biochemical process) are analyzed along with the total noise spectrum of the gene circuit. Although the FD approach shares some of the same limitations as Langevin analysis (e.g., linearization), the FD techniques applied below have been shown to accurately retain the spectral features of noise and remain valid in many regions of gene circuit operation, even at low molecular populations [15, 16]. Furthermore, FD analysis yields equations with simple forms that uncover relationships between circuit parameters and noise behavior [14-16]. The equations developed below are generalized and applicable to almost any unregulated or autoregulated gene circuit.

4.1.1 Unregulated Gene Circuit

Previous gene circuit models usually include only transcription, translation, and decay of mRNA and protein, and they describe all of these processes as simple one-step reactions (for example, see Figure 1.1 [14]). A more complete model would describe transcription and translation as two-phase processes that include initiation and elongation [67], creating a minimum time delay before mRNA (τ_r) and protein (τ_p) molecules are functionally available. It is expected that the total delay between transcription initiation

and the formation of functional protein ($\tau_r + \tau_p$) be on the order of a few minutes [68]. Oversimplified models also ignore protein multimerization processes found in many well-characterized biological systems [69]. For example, both CI and Cro monomers form homodimers before binding to the P_{RM} and P_R promoters in the λ -phage lysis-lysogeny decision circuit [70]. These reversible reactions are important because they are known to have significant and population-dependent effects on the spectral distribution of stochastic fluctuations in gene circuits [15].

Figure 4.1 shows a schematic representation of an improved model for an unregulated (open-loop) gene circuit along with the respective rate constants assigned to each reaction. While integrating first-order reactions from previous analyses, the model in Figure 4.1 also includes the missing features described above. Here, RNAP binds to DNA at rate K_{in} to form complex C, and transcript initiation follows at rate K_c . After clearing the promoter region, the DNA is available again for RNAP-binding. Meanwhile, polymerase in state M_0 continues to transcribe DNA (rates $K_{m0}, K_{m1}, \dots, K_{mM}$) through M sequential states (M_1, M_2, \dots, M_M). Although mRNA synthesis continues after these M steps, at this point the ribosomal binding site is available for the initiation of translation. Synthesized mRNA decays (*) at rate γ_r . Ribosomes bind with mRNA molecules at rate K_p , and translation initiation proceeds at rate K_{mC} . After clearing the translation initiation region, the mRNA is available again for ribosome binding. Ribosomes in state T_0 continue translating protein (rates $K_{t0}, K_{t1}, \dots, K_{tM}$) through M states (T_1, T_2, \dots, T_M) to create protein monomers, P. Protein monomers decay (*) at rate γ_p and also dimerize at rate K_f to form homodimers, Di. To incorporate dissolution of dimer molecules, the

Table 4.1. ODEs and steady-state equations for the unregulated gene circuit

Differential Equation	Steady-State Equation
$\frac{d[C]}{dt} = K_{in}[DNA] - K_c[C]$	$\langle C \rangle = \frac{K_{in}}{K_c} \langle DNA \rangle$
$\frac{d[M_0]}{dt} = K_c[C] - K_{m0}[M_0]$	$\langle M_0 \rangle = \frac{K_c}{K_{m0}} \langle C \rangle$
$\frac{d[M_M]}{dt} = K_{m(M-1)}[M_{M-1}] - K_{mM}[M_M]$	$\langle M_M \rangle = \frac{K_{m(M-1)}}{K_{mM}} \langle M_{M-1} \rangle$
$\frac{d[mRNA]}{dt} = K_{mM}[M_M] - \gamma_r[mRNA] - K_p[mRNA] + K_{mC}[mC]$	$\langle mRNA \rangle = \frac{K_{mM}}{\gamma_r} \langle M_M \rangle$
$\frac{d[mC]}{dt} = K_p[mRNA] - K_{mC}[mC]$	$\langle mC \rangle = \frac{K_p}{K_{mC}} \langle mRNA \rangle$
$\frac{d[T_0]}{dt} = K_{mC}[mC] - K_{i0}[T_0]$	$\langle T_0 \rangle = \frac{K_{mC}}{K_{i0}} \langle mC \rangle$
$\frac{d[T_M]}{dt} = K_{i(M-1)}[T_{M-1}] - K_{iM}[T_M]$	$\langle T_M \rangle = \frac{K_{i(M-1)}}{K_{iM}} \langle T_{M-1} \rangle$
$\frac{d[P]}{dt} = K_{iM}[T_M] - \gamma_p[P] - K_f[P]^2 + K_r[Di]$	$\langle P \rangle = \frac{K_{iM}}{\gamma_p} \langle T_M \rangle$
$\frac{d[Di]}{dt} = K_f[P]^2 - K_r[Di]$	$\langle Di \rangle = \frac{K_f}{K_r} \langle P \rangle^2$

RNAP is bound to the operator or one when the DNA site is freed. The average (steady-state) population, denoted by $\langle \rangle$, of the unbound operator is

$$\langle O \rangle = \frac{T_u}{T_u + T_b}, \quad (4.1)$$

where $T_u = 1/(K_{in}\langle O \rangle)$ is the average period of time the operator remains unbound, and $T_b = 1/K_c$ is the average period of time the operator is bound. The steady-state populations for the remaining chemical species in Figure 4.1 are found by setting the ODEs equal to zero and simultaneously solving the system of equations. Expressions for these steady-state populations are given in Table 4.1.

Using substitution, the ODEs in Table 4.1 can be reduced to four equations that include just four essential species: DNA, mRNA, monomer protein (P), and dimer protein (Di). The model for the unregulated gene is converted into an electrical circuit, shown in Figure 4.2, where each node, circled and numbered 1-4, corresponds to one of the four primary chemical species. Each capacitor is assigned a value of one so that the voltage at its node is the population or concentration, denoted by bracketed terms, of the respective species. The delay τ_r is included with the transcription current source, K_{in} . Likewise, the delay τ_p is built-in with the mRNA-dependent current source for translation. Finally, S_2 , S_3 , and S_{3b} are noise sources associated with biochemical processes and are characterized below.

The FD signal processing functionality of each biochemical process is found through Fourier transforms to yield gain transfer functions, $H(f) = \partial o(f)/\partial i(f)$, where o and i are output and input signals (e.g., molecular populations) respectively, and f is frequency in

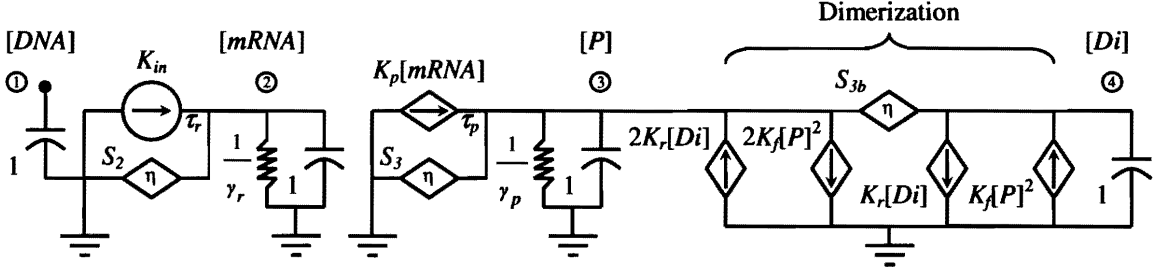


Figure 4.2. Electrical circuit equivalent for the unregulated gene circuit. The transcription delay, τ_r , and translation delay, τ_p , are combined with their related current sources.

Hz [15]. The noise ESD $S_{j,k}(f)$ at any node j due to a noise source at node k with ESD $S_k(f)$ is given by

$$S_{j,k}(f) = |H_{j,k}(f)|^2 \cdot S_k(f), \quad (4.2)$$

where $|H_{j,k}(f)|^2$ is the power transfer function from node k to node j . If all noise sources are statistically independent, then the total noise ESD at node j , $S_j(f)$, is

$$S_j(f) = \sum_{k=1}^N S_{j,k}(f), \quad (4.3)$$

with N equal to the number of noise sources in the circuit.

Noise sources in gene circuits are located at points of molecular transitions including synthesis, multimerization, decay or dissolution [15]. The gene circuit in Figure 4.1 has

at least six individual noise sources that account for the stochastic nature of mRNA production, mRNA decay, monomer protein synthesis, protein decay, dimerization, and dimer dissolution. At steady state where opposing processes (e.g., synthesis and decay) are equal, these noise components are easily condensed into three sources (Figure 4.2). The first noise source, entering the circuit at node 2, describes noise due to the steady-state production and decay of mRNA, which can be accurately modeled as wideband shot noise [14, 16]. The single-sided ESD (positive frequency only) for this source is given by

$$S_2(f) = 4K_m \langle O \rangle. \quad (4.4)$$

The second noise source for the network enters at node 3 and has a single-sided ESD given by

$$S_3(f) = 4K_p \langle mRNA \rangle, \quad (4.5)$$

which is a white noise source that accounts for the synthesis and decay of monomer protein [14]. The third noise source is due to stochastic fluctuations in dimer formation and dissolution, which at steady-state can be modeled as one wideband noise source located *between* nodes 3 and 4 with a single-sided ESD of

$$S_{3b}(f) = 4K_r \langle Di \rangle. \quad (4.6)$$

The frequency-dependent transfer functions needed to compute Eq. 4.2 are found by applying Fourier transforms to the deterministic ODEs for the chemical reactions in the gene circuit. The results of the transformation are shown in Figure 4.3. The function

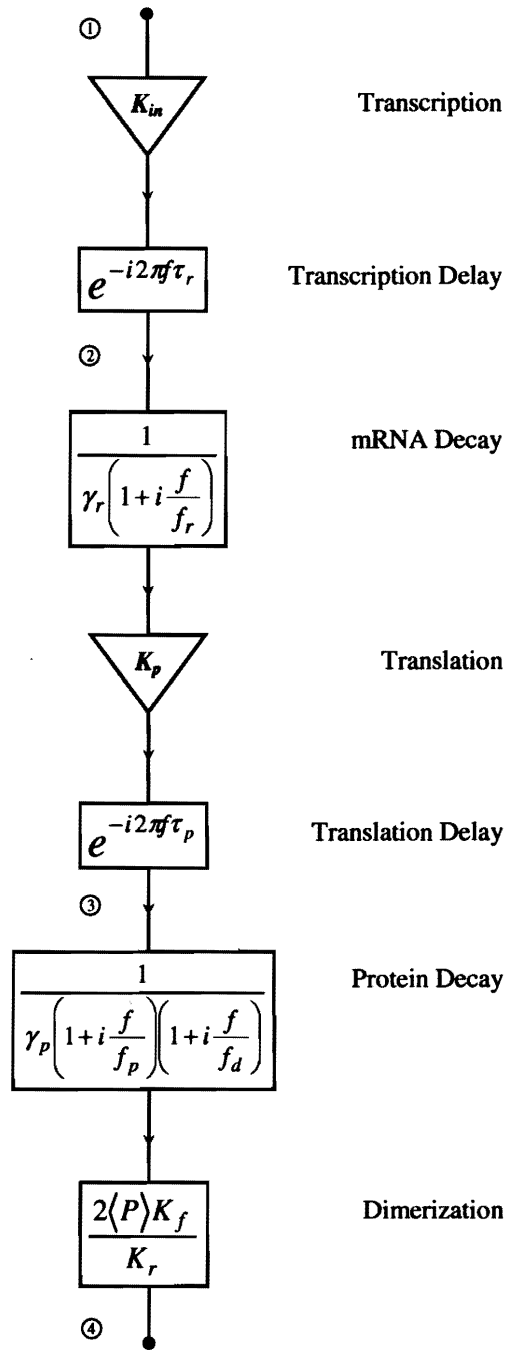


Figure 4.3. Frequency-dependent functions for the biochemical processes in the unregulated gene circuit.

blocks containing exponential terms account for phase shift [65] caused by the transcription time delay τ_r and translation (plus protein folding) time delay τ_p . The transfer functions given for protein decay and dimerization have been reported previously [15]. In the frequency domain, the gain transfer function from node j to node k of the circuit is simply the product of all the FD function blocks between the two nodes. As a result, the gain transfer function from mRNA synthesis to output dimer is given by

$$H_{4,2}(f) = \frac{2\langle P \rangle K_f K_p}{K_r \gamma_p \gamma_r} \left(\frac{1}{1 + i \frac{f}{f_r}} \right) \left(\frac{1}{1 + i \frac{f}{f_p}} \right) \left(\frac{1}{1 + i \frac{f}{f_d}} \right) \exp\{-i2\pi f(\tau_r + \tau_p)\}, \quad (4.7)$$

where the pole frequency $f_r = \gamma_r/2\pi$ is due to the decay of mRNA [14], and the coupled poles $f_p = \gamma_p/(2\pi[1+(2\langle P \rangle K_f)/K_r])$ and $f_d = K_r \cdot [1+(2\langle P \rangle K_f)/K_r]/2\pi$ are due to the dimerization and decay of monomer protein and move in frequency space as a function of protein population [15]. The gain from monomer to dimer can be written as

$$H_{4,3}(f) = \frac{2\langle P \rangle K_f}{K_r \gamma_p} \left(\frac{1}{1 + i \frac{f}{f_p}} \right) \left(\frac{1}{1 + i \frac{f}{f_d}} \right) \cdot \exp\{-i2\pi f \tau_p\}, \quad (4.8)$$

where the exponential term now accounts only for any translation and protein-folding time delay τ_p . Using transfer functions derived by Cox, *et al.* [15], the gain from the dimerization noise source S_{3b} , located between nodes 3 and 4, to the dimer output is given by

$$H_{4,3b}(f) = -\frac{1}{K_r} \frac{\left(1 + i \frac{2\pi f}{\gamma_p}\right)}{\left(1 + i \frac{f}{f_p}\right) \left(1 + i \frac{f}{f_d}\right)}, \quad (4.9)$$

where the pole frequencies f_p and f_d are given above for Eq. 4.7.

With all of the transfer functions derived, the ESD of the *total* noise generated in the output dimer population, $S_4(f)$ (Eq. 4.3), and its constituents, $S_{j,k}(f)$ (Eq. 4.2), can be calculated. Figure 4.4 shows calculated ESDs of the dimer noise and its components for the unregulated gene circuit using the parameter values listed in Table 4.2. Matlab (The MathWorks, Natick, MA) source code used to compute the noise ESDs can be found in Appendix A. The parameter values in Table 4.2 were chosen to fit within realistic physiological ranges and illustrate key features of the analysis. The number of elongation steps ($M = 10$) was selected arbitrarily and the values of K_{mM} and K_{tM} were set to 0.1 s^{-1} to produce an average delay time of 100 s for both $\tau_r (M/K_{mM})$ and $\tau_p (M/K_{tM})$. The decay rates for mRNA (γ_r) and protein (γ_p) correspond to a half-life ($\lambda = \ln(2)/\gamma$) of 2 and 60 min, respectively [71]. The burst rate ($b = K_p/\gamma_r$) was chosen to be ~ 20 proteins per transcript, which lies between the reported values of $b = 5$ for the *lacA* gene and $b = 40$ for *lacZ* [72]. The remaining parameters were selected to yield $\langle P \rangle \approx 100$ and $\langle Di \rangle \approx 10$ molecules. The results in Figure 4.4 reveal some important features about the composition of noise in the output dimer population. At lower frequencies, $S_4(f)$ is determined primarily by transcriptional noise ($S_{4,2}$) caused by fluctuations in the synthesis and decay of mRNA. Conversely, at higher frequencies the dimer noise is controlled

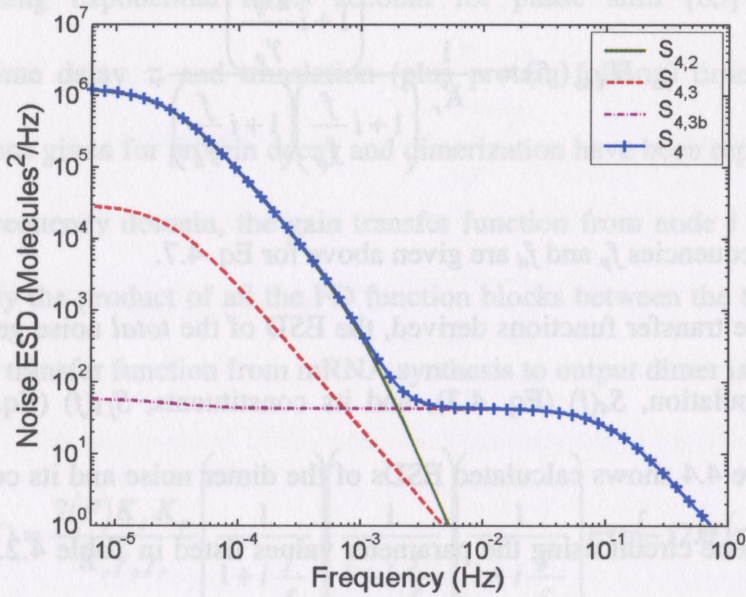


Figure 4.4. Total noise ESD in the steady-state dimer population, S_4 , and calculated contributions to S_4 by each of the noise sources, $S_{j,k}$, for the unregulated gene circuit.

Table 4.2. Parameters used to model the unregulated gene circuit

Parameter	Value	Reaction
K_{in}	0.001 s^{-1}	DNA-RNAP
K_c	0.1 s^{-1}	Transcription
$K_{mM} (M = 0-9)$	0.1 s^{-1}	Elongation
τ_r	100 s	Transcription delay
γ_r	0.0058 s^{-1}	mRNA decay
K_p	0.12 s^{-1}	MRNA-Ribosome
K_{mC}	0.1 s^{-1}	Translation
$K_{tM} (M = 0-9)$	0.1 s^{-1}	Elongation
τ_p	100 s	Translation delay
γ_p	0.0002 s^{-1}	Protein decay
K_f	0.0005 s^{-1}	Dimer formation
K_r	0.6 s^{-1}	Dimer dissolution

almost entirely by the dimerization process ($S_{4,3b}$). Interestingly, the noise contribution from the synthesis and decay of monomer protein ($S_{4,3}$) has a negligible effect on dimer noise throughout the entire frequency spectrum.

4.1.2 Autoregulated Gene Circuit

A great deal of stochastic analysis and experimentation has focused on negatively autoregulated gene circuits [14, 26, 31, 32, 73], a common control motif that regulates more than 40% of the known transcription factors in *Escherichia coli* [74]. For the most part, previous analyses have relied on a Hill kinetics model to describe protein multimerization, cooperative binding, and gene expression control by protein-DNA binding at an operator site. Unfortunately, important noise and dynamical features of gene expression control are neglected in this simplified representation [16].

Figure 4.5 shows a schematic representation of a single autoregulated (closed-loop) gene circuit, along with assigned reaction rate constants, that explicitly includes gene regulation by dimer-DNA binding at the operator site. The features of this model are exactly as described above for the unregulated gene circuit, except that protein dimers negatively regulate gene expression by binding and unbinding with the operator (DNA) at rates K_b and K_u , respectively. Since dimers are the regulatory elements in this gene circuit, the following noise analysis is applied to the output dimer population.

After simplification of the ODEs for the biochemical reactions in Figure 4.5, the average steady-state mRNA ($\langle mRNA \rangle$), protein ($\langle P \rangle$), and dimer ($\langle Di \rangle$) populations are

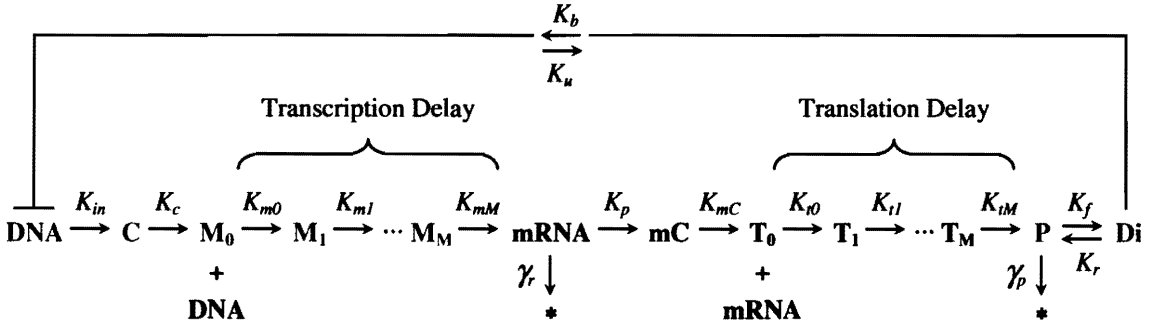


Figure 4.5. Autoregulated gene circuit model with reversible operator-dimer binding.

$$\langle mRNA \rangle = \frac{K_{in}}{\gamma_r} \langle O \rangle \quad (4.10)$$

$$\langle P \rangle = \frac{K_p}{\gamma_p} \langle mRNA \rangle \quad (4.11)$$

$$\langle Di \rangle = \frac{K_f}{K_r} \langle P \rangle^2 \quad (4.12)$$

where the basal gene expression rate (leakiness) is assumed to be negligible. Assuming a single copy of the gene, the unbound operator population, O , is either one or zero, and the average population is

$$\langle O \rangle = \frac{T_u}{T_u + T_b}, \quad (4.13)$$

where $T_u = 1/(K_b\langle Di \rangle)$ is the average period of time the operator remains unbound, and $T_b = 1/K_u$ is the average period of time the operator is bound.

For FD analysis, the model is converted into the electrical circuit shown in Figure 4.6. The three noise sources (S_2 , S_3 , and S_{3b}) account for the stochasticity of mRNA production, mRNA decay, monomer protein synthesis, protein decay, dimerization, dimer dissolution, and dimer-DNA binding and unbinding. The first noise source, S_2 , combines the noise components due to dimer-DNA binding, mRNA production, and mRNA decay. The single-sided ESD for this source is given by

$$S_2(f) = 4 \frac{K_{in}^2 (\langle O \rangle - \langle O \rangle^2)}{K_u + K_b \langle Di \rangle} \left(\frac{1}{1 + \left(\frac{f}{f_b} \right)^2} \right) + 4K_{in} \langle O \rangle, \quad (4.14)$$

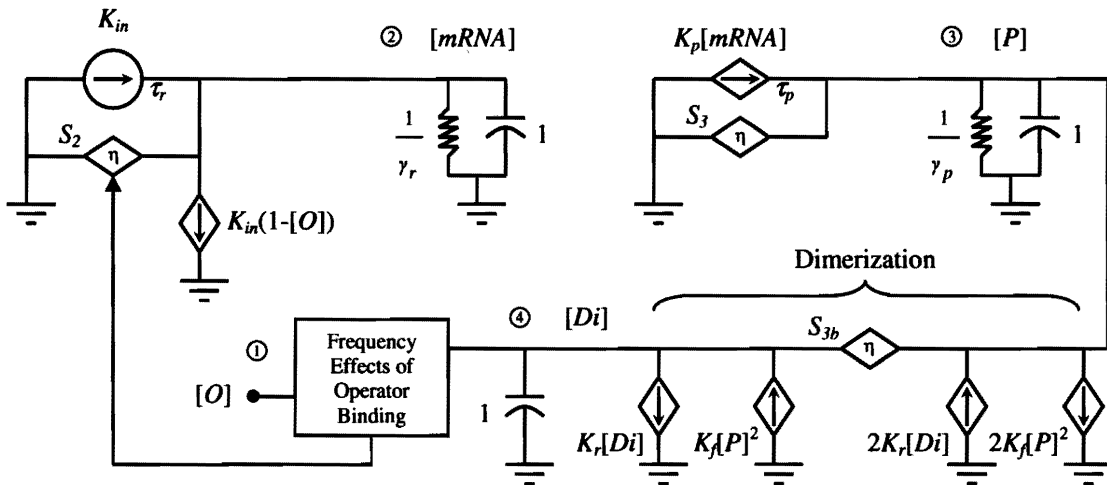


Figure 4.6. Electrical circuit equivalent for the autoregulated gene circuit.

where the pole frequency $f_b = (K_u + K_b/Di) + K_b/2\pi$ [16]. The first term on the right-hand side of Eq. 4.14 is the noise ESD for mRNA synthesis due to random fluctuations in dimer-DNA binding (operator noise) [16]; the second term describes noise due to the steady-state production and decay of mRNA, which is modeled as white noise [14, 16]. The other two noise sources S_3 and S_{3b} , which account for synthesis and decay of monomer and dimer protein, are the same as in the unregulated gene circuit and are characterized by Eqs. 4.5 and 4.6, respectively.

The loop transmission technique is used to derive the transfer functions $H_{j,k}(f)$ of the autoregulated gene circuit. The frequency-dependent functions of the biochemical processes, shown in Figure 4.7, are again found by Fourier transforms. For negative feedback systems, the closed-loop transfer functions can be written in the form

$$H_{j,k}(f) = \frac{A_{j,k}(f)}{1+T(f)}, \quad (4.15)$$

where $A_{j,k}(f)$ is the forward gain transfer function from node k to node j and $T(f)$ is the loop transmission. The loop transmission, T , is the transfer function around the entire loop and describes how the system responds to dampen perturbations introduced at any node in the circuit [14]. The loop transmission can be calculated as $T(f) = A_{j,k}(f) \cdot \beta_{k,j}(f)$, where $\beta_{k,j}(f)$ is the feedback transfer gain from node j back to node k . Hence, the loop transmission is a defining feature of the circuit architecture and at a given steady-state condition remains the same regardless of the selected input and output nodes.

Again, the gain transfer function from node j to node k of the circuit is simply the product of all the FD function blocks between the two nodes. Thus, the feed-forward

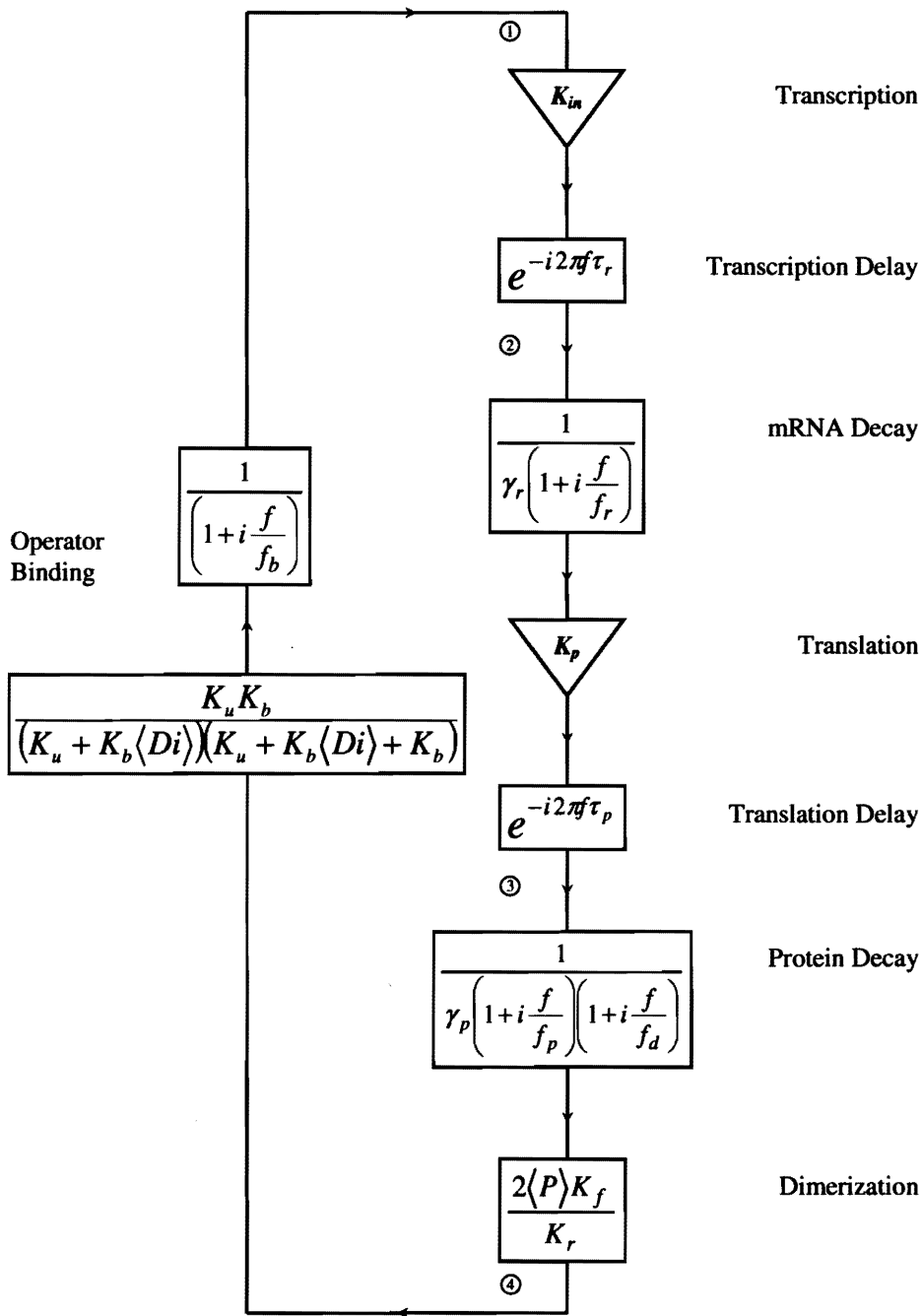


Figure 4.7. Frequency-dependent functions for the biochemical processes in the autoregulated gene circuit.

gain, $A_{4,2}$, from S_2 to the dimer is given by Eq. 4.7. Likewise, Eq. 4.8 gives the feed-forward gain, $A_{4,3}$, from S_3 to the dimer; and Eq. 4.9 gives the feed-forward gain, $A_{4,3b}$, from S_{3b} to the dimer. Finally, the feedback transfer function from dimer back to mRNA synthesis is given by

$$\beta_{2,4}(f) = K_{in} \left(\frac{K_u K_b}{(K_u + K_b \langle Di \rangle)(K_u + K_b \langle Di \rangle + K_b)} \right) \left(\frac{1}{1 + i \frac{f}{f_b}} \right), \quad (4.16)$$

where the pole frequency $f_b = (K_u + K_b \langle Di \rangle + K_b)/2\pi$ [16]. The loop transmission for the gene circuit is found as the product of $A_{4,2}(f)$ and $\beta_{2,4}(f)$, given as

$$T(f) = \frac{2 \langle P \rangle K_f K_p K_{in}}{K_r \gamma_p \gamma_r} \left(\frac{K_u K_b}{(K_u + K_b \langle Di \rangle)(K_u + K_b \langle Di \rangle + K_b)} \right) \times \left(\frac{1}{1 + i \frac{f}{f_r}} \right) \left(\frac{1}{1 + i \frac{f}{f_p}} \right) \left(\frac{1}{1 + i \frac{f}{f_d}} \right) \left(\frac{1}{1 + i \frac{f}{f_b}} \right) \exp\{-i2\pi f(\tau_r + \tau_p)\}, \quad (4.17)$$

and is used to calculate the closed-loop gains (Eq. 4.15) for the three noise sources defined above for the autoregulated gene circuit. The pole frequencies are given above by Eqs. 4.7 and 4.16. These closed-loop gains, $H_{j,k}(f)$, are applied with their respective noise sources in Eqs. 4.2 and 4.3 to determine $S_d(f)$, the total noise generated in the output dimer population.

Figure 4.8 shows calculated ESDs of the total dimer noise, S_d , and its components for the autoregulated gene circuit using the parameter values listed in Table 4.3. Matlab

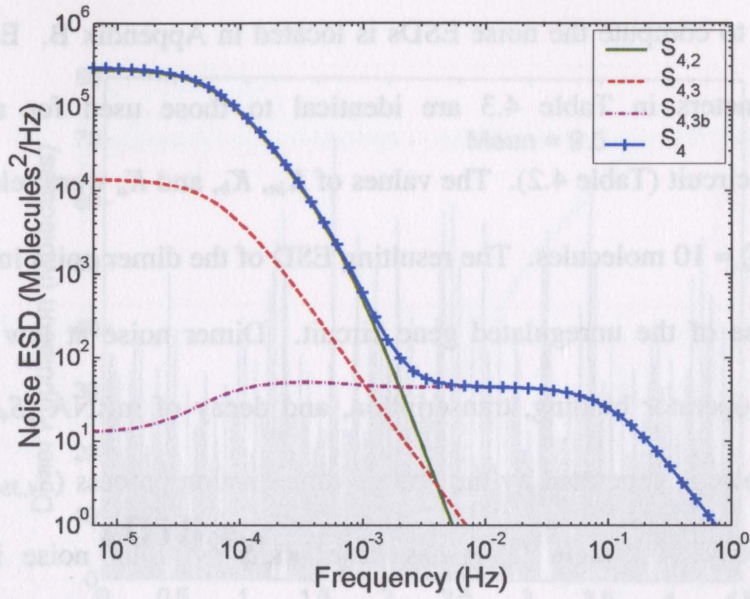


Figure 4.8. Total noise ESD in the steady-state dimer population, S_4 , and calculated contributions to S_4 by each of the noise sources, $S_{j,k}$, for the autoregulated gene circuit.

Table 4.3. Parameters used to model the autoregulated gene circuit

Parameter	Value	Reaction
K_{in}	0.003 s^{-1}	DNA-RNAP
K_c	0.1 s^{-1}	Transcription
$K_{mM} (M = 0-9)$	0.1 s^{-1}	Elongation
τ_r	100 s	Transcription delay
γ_r	0.0058 s^{-1}	mRNA decay
K_p	0.12 s^{-1}	MRNA-Ribosome
K_{mC}	0.1 s^{-1}	Translation
$K_{tM} (M = 0-9)$	0.1 s^{-1}	Elongation
τ_p	100 s	Translation delay
γ_p	0.0002 s^{-1}	Protein decay
K_f	0.0005 s^{-1}	Dimer formation
K_r	0.6 s^{-1}	Dimer dissolution
K_b	0.02 s^{-1}	Operator binding
K_u	0.1 s^{-1}	Operator unbinding

source code used to compute the noise ESDs is located in Appendix B. Except for K_{in} , the shared parameters in Table 4.3 are identical to those used for modeling the unregulated gene circuit (Table 4.2). The values of K_{in} , K_b , and K_u were selected to yield $\langle P \rangle \approx 100$ and $\langle Di \rangle \approx 10$ molecules. The resulting ESD of the dimer noise in Figure 4.8 is similar to the case of the unregulated gene circuit. Dimer noise at low frequency is primarily due to operator binding, transcription, and decay of mRNA ($S_{4,2}$), while the high frequency noise is generated by the protein dimerization process ($S_{4,3b}$). Again, the synthesis and decay of protein ($S_{4,3}$) has little effect on total noise in the dimer population.

4.2 Simulation Results

Monte Carlo simulations were performed using Biospreadsheet, an ESS software package developed by researchers at the University of Tennessee [75], which implements the Gibson and Bruck optimization of the Gillespie algorithm [19, 20]. In general, simulations were sampled at different rates ($f_s = 0.1, 1, \text{ and } 10 \text{ Hz}$) to generate output files containing 450,000 data points for the primary species: DNA, mRNA, P, and Di. Except for the step induction simulations in Section 4.2.2, the initial conditions for all molecular populations were set to their theoretical steady-state values. Figure 4.9 shows an example of a simulated time series for dimer protein, sampled every 10 s. To calculate the noise ESD at steady state, the first 18,000 data points were discarded to insure that any transient response was removed. The noise for an individual species was found by subtracting its mean value from its time series data. Noise ESDs for each sampling rate

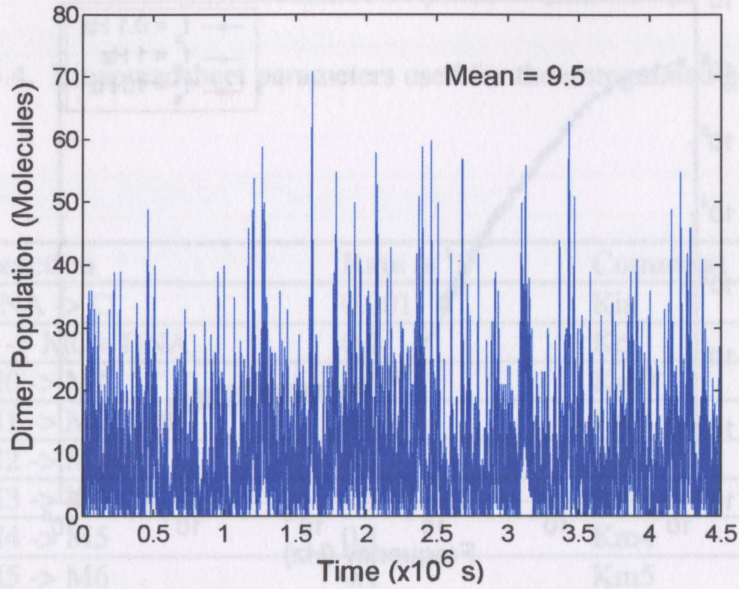


Figure 4.9. Simulated dimer population versus time at sampling rate $f_s = 0.1$ Hz.

were calculated by Welch's method in Matlab using Hanning windows of 12,000 samples with no overlap [66]. Matlab source code for computing the noise ESD of simulated data is located in Appendix C. Completed noise ESDs were then created by appending valid regions (low, medium, and high frequency) of ESDs from each of the three sampling rates to remove aliasing effects. The formation of a complete noise ESD for simulated data is shown in Figure 4.10.

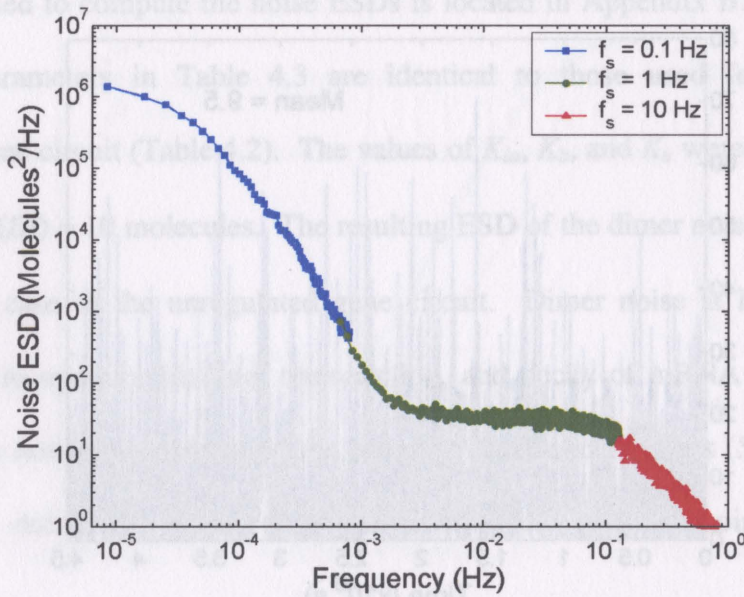


Figure 4.10. Representative noise ESD from ESS results created by appending spectrums from different sampling rates, f_s , to remove aliasing artifacts.

4.2.1 Unregulated Gene Circuit

To simulate the unregulated gene circuit, Biospreadsheet was setup using the reactions and rates listed in Table 4.4, which correspond directly to the reactions shown schematically in Figure 4.1 and the parameter values used previously from Table 4.2. Steady-state results from ESS are listed in Table 4.5 and agree nicely with the calculated steady-state values found using the equations in Table 4.1. The simulated noise ESD for the dimer population is shown in Figure 4.11 along with the calculated dimer noise ESD, S_4 from Figure 4.4. Agreement between simulation and theory is excellent in the passband ($f < 10^{-4}$ Hz). Error reaches nearly 100% at $f = 10^{-3}$ Hz and then becomes

Table 4.4. Biospreadsheet parameters used for the unregulated gene circuit

Reaction	Rate (s^{-1})	Comment
DNA \rightarrow C	0.001	Kin
C \rightarrow M0 + DNA	0.1	Kc
M0 \rightarrow M1	0.1	Km0
M1 \rightarrow M2	0.1	Km1
M2 \rightarrow M3	0.1	Km2
M3 \rightarrow M4	0.1	Km3
M4 \rightarrow M5	0.1	Km4
M5 \rightarrow M6	0.1	Km5
M6 \rightarrow M7	0.1	Km6
M7 \rightarrow M8	0.1	Km7
M8 \rightarrow M9	0.1	Km8
M9 \rightarrow mRNA	0.1	Km9
mRNA \rightarrow *	0.0058	Gamma-R
mRNA \rightarrow mC	0.12	Kp
mC \rightarrow T0 + mRNA	0.1	KmC
T0 \rightarrow T1	0.1	Kt0
T1 \rightarrow T2	0.1	Kt1
T2 \rightarrow T3	0.1	Kt2
T3 \rightarrow T4	0.1	Kt3
T4 \rightarrow T5	0.1	Kt4
T5 \rightarrow T6	0.1	Kt5
T6 \rightarrow T7	0.1	Kt6
T7 \rightarrow T8	0.1	Kt7
T8 \rightarrow T9	0.1	Kt8
T9 \rightarrow P	0.1	Kt9
P \rightarrow *	0.0002	Gamma-P
2P \rightarrow Di	0.001	2*Kf
Di \rightarrow 2P	0.6	Kr

Table 4.5. Comparison of steady-state populations for the unregulated gene circuit

Species	Calculated Mean	Simulation Mean
Operator (unbound)	0.990	0.990
mRNA	0.171	0.167
P	102	99.8
Di	8.74	9.5

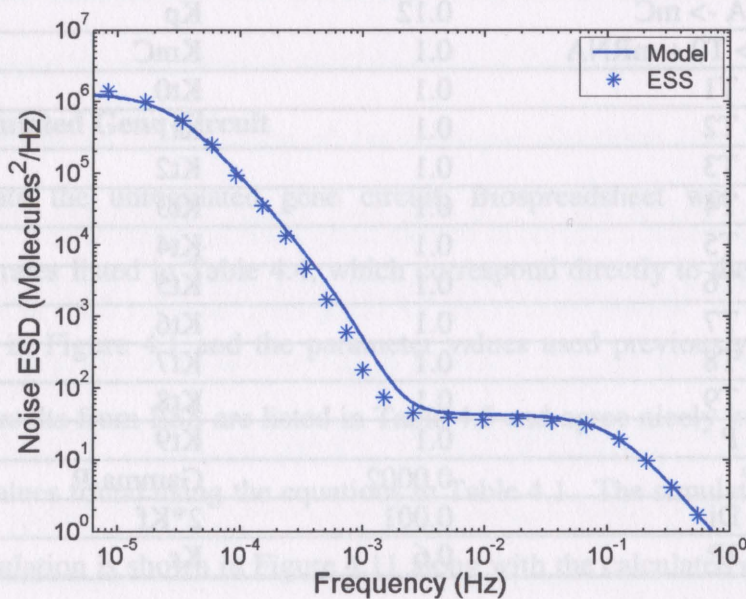
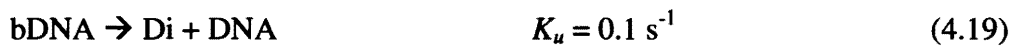


Figure 4.11. Comparison of dimer noise ESD between the analytical model and ESS for the unregulated gene circuit.

negligible for $f > 10^{-1}$ Hz. This error is mostly likely attributed to nonlinear effects that are not captured by the analytical model.

4.2.2 Autoregulated Gene Circuit

For the negatively autoregulated gene circuit, the Biospreadsheet setup in Table 4.4 was amended with the following two reactions to add negative feedback created by reversible binding of dimer molecules and DNA:



where bDNA represents bound DNA. All of the reaction rates for this simulation were taken from Table 4.3. Steady-state results from ESS are listed in Table 4.6 and agree well with the calculated steady-state values (Eqs. 4.10-4.13). The simulated noise ESD of the dimer population is shown in Figure 4.12 along with the calculated dimer noise ESD, S_d from Figure 4.8, for the autoregulated gene circuit. Even with an average dimer population of just 10 molecules, agreement between simulation and theory is excellent throughout the entire spectrum.

Up to now, the transcription and translation delays have each been set equal to 100 s. To study the effects of delay time in mRNA and protein synthesis, Biospreadsheet was configured using the setup in Table 4.7 for the autoregulated gene circuit. With $M = 10$, both K_{mM} and K_{tM} were first set equal to 0.4 s^{-1} , resulting in $\tau_r = 25$ and $\tau_p = 25$ s. Figure 4.13 shows approximated deterministic time-domain responses of the dimer protein population for step induction when the total time delay ($\tau_r + \tau_p$) is 50, 200, and 800 s

Table 4.6. Comparison of steady-state populations for the autoregulated gene circuit

Species	Calculated Mean	Simulation Mean
Operator (unbound)	0.344	0.351
mRNA	0.178	0.181
P	107	108
Di	9.52	10.3

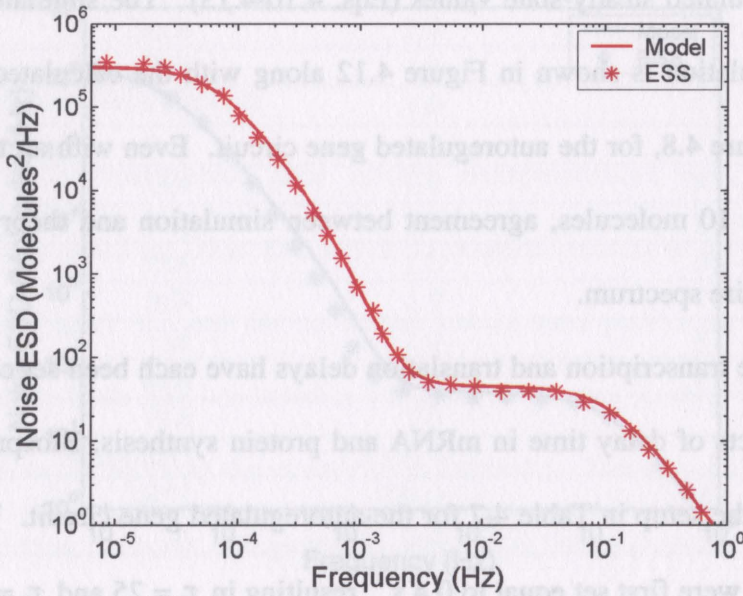


Figure 4.12. Comparison of dimer noise ESD between the analytical model and ESS for the autoregulated gene circuit.

Table 4.7. Biospreadsheet parameters used to study the effects of time delays in the autoregulated gene circuit

Reaction	Rate (s⁻¹)	Comment
DNA -> C	0.1	Kin
C -> M0 + DNA	0.1	Kc
M0 -> M1	0.4	Km0
M1 -> M2	0.4	Km1
M2 -> M3	0.4	Km2
M3 -> M4	0.4	Km3
M4 -> M5	0.4	Km4
M5 -> M6	0.4	Km5
M6 -> M7	0.4	Km6
M7 -> M8	0.4	Km7
M8 -> M9	0.4	Km8
M9 -> mRNA	0.4	Km9
mRNA -> *	0.0058	Gamma-R
mRNA -> mC	0.05	Kp
mC -> T0 + mRNA	0.1	KmC
T0 -> T1	0.4	Kt0
T1 -> T2	0.4	Kt1
T2 -> T3	0.4	Kt2
T3 -> T4	0.4	Kt3
T4 -> T5	0.4	Kt4
T5 -> T6	0.4	Kt5
T6 -> T7	0.4	Kt6
T7 -> T8	0.4	Kt7
T8 -> T9	0.4	Kt8
T9 -> P	0.4	Kt9
P -> *	0.0005	Gamma-P
2P -> Di	0.0005	2*Kf
Di -> 2P	0.6	Kr
Di + DNA -> bDNA	0.1	Kb
bDNA -> Di + DNA	0.1	Ku

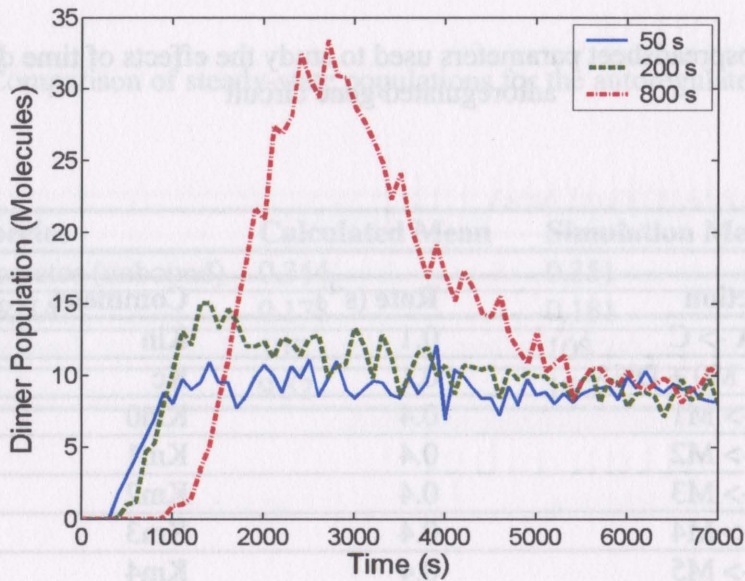


Figure 4.13. Overshooting of dimer steady-state population in time domain as total delay time is increased from 50 to 800 s for the autoregulated gene circuit.

($K_{mM} = K_{IM} = 0.4, 0.1, \text{ and } 0.025$, respectively). Each response was found by averaging 10 simulated trajectories with gene copy number = 1 and all other species initially zero. At $t = 0$, a copy of the gene became available (by transformation, for example) and the molecular populations increased to their steady-state values. With total delay < 50 s, the dimer population reached steady-state without overshooting its mean value of 10 molecules. Overshooting became evident when the total delay was > 200 s and was significant when the total delay was > 800 s, which was simply due to the fact that many proteins were already in production when the first repressors (i.e., dimers) finally became active and down-regulated gene expression. Figure 4.14 shows the simulated and calculated noise ESD, $S_d(f)$, of the steady-state dimer population with total time delays

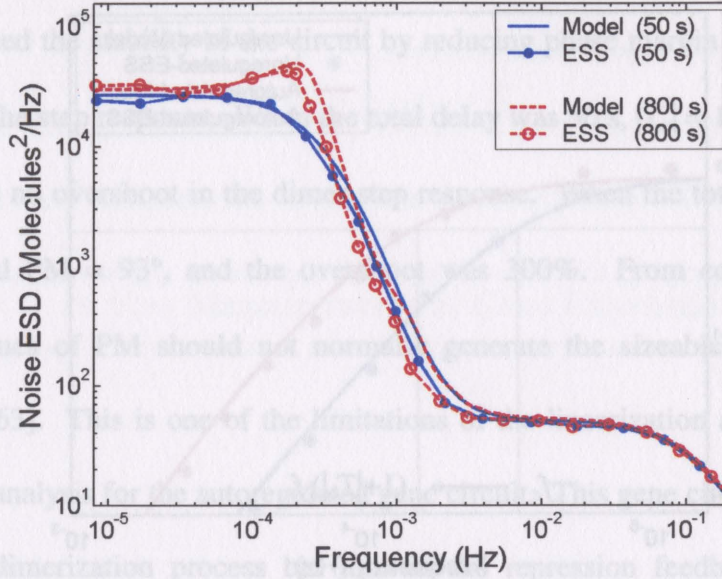


Figure 4.14. Peaking in the dimer noise ESD for the autoregulated gene circuit as total delay time is increased from 50 to 800 s.

($\tau_r + \tau_p$) of 50 and 800 s for the autoregulated circuit. Increasing the delay time from 50 to 800 seconds produced peaking in the noise ESD. This frequency peaking is the FD manifestation of the time-domain overshoot observed in Figure 4.13.

4.3 Discussion

Noise performance of regulated gene networks is dependent on the loop transmission $T(f)$ of the closed-loop circuit [14]. Consider the unregulated and autoregulated dimer noise ESDs, shown in Figure 4.15, calculated using the parameter values in Tables 4.2 and 4.3, respectively. For the unregulated gene circuit, the cutoff frequency, $f_c = 2.8 \times 10^{-5}$ Hz, occurred at the half-power point indicated by the horizontal dashed line. This

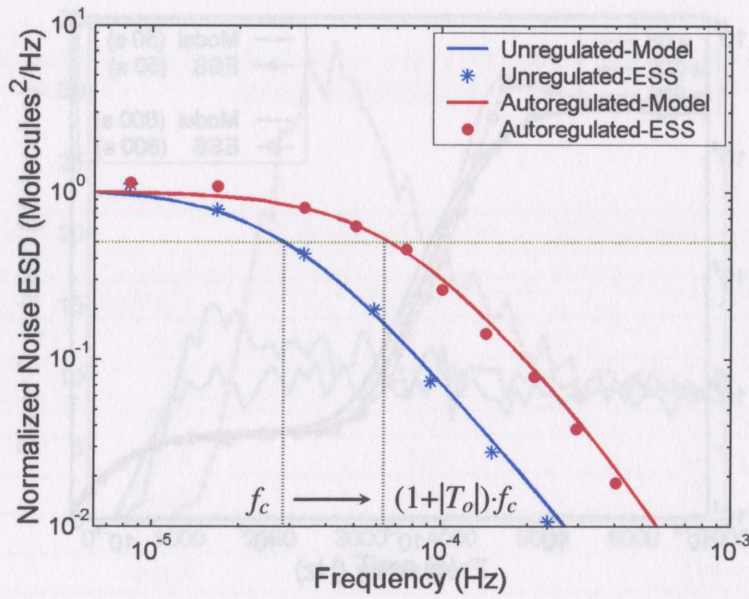


Figure 4.15. Bandwidth comparison of the unregulated and autoregulated gene circuits.

cutoff frequency was determined by the protein pole, f_p given for Eq. 4.7, which was set primarily by the protein decay rate γ_p . For the negatively autoregulated gene circuit with the same protein pole, the bandwidth was expected to be increased by a factor of $1+|T(0)|$. Applying Eq. 4.17 with the parameter values in Table 4.3, $|T(0)| = 1.23$, and the cutoff frequency of the negatively regulated gene circuit was extended to 6.2×10^{-5} Hz (Figure 4.15) exactly as predicted [14].

Depending on the total amount of delay time and the strength of the promoter, molecular species may overshoot their steady-state targets [73]. Overshoot in population of species is typically undesirable because it costs nutrients and can have toxic effects on the cell. The effect of the delays, τ_r and τ_p , on the loop transmission of the autoregulated

gene circuit is shown in the calculated (Eq. 4.17) Bode plot in Figure 4.16. The increased delays impacted the stability of the circuit by reducing phase margin (PM), which led to overshoot in the step response. When the total delay was 50 s, $|T_o| = 1.7$ with $PM = 119^\circ$, and there was no overshoot in the dimer step response. When the total delay was 800 s, $|T_o| = 1.7$ and $PM = 93^\circ$, and the overshoot was 300%. From control theory, these sufficient values of PM should not normally generate the sizeable overshoot seen in Figure 4.13 [63]. This is one of the limitations of the linearization applied during loop transmission analysis for the autoregulated gene circuit. This gene circuit is nonlinear not only in the dimerization process but also in the repression feedback stage. Dimer molecules can block initiation of transcription, but they cannot reverse this process. So,

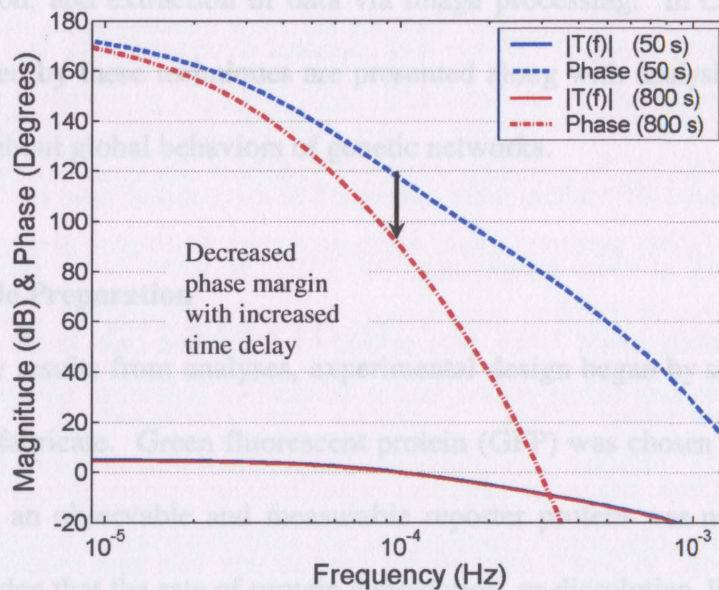


Figure 4.16. Bode plot of calculated loop transmission, $T(f)$, for the autoregulated gene circuit when total delay time is increased from 50 to 800 s.

even as dimers try to halt gene expression, proteins are still translated until the level of mRNA is decreased by degradation machinery.

Despite its shortcomings, the FD method presented in this chapter is a powerful analytical tool that yields relatively simple and easy-to-use equations. As long as caution is exercised when applying this type of analysis, the results can reveal key features of gene circuit behavior, and as demonstrated, remain accurate even at low populations of molecular species.

Chapter 5

In Vivo Measurements of Gene Expression

This chapter describes the development and implementation of experimental protocols used to acquire *in vivo* measurements of stochastic gene expression in bacteria cells. The procedures are discussed in three parts: preparation of biological samples, data acquisition, and extraction of data via image processing. In Chapter 6, measured results obtained by these techniques are presented along with analysis that reveals some new insights about global behaviors of genetic networks.

5.1 Sample Preparation

Guided by results from analyses, experimental design began by selecting the genetic constructs to fabricate. Green fluorescent protein (GFP) was chosen as the desired gene product since an observable and measurable reporter protein was needed. With the *a priori* knowledge that the rate of protein degradation, or dissolution, limits the bandwidth of gene circuits, it was decided that three variations of GFP would be constructed, each having a different half-life (i.e., degradation rate). The variants included wild-type (WT)

GFP, GFP-asv, and GFP-aav, listed here in order of decreasing protein half-life, λ . Each suffix corresponds to an amino acid sequence appended to the polypeptide chain that targets the GFP molecule for degradation by bacterial proteases. At 37°C, the approximate half-life of wild-type GFP is 24 hr, while $\lambda \approx 110$ min and $\lambda \approx 60$ min for GFP-asv and GFP-aav, respectively [76].

Unregulated gene circuits were constructed from plasmid pGFP-asv, graciously donated by Elowitz [25], which contained the $P_{\text{Lact0-1}}$ promoter [77] followed by the asv-mutant of the *gfp* gene. This high copy number plasmid also encoded the kanamycin acetyltransferase gene, *KmR*, which conferred the host cell resistance to kanamycin and acted as the selectable marker of the plasmid. The donated plasmid was used as the basis for generating the other variant forms of GFP described above. Figure 5.1(a) shows the genetic constituents of the derived plasmids: pGFP-WT, pGFP-asv, and pGFP-aav. For construction of all three cases, pGFP-asv was digested overnight with *StuI* and *HinDIII* (New England Biolabs), and the parent vector was gel purified. Synthesized oligonucleotide sets for each GFP variant were combined in a 1:1 molar ratio in ligation buffer and allowed to anneal at room temperature for 1 hr. A 100:1 molar ratio of double-stranded oligonucleotide insert DNA was added to a ligation mix containing 100 ng digested, gel-purified vector DNA. Ligations were conducted in 25- μ l reactions containing 4U T4 DNA ligase in LigaFast rapid ligation buffer (Promega) at room temperature for 10 min. A 5- μ l sample of the ligation mix was then transformed into *E. coli* cells per manufacturer's instructions.

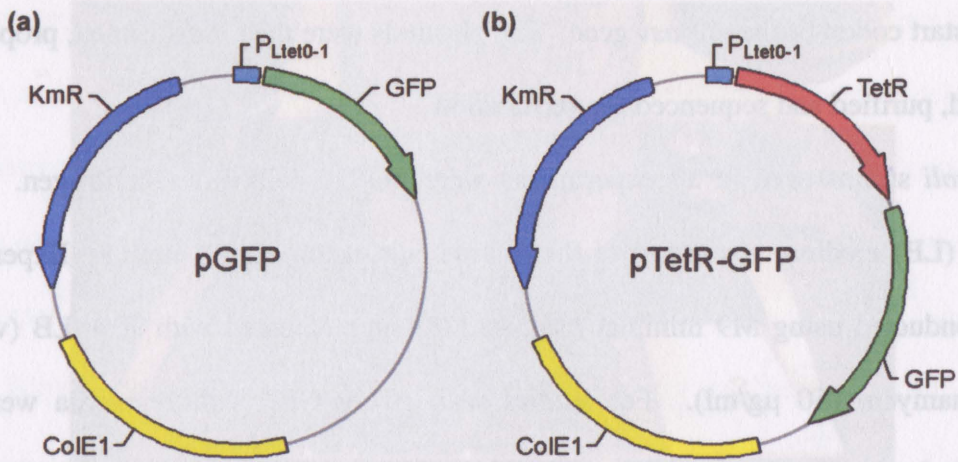


Figure 5.1. Plasmids constructed for *in vivo* measurements of gene expression. (a) pGFP with constitutively expressed variants of *gfp*, and (b) pTetR-GFP with negatively autoregulated *tetR* plus GFP-asv reporter. ColE1 is the origin of replication.

To make an autoregulated gene circuit, *tetR* was inserted into pGFP-asv to produce pTetR-GFP, as shown in Figure 5.1(b). The binding of TetR protein and the P_{Ltet0-1} promoter generated negative feedback for autoregulation of the *tetR* and *gfp-asv* genes. The *tetR* gene was amplified by polymerase chain reaction (PCR) from Repressilator plasmid donated by Elowitz [25] and cloned into pCR.2.1-TOPO (Invitrogen). The reverse primer was modified to include a stop codon at the end of the normal coding sequence to generate mature TetR protein with wild-type half-life. In order to minimize a potential difference in the burst rates, the untranslated region and ribosomal binding sites between the P_{Ltet0-1} promoter and the *gfp* gene in pGFP-asv were duplicated for *tetR* by including a copy of the untranslated region between the promoter and the KpnI site 5' of

the atg start codon of the *gfp-asv* gene. The plasmids were then transformed, propagated, screened, purified and sequenced for verification.

E. coli strains used in all experiments were TOP10 cells from Invitrogen. Luria-Bertani (LB) medium was used for the routine cultivation of cell strains. Experiments were conducted using M9 minimal medium [78] supplemented with 10% LB (vol:vol) and kanamycin (50 µg/ml). For studies with pTetR-GFP, culture media were also augmented with anhydrotetracycline (ATc) to induce gene expression. The M9 medium was chosen to reduce background fluorescence during imaging, and the LB medium was added to provide extra nutrients for cell growth. Cell cultures containing each of the plasmids of interest were grown overnight, diluted back 1:10 in fresh media (M9 + 10% LB) and allowed to recover for 1 hr prior to deposition on slides. Slides were prepared using M9 + 10% LB with 1% low-melt electrophoresis-grade agarose (FMC). Melted agar medium (1 ml) was deposited onto glass microscope slides (Fisher Scientific) and left to solidify. Once cooled to room temperature, a heated platinum wire was used to melt channels in the agarose to permit air exchange within the sample. Then, a 10-µl solution of transformed *E. coli* cells in exponential growth phase was spread onto the cooled agarose and covered with a glass slip, as shown in Figure 5.2. The cover slip and solidified agarose helped immobilize the cells. Before imaging, samples were incubated for ~1 hr at room temperature, 28°C, or 32°C, depending upon the experiment. This incubation period allowed the cells time to acclimate to their new environment.

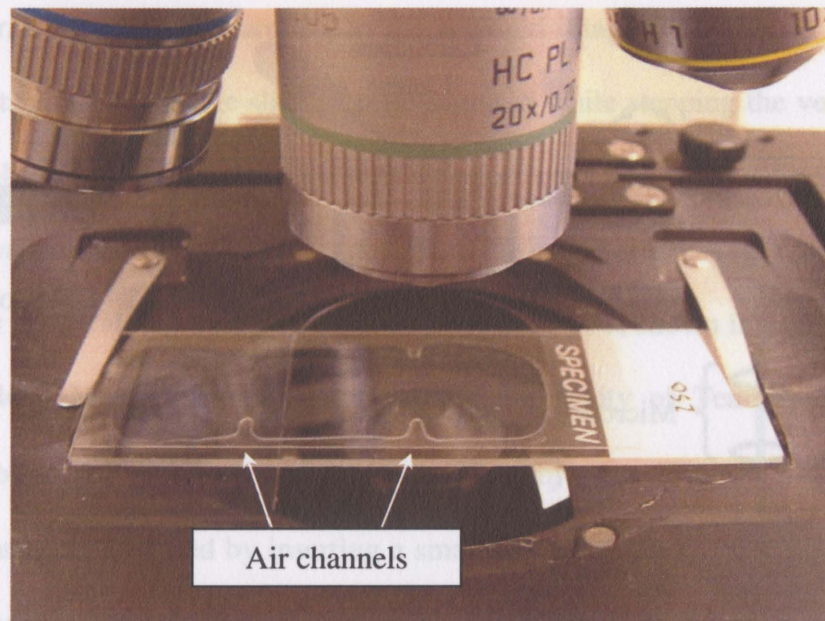


Figure 5.2. Biological sample ready for microscopy. A cover slip lies on top of *E. coli* cells deposited on culture media, supported by a glass slide. The air channels were pinched off sometimes after the cover slip was pressed down.

5.2 Data Acquisition

Confocal microscopy is a powerful imaging modality that facilitates 3-dimensional (3-D) image reconstruction with sub-micron resolution [79, 80]. When equipped with an excitation source, such as a mercury lamp or laser, a confocal microscope can also be used to observe fluorescence of a substance, such as GFP. The primary components of a laser confocal microscope are shown in Figure 5.3. A pulsed laser beam is deflected by a dichroic mirror and guided through the microscope by scanning mirrors. These mirrors rotate under computer control to scan the laser beam over the sample. During laser excitation, fluorescent molecules in the sample are stimulated by the incident high-energy radiation, which results in emission of photons (fluorescence) with lower energy. This

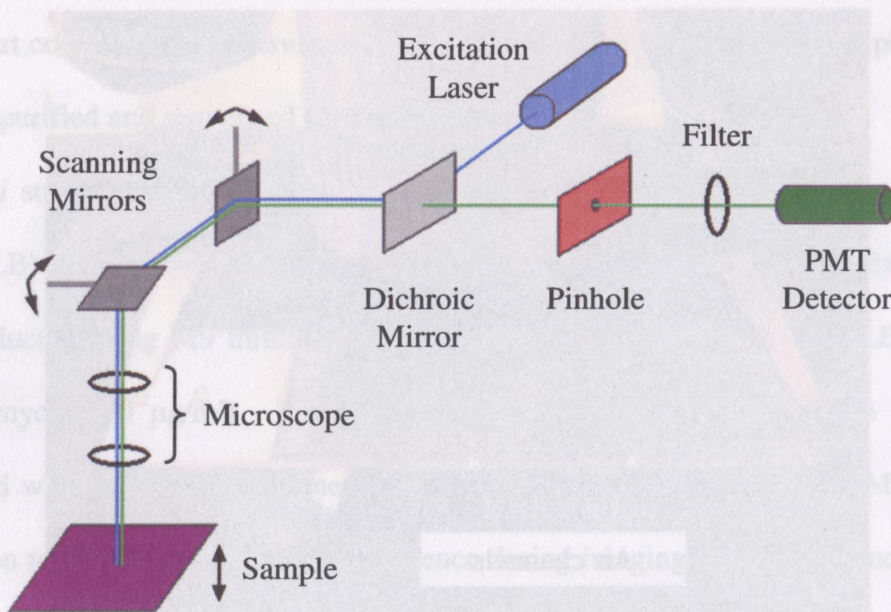


Figure 5.3. Primary components of a laser confocal microscope.

fluorescent light is directed back through the microscope by the scanning mirrors, passes through the dichroic mirror due to the difference in wavelength, and is amplified by a detector, such as a photomultiplier tube (PMT). However, before the fluorescent light reaches the detector, it must pass through a pinhole screen, which forms a conjugate plane with the sample plane, thereby blocking out-of-focus light. That is, the confocal pinhole allows only light originating from a specific point in the sample to reach the detector. As the scanning mirrors raster the laser across the sample, a computer then digitizes the PMT output to form a 2-dimensional (2-D) image depicting the thin section of the sample that is in the focal plane. Adjusting the size of the pinhole effectively increases the thickness of the viewable area in the sample by allowing more light to reach

the detector, but sacrifices sharpness in the acquired images. Finally, to construct 3-D images, a stack of 2-D image slices can be acquired while stepping the vertical position of the sample, typically accomplished with a piezoelectric stage for precision control.

The *in vivo* expression of GFP for all constructed gene circuits was recorded by time-lapsed laser confocal microscopy using the Leica TCS SP2, shown in Figure 5.4, located in the Biology Microscopy Laboratory at the University of Tennessee [81]. Cells deposited on slides were viewed under a 20× objective (Figure 5.2). Local cell environment was monitored by inserting a small thermocouple into the agarose between the slide and cover slip. The temperature was maintained by adjusting a heating lamp near the microscope stage, and aluminum foil sheeting was used to block direct light from the lamp to eliminate interference with fluorescence measurements. Temperature variation during the course of an experiment was $\pm 1^\circ\text{C}$. The laser and confocal microscope settings used to observe fluorescence of GFP are given in Table 5.1. The laser's excitation wavelength was 488 nm, and the detected green fluorescent light was band-limited from 500-550 nm. To prevent photobleaching of GFP [82], the laser power was set as low as possible (5-10% of max power) and the gain of the PMT was increased as high as possible (650-720 V), while preventing saturation of the detector's analog-digital converter and preserving favorable signal-to-noise ratio in acquired images. The confocal pinhole was adjusted such that the thickness of an image in the sample plane was $>1\ \mu\text{m}$ (i.e., more than the height of the cells). The microscope zoom was adjusted to view a $40\ \mu\text{m} \times 40\ \mu\text{m}$ area, and the size of each image was 512×512 pixels. The laser beam was rastered at 800 lines/sec, so that an image was acquired in <1 sec. Output



Figure 5.4. Leica TCS SP2 laser confocal microscope.

Table 5.1. Settings used for laser confocal microscopy

Parameter	Setting
Excitation Laser Wavelength	488 nm
Excitation Laser Power	5-10%
Detector Filter	500-550 nm
PMT (Gain)	650-720 V
Objective	20×
Pinhole Size	4.0 Airy
Image Size	40 μm \times 40 μm
Image Resolution	512 \times 512 pixels
Image Type (Pixel Range)	8-bit (0-255)
Scan Type	xyzt
Scan Rate	800 Hz
Line Averaging	1
Frame Averaging	1
Number of Slices	12
Distance Between Slices	0.28 μm

images were 8-bit Tagged Image File Format (TIFF), with 0-255 grayscale values. To record time-sampled fluorescence data, cells were imaged every 5 min (sampling rate = 1/300 Hz). In the interest of speed, the line and frame (i.e., image) averaging capabilities were disabled during the time-lapsed acquisitions. However, high-quality before and after images, like those shown in Figure 5.5, were captured with 4 frames averaged. Ideally, observation periods were as long as possible, with some as long as 8-9 hr. However, some experiments were as short as 3 hr because cell fluorescence faded severely. As seen in Figure 5.5, the image size of $40\ \mu\text{m} \times 40\ \mu\text{m}$ was needed to provide enough viewable area for the cells to grow and divide over the duration of the experiment.

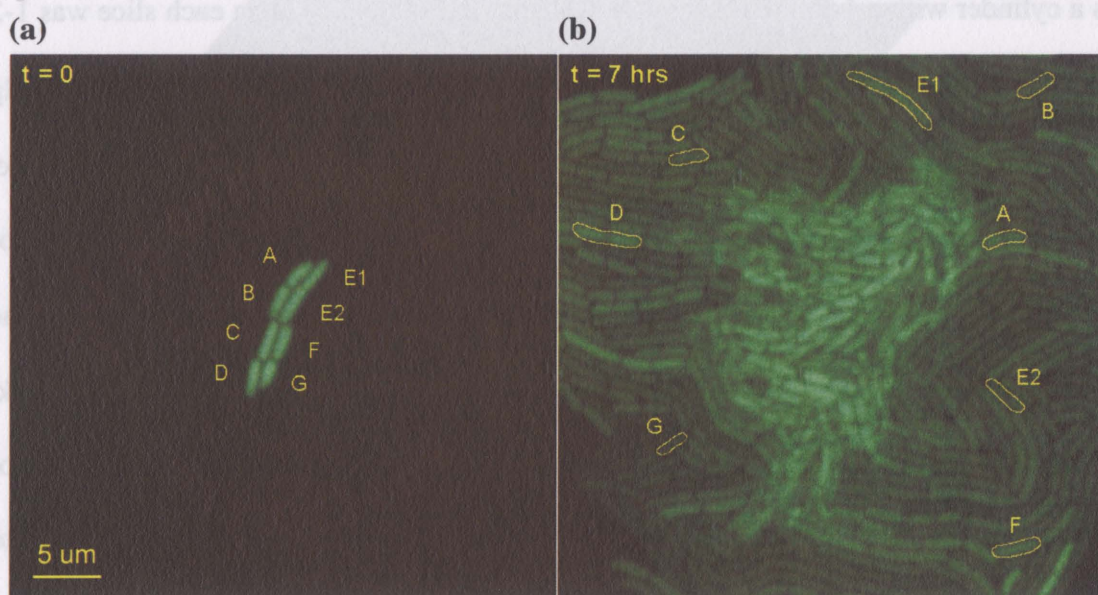


Figure 5.5. Before (a) and after (b) images for a 7-hour experiment using *E. coli* cells with pGFP-asv at 26°C. Cells were labeled for tracking. E1 and E2 had not yet divided at $t = 0$ and therefore shared some history during the observation.

Even for short periods of time, exposure to air caused the culture media on the slides to evaporate noticeably under a microscope. The result was that the cells eventually fell out of the focal plane. To remedy this problem, the confocal microscope was set up to acquire xyzt scans, which yielded images (in the xy-plane) in stacks (z-dimension) sampled over time. At each time sample, 12 frames were acquired as the microscope's piezoelectric stage was stepped $0.28\ \mu\text{m}$ between each exposure. Two disadvantages of this technique were that the cells received 12 times the radiation dose and that the number and total size of data files increased 12-fold. Figure 5.6 shows a stack of images acquired for a single time sample and provides a way to estimate the sample thickness represented by each frame. Because the frames above and below the best-focused image appeared identical to it, the thickness of each image slice was greater than that of the cells. If a cell was a cylinder with a $1\text{-}\mu\text{m}$ diameter (see scale bar in Figure 5.6), then each slice was $1\text{-}2\ \mu\text{m}$ thick given that the step size was $0.28\ \mu\text{m}$. Although the xyzt-type scan did help correct for sample drift, this alone was not enough to keep the cells in focus over time. During evaporation of the culture media, the cells tended downward at an average rate of $\sim 1\ \mu\text{m}/\text{min}$. To compensate for this effect, the manual fine adjustment for the microscope stage was repositioned as needed before the acquisition of each image stack. As described above, a stack of images was obtained every 5 min for the entire length of observation. In the end, the image from each stack with the best focus was used to measure the total cell fluorescence for the respective time sample. Since each frame of a stack was acquired in $<1\ \text{sec}$ and the sampling period was 5 min, sampling rate error introduced by this method was negligible.

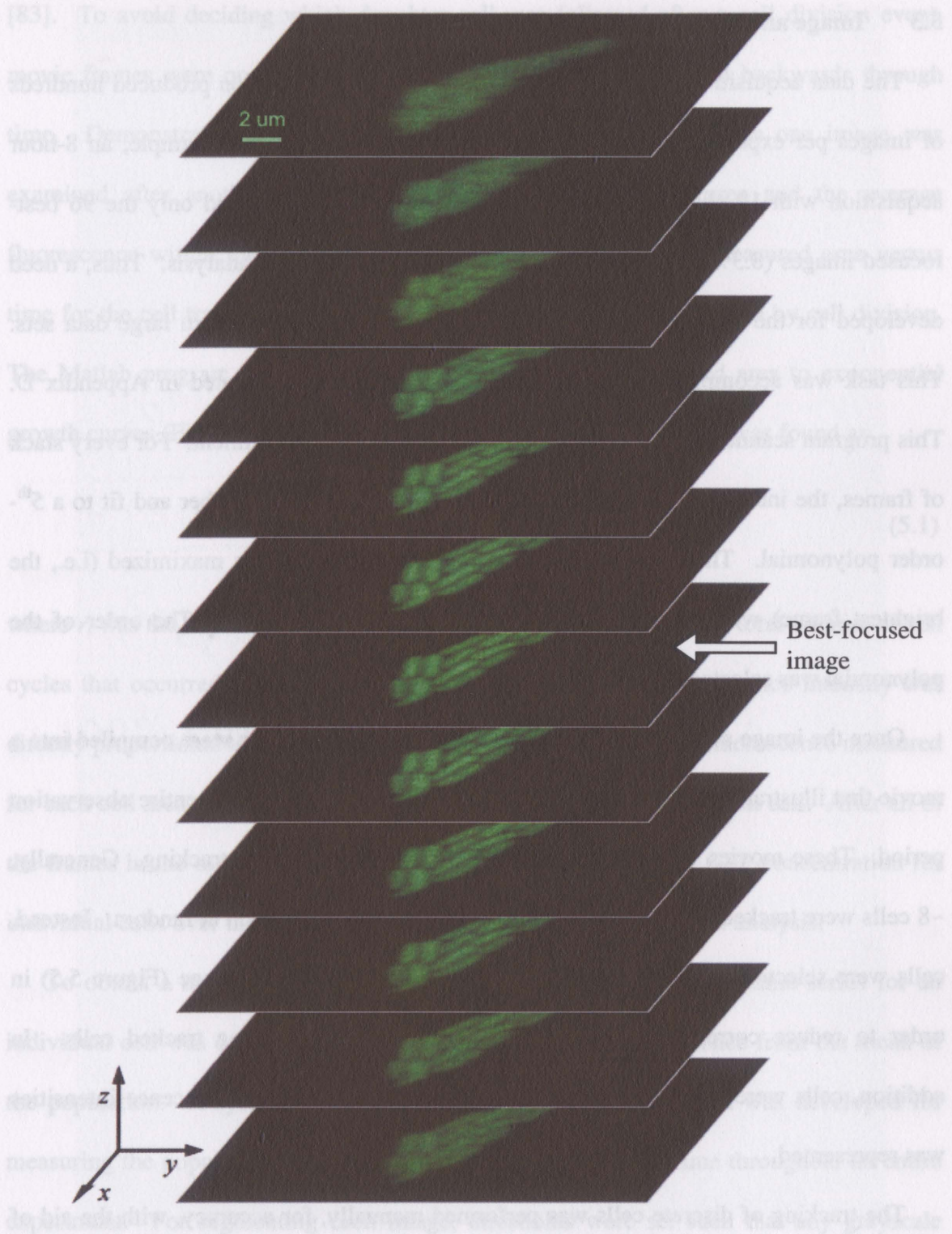


Figure 5.6. Stack of cropped images acquired for a single time sample. The stage was stepped $0.28 \mu\text{m}$ between each exposure (z-dimension not to scale).

5.3 Image and Data Processing

The data acquisition procedure described in the previous section produced hundreds of images per experiment, many of which were out of focus. For example, an 8-hour acquisition with 12 slices every 5 min resulted in 1,152 frames, and only the 96 best-focused images (8.3% of the total) were actually used for further analysis. Thus, a need developed for the ability to automatically extract focused images from large data sets. This task was accomplished using a custom Matlab program, located in Appendix D. This program scanned through every image generated by an experiment. For every stack of frames, the integral of each image was plotted against image number and fit to a 5th-order polynomial. The image number where this polynomial was maximized (i.e., the brightest frame) was then taken as the image with the best focus. The order of the polynomial was selected based on trial and error results.

Once the image stacks were processed, the best-focused images were compiled into a movie that illustrated the growth and time course of all the cells for an entire observation period. These movies were then used to select individual cells for tracking. Generally, ~8 cells were tracked in each movie. Tracked cells were not chosen at random. Instead, cells were selected so that they had the least amount of shared lineage (Figure 5.5) in order to reduce correlation in measured fluorescence data among tracked cells. In addition, cells were chosen so that a fairly even distribution of fluorescence intensities was represented.

The tracking of discrete cells was performed manually, for accuracy, with the aid of an image-processing tool called ImageJ, available from the National Institutes of Health

[83]. To avoid deciding which daughter cell was followed after a cell division event, movie frames were opened in reverse order and cells were tracked backwards through time. Demonstrated in Figure 5.7, a cell of interest was circled as one image was examined after another. ImageJ then measured the selected area and the average fluorescence within the drawn boundary. Figure 5.8 shows the measured area versus time for the cell tracked in Figure 5.7. The discontinuities were caused by cell division. The Matlab program in Appendix E was used to fit the measured area to exponential growth curves (Figure 5.8). The average doubling time, T_d , for a cell was found as

$$T_d = \frac{1}{N} \sum_{i=1}^N \frac{\ln(2)}{r_i}, \quad (5.1)$$

where r_i was the exponential growth rate per cell cycle and N was the total number of cell cycles that occurred during the observation. Assuming that fluorescence intensity was directly proportional to the number of GFP molecules, the average fluorescence measured for each cell area corresponded directly to the GFP concentration in that cell. After all of the frames in the movies had been examined, time trajectories of GFP concentration for individual cells over the length of each experiment were available for analysis.

To obtain a measurement of noise in GFP expression, the noise time series for an individual cell was defined as the deviation of that cell's fluorescence from the mean of the population. A systematic method using threshold segmentation was developed for measuring the population's mean fluorescence as a function of time throughout an entire experiment. For segmenting each image, thresholds were set such that any grayscale pixel value above the threshold was taken to be fluorescence from a cell, while pixels

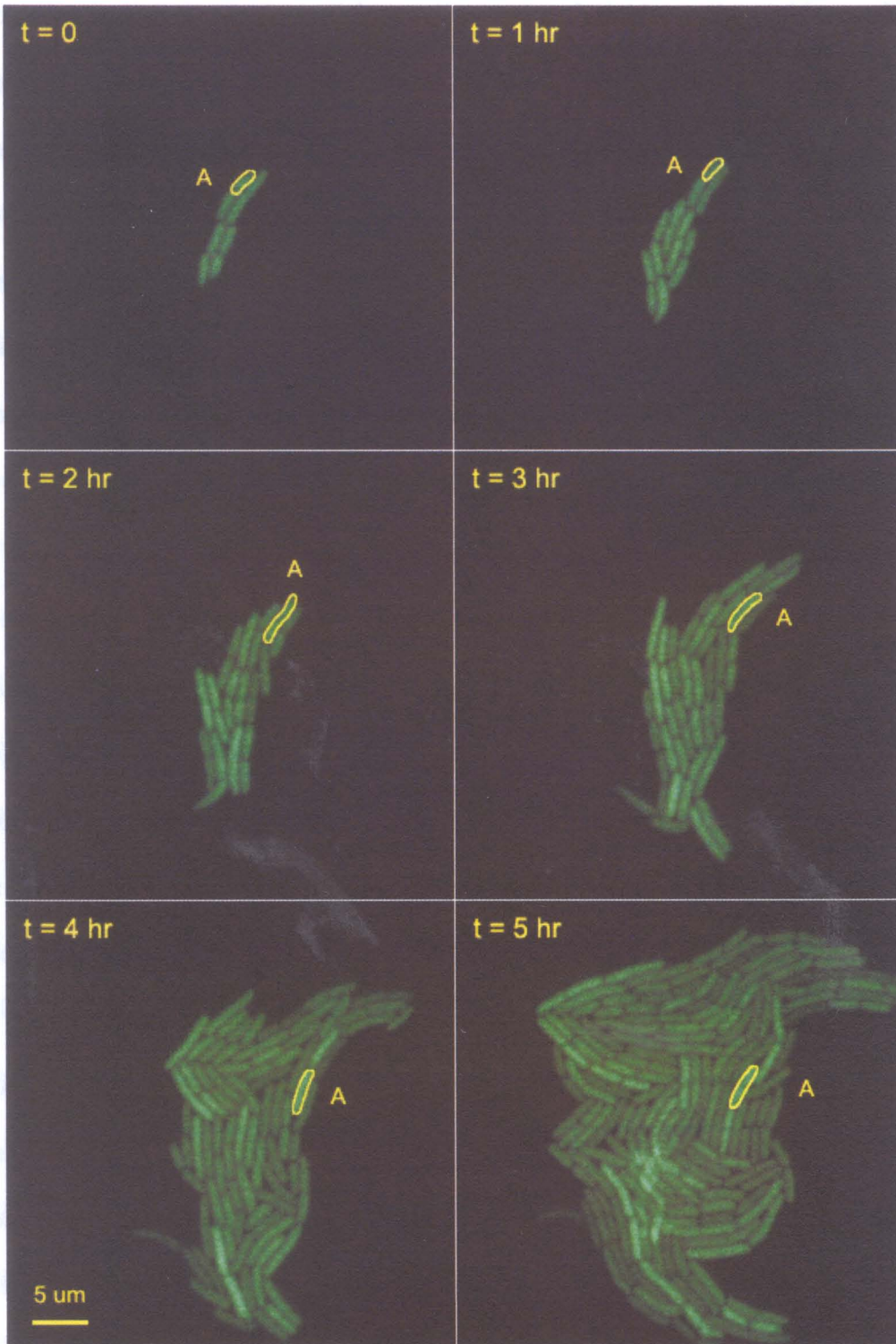


Figure 5.7. Tracking a single cell through selected frames of a movie.

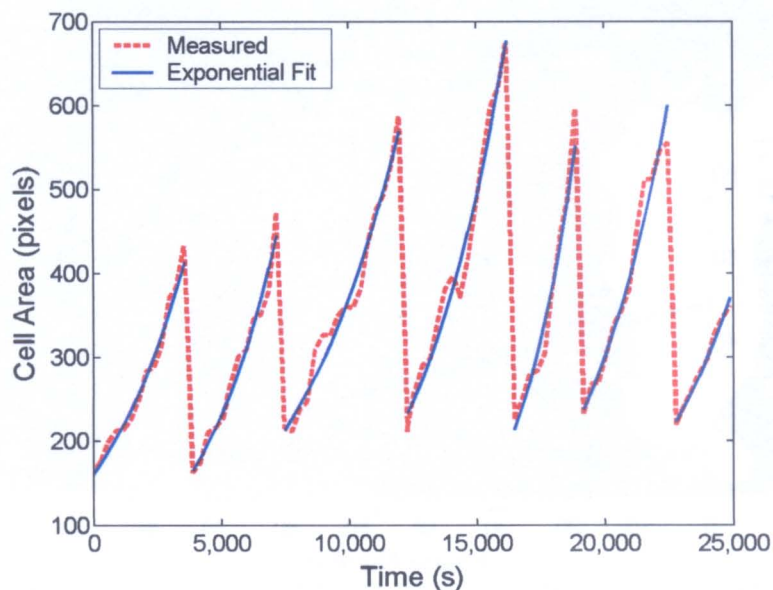


Figure 5.8. Measured cell area and exponential growth for a single cell.

below the threshold were considered to be part of the image background. To ensure that the thresholds were set above background levels, measurements of dark regions in images were taken using the Matlab program in Appendix F. This program scanned every frame of a movie, analyzed a user-defined area (e.g., upper left-hand corner), and returned the mean and max pixel values in the selected region. Figure 5.9 shows mean and max background values (in arbitrary units, a.u.) measured for a complete set of images from an experiment. The max values corresponded to speckles, or noise, in the image backgrounds. Next, the minimum fluorescence values of the hand-traced cells were plotted and fitted to a line. This line was then shifted down as low as possible, while remaining above the background noise, and used to define the decision thresholds for

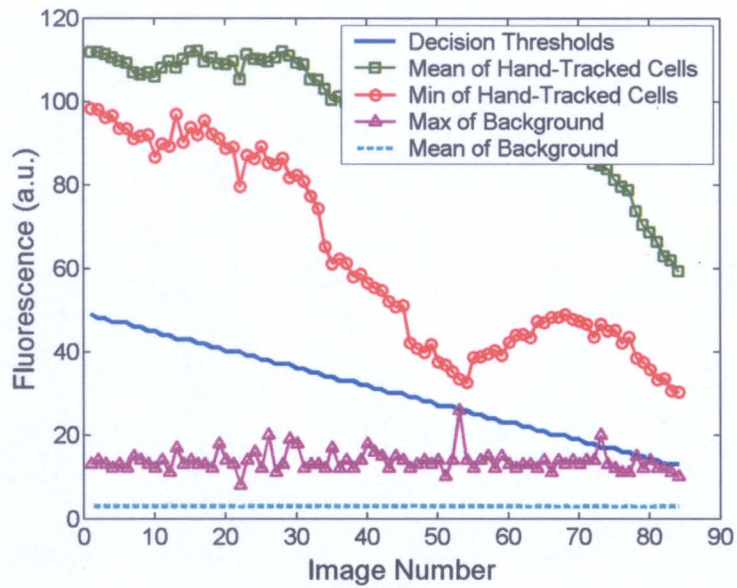


Figure 5.9. Selection of thresholds used to segment cells from their image background.

image segmentation (Figure 5.9). These threshold values were applied to images by the Matlab program in Appendix G. This program integrated the total fluorescence in each movie frame, and used the predetermined thresholds to segment images, as shown in Figure 5.10, and to calculate the total cell area per frame (i.e., time sample). Then, the background-corrected mean fluorescence of the cell population at each time sample was determined as follows: total background fluorescence (mean of background times size of image) was subtracted from the total image fluorescence to get total cell fluorescence; total cell fluorescence was divided by total cell area to obtain average fluorescence per cell pixel; average fluorescence/pixel was smoothed with an averaging filter given by

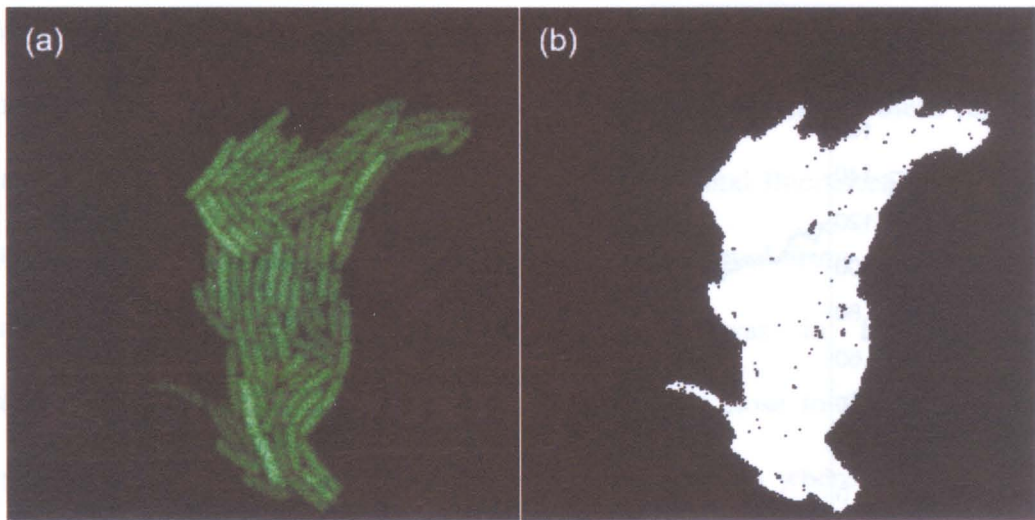


Figure 5.10. Image segmentation used to determine total cell area per image. (a) Input image from a single time sample, and (b) the corresponding thresholded image.

$$y[n] = \frac{1}{M} \sum_{i=1}^M x \left[n + \frac{2i - M - 1}{2} \right], \quad (5.2)$$

where M was an odd number set roughly as 10% of the total number of samples. Finally, the noise for a single, tracked cell was found, as shown in Figure 5.11, by subtracting the background and the population means from the measured fluorescence of the cell. After repeating this procedure for all cells tracked in a movie, a set of single-cell noise measurements, as shown in Figure 5.12, was obtained for each experiment performed.

Close examination of images revealed cell crowding that sometimes occurred near the end of long observation periods. Instead of continuing to spread out laterally, cells began to overlap each other as they grew. This phenomenon, indicated by the brighter and somewhat blurred region in center of Figure 5.5(b), skewed the calculations for the mean

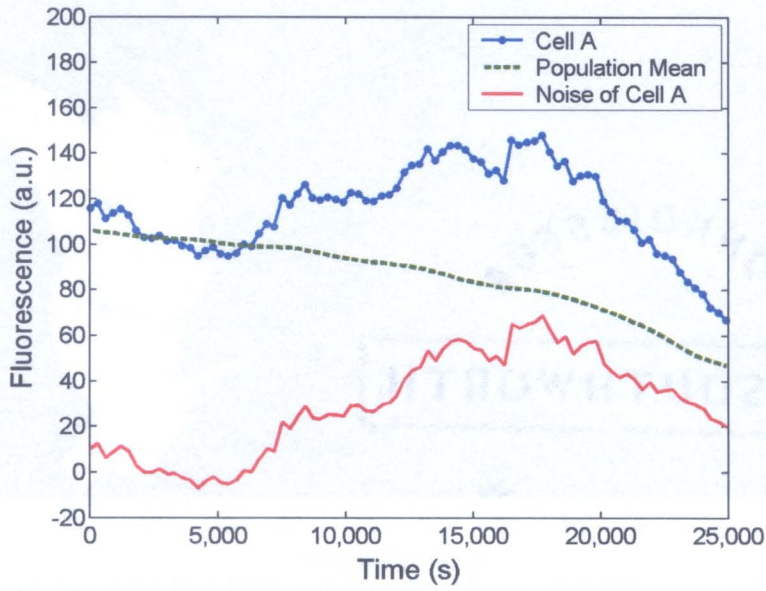


Figure 5.11. Noise measured for a single cell. Noise (red) for a cell is defined as its background-corrected fluorescence (blue) minus the population mean (green).

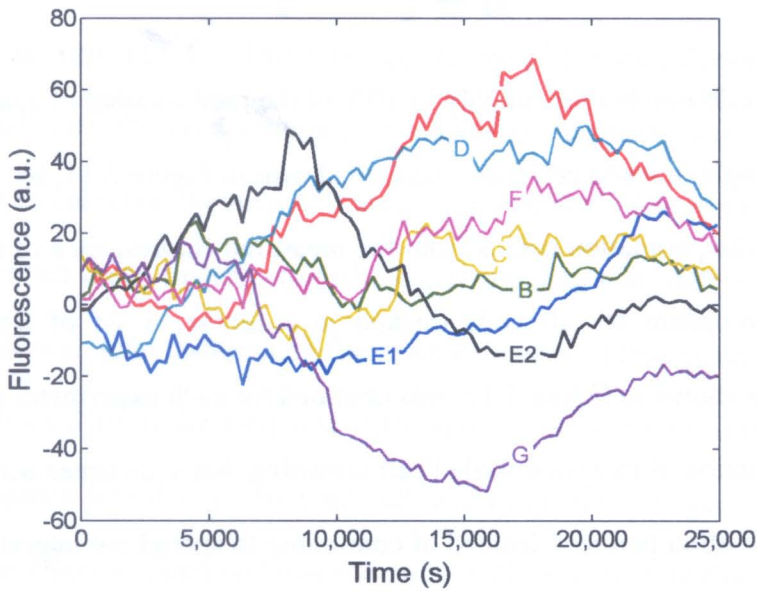


Figure 5.12. Noise in GFP concentrations for the 8 labeled cells of Figure 5.5.

fluorescence of the cell population. To correct for this effect, these specific areas were manually traced in ImageJ. Then, the total fluorescence and area of these regions was appropriately subtracted out from the measurements of total fluorescence and total cell area. Thus, near the end of long acquisitions that suffered from cell crowding, the estimate for the population's mean fluorescence was in fact the average fluorescence/pixel for the remaining single-layered cells close to the perimeter of the images, which was acceptable since this was also the region where tracked cells were always selected (Figure 5.5(b)).

The experimental and data processing techniques described in this chapter comprised the development of a novel method for acquiring *in vivo* measurements of stochasticity in gene expression. These procedures were repeated many times while varying several experimental parameters. *E. coli* cells were observed as they expressed each of the constructed plasmids described above, under various temperatures, and with different concentrations of inducer (ATc) when applicable. Results from all of these experiments are presented and analyzed in the following chapter.

Chapter 6

Analysis of *In Vivo* Noise Measurements

In Chapter 5, experimental procedures were developed for acquiring measurements of real-time gene expression in living whole cells. In this chapter, noise data obtained from predefined experiments are analyzed and compared. The results show that cell behavior and gene circuit noise depend on factors such as temperature and the variant, or half-life, of the GFP reporter protein. In addition, spectral analysis of noise data verifies that negative autoregulation shifts gene circuit noise to higher frequencies, confirming the predictions of previous theoretical analysis [14]. Finally, measured noise data presented here are used to support gene circuit models and demonstrate that noise may have a probative value. That is, frequency content of noise may reveal information about enzyme kinetics and subtle condition-dependent feedback mechanisms.

6.1 Experiments and Conditions

Table 6.1 provides a list of completed experiments along with each of the varied conditions. All experiments were conducted using *E. coli* cells. Studies began with the

Table 6.1. List of experiments and conditions

Experiment Name	Plasmid Type	Inducer/Additive (conc.)	Temp (°C)	Duration (min)	Cells Tracked
AAV25	pGFP-aav	None	25	360	6
AAV30	pGFP-aav	None	30	180	7
ASV22	pGFP-asv	None	22	480	8
ASV26	pGFP-asv	None	26	420	8
ASV26+ATc	pGFP-asv	ATc (100 ng/ml)	26	240	6
ASV32	pGFP-asv	None	32	270	8
WT26	pGFP-WT	None	26	480	7
WT36	pGFP-WT	None	36	250	8
TetR21*	pTetR-GFP*	ATc (100 ng/ml)	21	300	6
TetR26*	pTetR-GFP*	ATc (100 ng/ml)	26	240	7

*Negatively autoregulated gene circuit.

pGFP-asv plasmid since it was available first. Results with GFP-asv protein, having the median half-life ($\lambda \approx 110$ min, [76]), were acquired at several different temperatures. Likewise, the short half-life GFP variant (aav) with $\lambda \approx 60$ min and the wild-type (WT) GFP with $\lambda \approx 24$ hr were also observed at various temperatures. All of the experiments listed so far incorporated unregulated gene circuits on high copy number plasmids. To investigate the effects of negative autoregulation, expression of pTetR-GFP was also studied and at different temperatures. Results with this plasmid were compared directly to pGFP-asv since they both contained the same *gfp-asv* reporter gene. As a control, cells with pGFP-asv were also observed in the presence of anhydrotetracycline (ATc), which was the inducer used to activate fluorescence in cells transformed with pTetR-GFP.

Durations of experiments also varied, and as pointed out in Chapter 3, data sets with limited samples can bias spectral density estimates. As a rule, a low bandwidth signal recorded over a long period of time should contain approximately the same amount of spectral information as a high bandwidth signal sampled for a proportionately shorter time interval. Results from Chapter 4 showed that gene circuit bandwidth was affected by the decay rate of output protein and negative autoregulation. Keeping all of this in mind, experiments with long half-life variants (pGFP-asv and pGFP-WT) at lower temperatures (slower kinetics) ran for durations that were 2-3 times longer than experiments with gene circuits anticipated to have higher bandwidths (pGFP-aav and pTetR-GFP). Results presented below in Section 6.3 justify the differences in the experimental durations found in Table 6.1.

6.2 Autocorrelations

To study *in vivo* noise behavior of genetic systems, correlation analysis was applied to measured noise data collected by the methods developed in Chapter 5. Cells from each experiment were grouped together. Adjusted noise time series, $X_m(n \cdot T_s)$, for cells (1, 2, ..., M) were found by subtracting the mean noise of the group from each individual noise series, which consisted of N time samples acquired with a five-minute sampling period, T_s . Normalized autocorrelation functions (ACFs) for individual cells were then found using the following biased algorithm [84]:

$$\Phi_i(jT_s) = \frac{\sum_{n=1}^{N-j} X_m(nT_s)X_m((n+j)T_s)}{\sum_{n=1}^N X_m^2(nT_s)}, \quad (6.1)$$

where n was the sample number (1, 2, ..., N) and j had integer values from 0 to $N-1$. A biased ACF was used because N was <100 for all experiments. The composite ACF for M cells was found using

$$\Phi_c(jT_s) = \frac{\sum_{m=1}^M \sum_{n=1}^{N-j} X_m(nT_s)X_m((n+j)T_s)}{\sum_{m=1}^M \sum_{n=1}^N X_m^2(nT_s)}, \quad (6.2)$$

which is the normalized average of the individual (unnormalized) ACFs for each cell in a group. The Matlab program used to compute $\Phi_i(\tau)$ and $\Phi_c(\tau)$ is located in Appendix H. The mean noise for a group of cells was removed from the individual time series above so that $\Phi_c(\tau) \rightarrow 0$ as $\tau \rightarrow \infty$. To illustrate this, Figure 6.1 shows individual cell and composite normalized ACFs for the ASV22 experiment conducted at room temperature. Individual and composite ACFs were calculated for tracked cells in every experiment. Below, these autocorrelations are presented to unveil some of the underlying information that is inherent to gene circuit noise.

6.2.1 Temperature Effects

Gene circuits are coupled chemical reactions and their behavior is often temperature-dependent as reaction rates are affected by changes in the kinetic energy of involved molecular species. To explore temperature effects on gene circuit noise, each of the

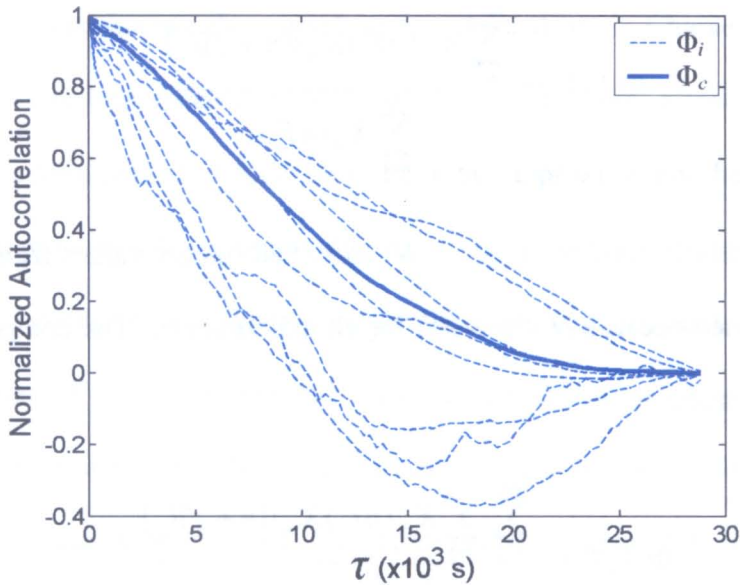


Figure 6.1. Individual cell ($\Phi_i(\tau)$) and composite ($\Phi_c(\tau)$) normalized ACFs of GFP noise for pGFP-asv at 22°C.

constructed gene circuits was studied at various temperatures (Table 6.1). Figure 6.2(a) shows measured growth curves of cell population for pGFP-asv cells at different temperatures. In general, the rate of cell growth increased at higher temperatures for all experiments. Conversely, the doubling time of cell volume decreased with increasing temperature in a linear fashion, as shown in Figure 6.2(b). Doubling times for all other experiments are covered in Section 6.3.

The individual and composite ACFs in Figure 6.3 reveal indirectly how frequency content of GFP noise was modified by the pGFP-asv gene circuit under different temperature conditions. The individual normalized ACFs, $\Phi_i(\tau)$, are shown to illustrate the variation of cells among a group and potential error in autocorrelation estimation

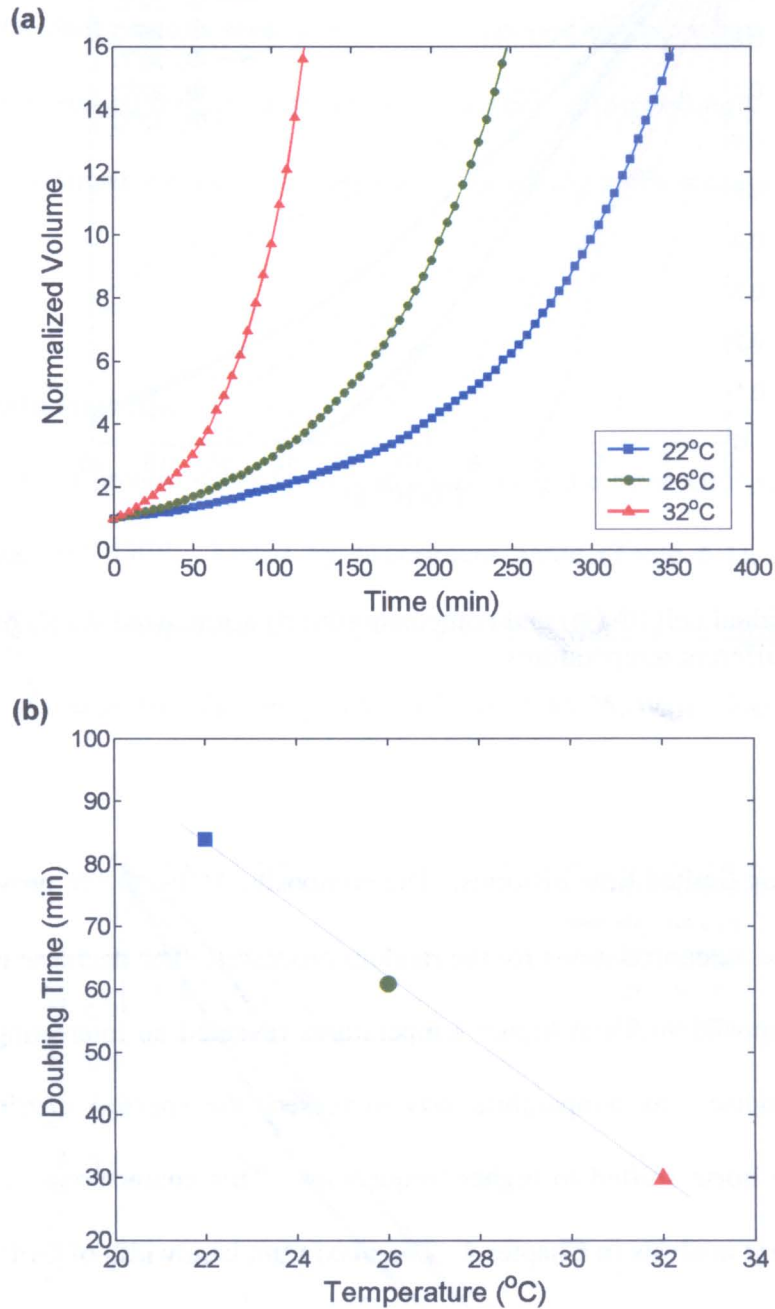


Figure 6.2. Cell growth versus temperature for pGFP-asv cells. (a) Volume of entire cell population grew exponentially over time as a function of temperature. (b) Doubling time of cell volume was inversely related to temperature.

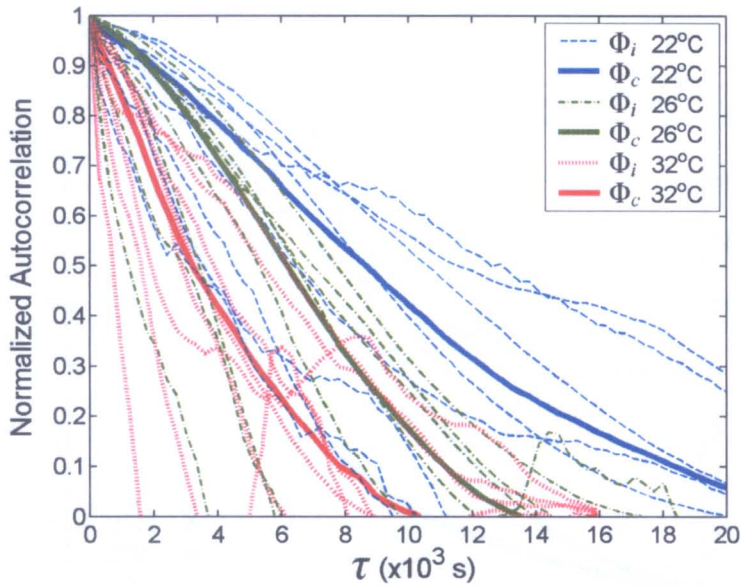


Figure 6.3. Individual cell ($\Phi_i(\tau)$) and composite ($\Phi_c(\tau)$) normalized ACFs of GFP noise for pGFP-asv at different temperatures.

caused by obtaining limited time histories. The composite ACFs, $\Phi_c(\tau)$, provide a better estimate of the true autocorrelations for the random processes. The decrease in spread, or width, of the composite ACFs at higher temperatures revealed an interesting feature of the gene circuit noise: as temperature was increased, the spectral distribution (i.e., bandwidth) of the noise shifted to higher frequencies. This characteristic is consistent with the gene circuit analysis in Chapter 4. The maximum bandwidth of GFP noise in an unregulated gene circuit should be largely determined by the rate at which the protein decays [14]. In growing cells, the protein decay rate is controlled by two mechanisms: degradation of the linked polypeptide chains and dilution of protein concentration due to cell growth. The degradation rate of protein, γ_p , is determined by its half-life

($\gamma_p = \ln(2)/\lambda$), while the dilution rate, δ , is set by the cell doubling time ($\delta = \ln(2)/T_d$). A protein's total decay rate is then $\gamma_p + \delta$. As suggested by the results in Figure 6.2, the dilution rate increased directly with temperature, and correspondingly, the ACFs of GFP noise shifted to lower values of τ (Figure 6.3), signifying an increase in the bandwidth of the circuit.

6.2.2 Protein Half-life

The effects of protein half-life were studied using the three GFP variants: GFP-aav, GFP-asv, and WT GFP. Listed by experiment name (Table 6.1), normalized ACFs acquired at similar temperatures (25-26°C) for these three gene circuits are shown in Figure 6.4. As expected, the composite ACF for AAV25, which had the shortest GFP

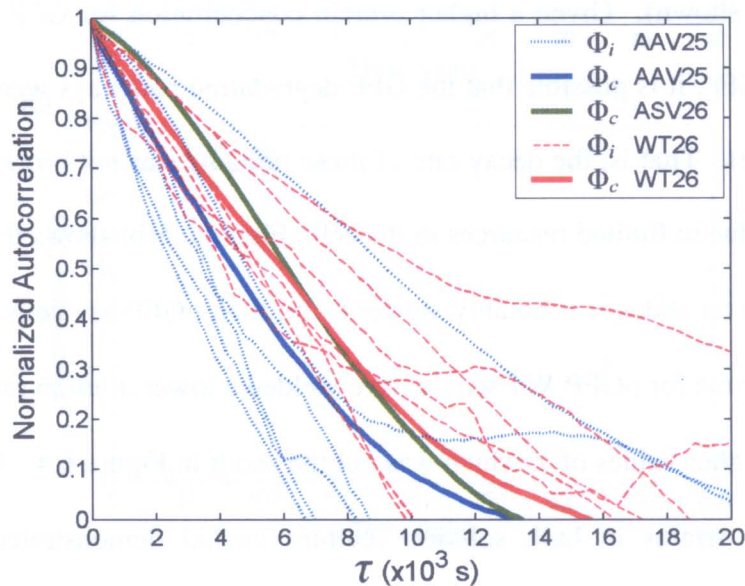


Figure 6.4. Individual cell ($\Phi_i(\tau)$) and composite ($\Phi_c(\tau)$) normalized ACFs of noise for GFP half-life variants at 25-26°C. Φ_i s for ASV26 are given above in Figure 6.3.

half-life, was located furthest to the left (smaller values of τ), indicating the increase in bandwidth associated with the fast decay of GFP-aav. The ACFs of ASV26 and WT26, on the other hand, exhibited an unexpected behavior. Instead of $\Phi_c(\tau)$ for WT26 being shifted toward higher values of τ , it seemed to follow $\Phi_c(\tau)$ for ASV26. These results motivated experiments described below to explore the existence of any global cellular responses that may have contributed to the peculiar behavior of these two gene circuits.

Differential equations describing steady-state production of protein (Table 4.1) describe how fluorescence (i.e., concentration) of GFP is directly proportional to its half-life. An unbiased comparison of pGFP-asv and pGFP-WT was performed by observing cells of each type deposited on opposite ends of the same microscope slide. As shown in Figure 6.5(a), the fluorescence intensity was greater for cells carrying pGFP-WT, as expected. Similarly, cells with pGFP-aav were the least fluorescent of the three GFP variants (data not shown). Given a higher protein concentration for GFP-asv and even more so for WT GFP, it is possible that the GFP degradation pathways were saturated in these two cell types. That is, the decay rate of these proteins was not proportional to the half-life, perhaps due to limited resources in the cell. Figure 6.5(b) shows that the growth of pGFP-WT cells on slides was notably slower ($\sim 2\times$) than pGFP-asv cells. However, a reduced doubling time for pGFP-WT would have yielded a lower dilution rate and shifted the ACF toward higher values of τ , which was not the result in Figure 6.4. A comparison of these two cell strains in bulk solution (culture media) demonstrated completely opposite behavior in cell growth. Figure 6.5(c) shows that the growth of pGFP-WT cells in solution was in fact better than that of pGFP-asv. The combination of all these results

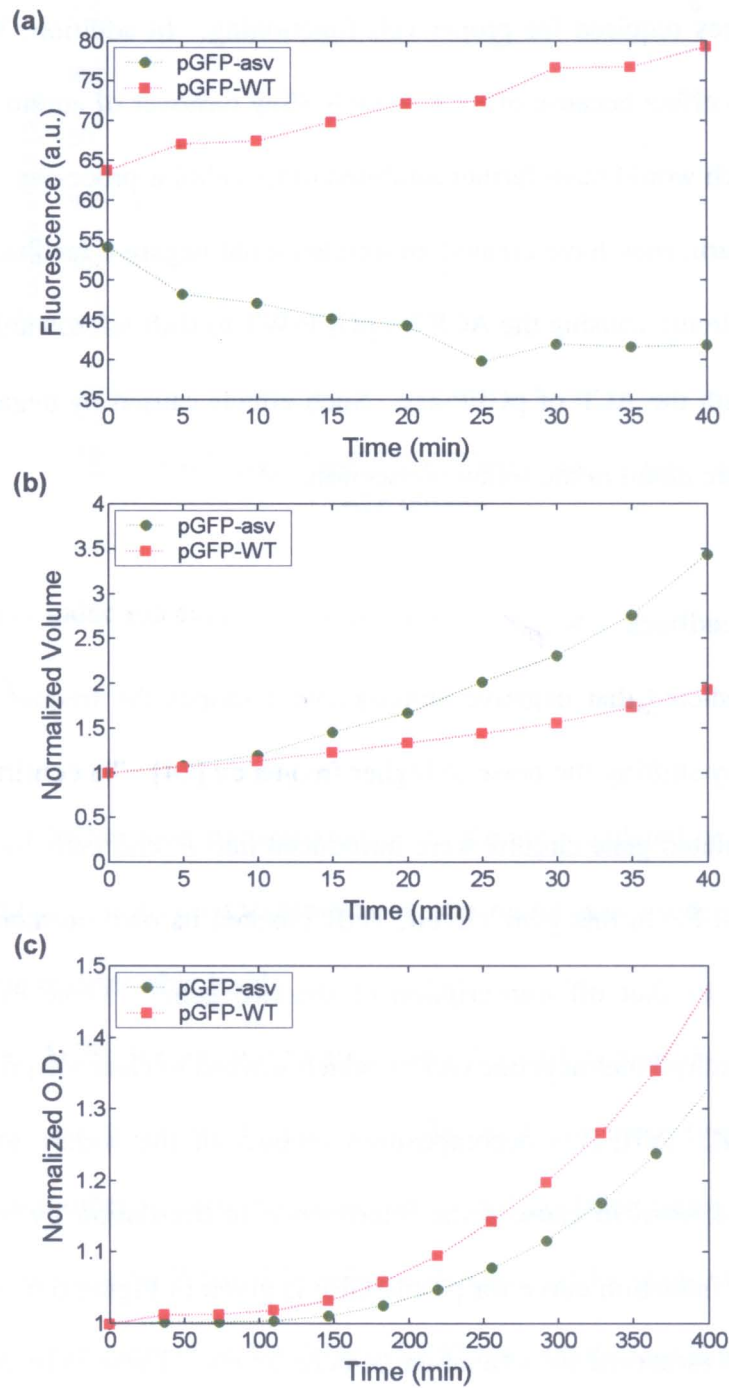


Figure 6.5. Fluorescence and growth comparison of pGFP-asv and pGFP-WT. (a) Fluorescence (i.e., GFP concentration) measured over time. (b) Growth curves of cells on a shared slide. (c) Optical density (O.D.) at 405 nm of cells growing in solution.

suggested that the microenvironment on the slides might have affected cellular behavior by limiting resources required for proper cell functioning. In addition, WT GFP may have amplified this effect because of its extremely slow turnover of amino acids through protein decay, which would have further inhibited other cellular processes, including cell growth. This, in turn, may have created an unintentional negative feedback path in the *unregulated* gene circuit, causing the ACF for pGFP-WT to shift left toward lower values of τ and line up with the ACF of pGFP-*asv*. Such effects caused by negative feedback are described in more detail in the following section.

6.2.3 Negative Feedback

It has been predicted that negative autoregulation shapes the frequency content of gene circuit noise by shifting the noise to higher frequency [14]. To confirm this theory, negatively autoregulated gene circuits were introduced into *E. coli* cells by transforming them with pTetR-GFP. In this gene circuit, TetR blocked its own promoter region and that of the *gfp-asv* to shut off transcription of the two genes. Gene expression was induced by adding anhydrotetracycline (ATc), which worked to clear the promoter region by binding up TetR. ATc is a decomposition product of the widely used antibiotic tetracycline and is known to cause some interference in translation by binding to the ribosome. An ATc induction curve for pTetR-GFP is given in Figure 6.6, describing the output fluorescence measured as a function of ATc levels. These ATc concentrations were supplements to culture media deposited on slides. For the pTetR-GFP experiments, it was desired to operate in a region with large slope (i.e., sensitivity) because this

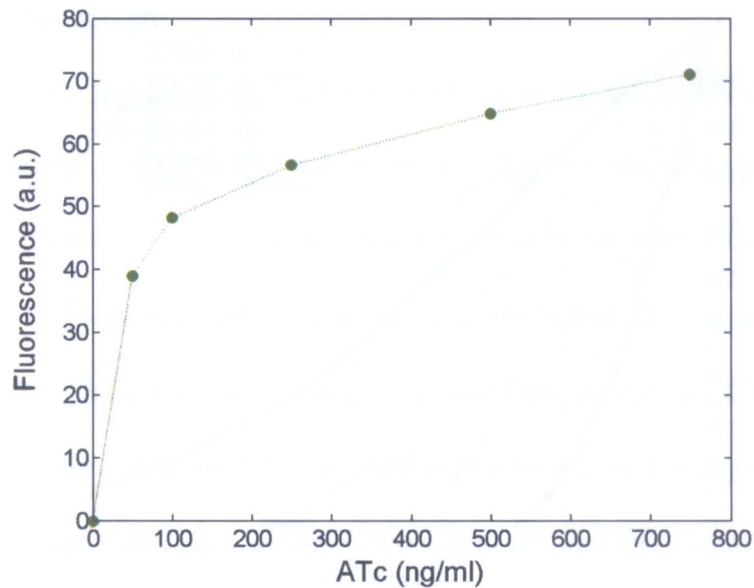


Figure 6.6. ATc induction curve for pTetR-GFP.

corresponded to higher loop transmission, T , in the autoregulated gene circuit. Though under compromise, an ATc concentration of 100 ng/ml was used in order to maintain GFP production at detectable levels.

Normalized ACFs for an unregulated (ASV22) and negatively autoregulated gene circuit (TetR26) are shown in Figure 6.7. The ACFs of these two experiments were compared to each other because they had similar cell doubling times, ranging from ~70-90 min, and the same GFP half-life. The composite ACFs showed a clear separation from each other. The spread of $\Phi_c(\tau)$ for ASV22 was nearly four times that of $\Phi_c(\tau)$ for TetR26. The significant shift toward lower τ for the ACF of the autoregulated gene circuit proved that the frequency spectrum of the GFP noise had been increased. For a

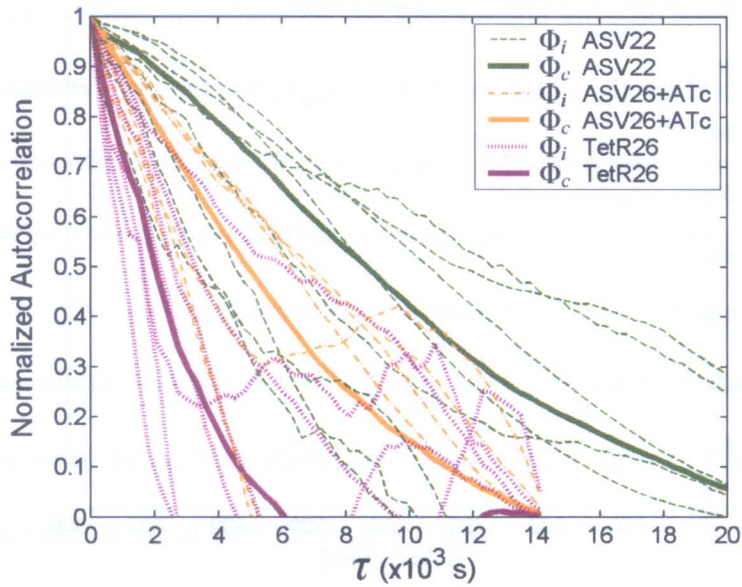


Figure 6.7. Individual cell ($\Phi_i(\tau)$) and composite ($\Phi_c(\tau)$) normalized ACFs of noise for an unregulated (ASV22), control (ASV26+ATc), and autoregulated (TetR26) gene circuit.

control experiment (ASV26+ATc), 100 ng/ml of ATc was added to pGFP-asv cells lacking the *tetR* gene. With respect to ASV22, the $\Phi_c(\tau)$ for ASV26+ATc showed a measurable shift toward lower τ . This shift in the ACF may have been due to a noise whitening effect introduced by ATc-ribosome binding events [15]. Even so, the shift of $\Phi_c(\tau)$ for TetR26 was significantly larger than the shift seen in the control experiment. Thus, the results presented here confirmed previous theoretical predictions and provided the first measured evidence that gene circuit noise is shifted to higher frequency by negative autoregulation.

6.3 Noise Bandwidth

The term noise bandwidth is used here to describe the frequency range that contains most of the spectral content of a noise signal. To visualize frequency ranges for all of the gene circuits studied in this work, the noise bandwidth of each circuit was defined as $1/\tau_{1/2}$, where $\tau_{1/2}$ was the half-correlation time, the value of τ where $\Phi_i(\tau)$ or $\Phi_c(\tau)$ decreased to 0.5. Measured noise bandwidth as a function of cell doubling time, T_d , is shown in Figure 6.8 for all of the conducted experiments. In this figure, empty symbols represent individual cells while filled symbols are derived from composite ACFs. The composite measurements provide a better estimate of the true noise bandwidth, while the individual cell measurements illustrate variance in the estimation. T_d for each cell was found using Eq. 5.1, while doubling times for composite data points were found as the average T_d of the individual tracked cells in the respective cluster. Cells were clustered by experiment, as noted in the figure legend.

Some trends in the measured data of Figure 6.8(a) illustrate spectral features for each of the ACFs discussed above in Section 6.2. For example, the three ASV data points (green) show the increase in noise bandwidth that occurred with decreasing doubling time as the temperature was increased. As expected, the noise bandwidth for the AAV points (blue) was shifted above the ASV points due to, at least in part, the shorter half-life of the GFP-aav. However, the WT data points (red) were not shifted below the ASV points, despite the longer half-life of WT GFP. As suggested earlier, the unexpected increase in the noise bandwidth of the WT data points may have been the result of global negative feedback in an unknown pathway.

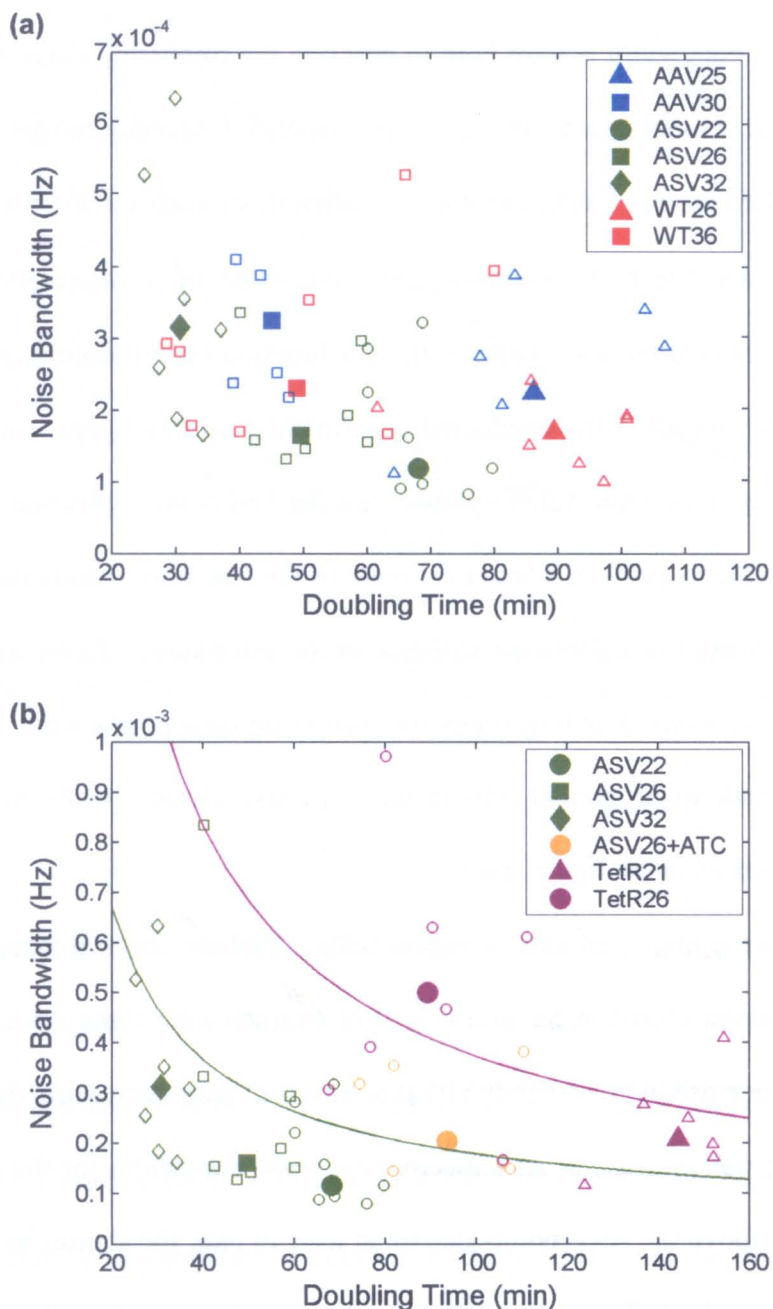


Figure 6.8. Bandwidth of gene circuit noise found from half-correlation times. (a) Noise bandwidth of pGFP variants at different temperatures. (b) Noise bandwidth of unregulated (ASV), control (ASV26+ATc), and autoregulated (TetR) gene circuits. Empty symbols represent individual cells while filled symbols are derived from composite ACFs. Trend lines in (b) are from analytical noise models in Section 6.4.

Increased noise bandwidth caused by deliberate addition of negative feedback is displayed in Figure 6.8(b) for the ASV (green) and TetR (purple) data points. At doubling times of ~ 80 min, the increase in noise bandwidth due to negative autoregulation was $\sim 4\times$. As discussed above, a slight increase in noise bandwidth did occur for the ASV26+ATc control experiment. The increase in doubling time for the ASV26+ATc and TetR experiments was a side effect of the added ATc. The trend lines in Figure 6.8(b) are derived from models presented in the next section.

6.4 Gene Circuit Models

Figure 6.9 shows simplified models for the variant pGFP and the autoregulated pTetR-GFP gene circuits. For the pGFP model in Figure 6.9(a), plasmid DNA is transcribed at rate K_m to produce mRNA molecules, which are translated at rate K_p to

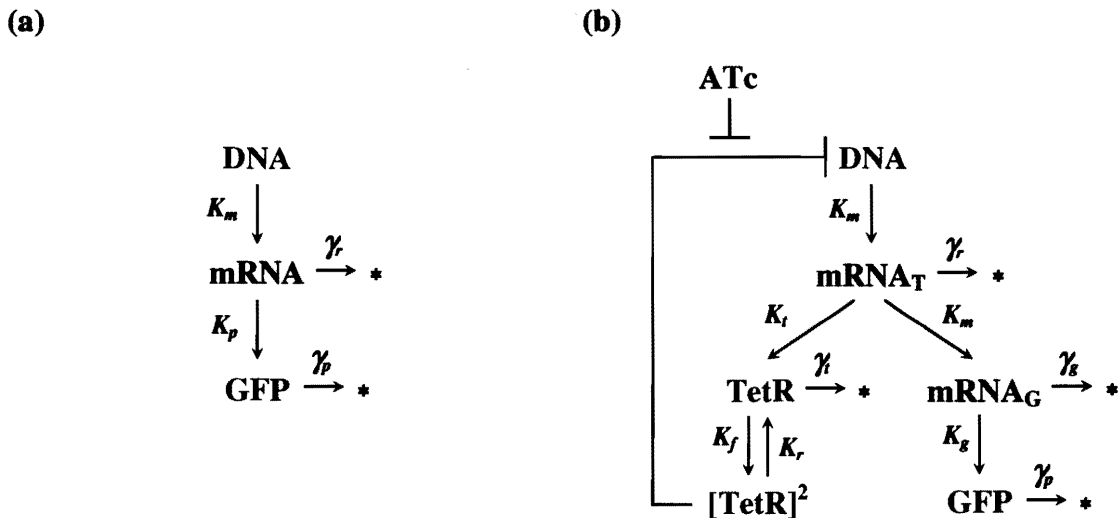


Figure 6.9. Simple models for (a) variant pGFP and (b) pTetR-GFP gene circuits.

synthesize GFP. The mRNA and GFP decay (*) at rates γ_r and γ_p , respectively. In the pTetR-GFP model, mRNA_G is not transcribed until after mRNA_T has been created. GFP is then translated from mRNA_G. The TetR is translated from mRNA_T but then forms a dimer before it actively binds to the DNA to regulate transcription [77]. ATc induces gene expression by suppressing TetR-DNA binding. The indicated species decay at their respective rates.

Recent reports by Elowitz and Rosenfeld indicate that both intrinsic and extrinsic noise sources are prevalent in gene circuits [33, 85]. Therefore, a model was developed that included high bandwidth intrinsic noise and bandwidth-limited extrinsic noise. The energy spectral density (ESD) of GFP noise for this noise model is shown in Figure 6.10(a). As indicated in the figure, the bandwidth of the extrinsic noise is set by the dilution rate, δ , which is determined by the cell doubling time [85]. Then both extrinsic and intrinsic noise sources are filtered by the dilution and protein decay rates ($\delta + \gamma_p$). An equivalent noise circuit for the noise model is shown in Figure 6.10(b). Through reverse Fourier transform of the ESD, the theoretical ACF for GFP noise is given by

$$\Phi_{theory}(\tau) = W_E \left(\frac{a^2}{a^2 - 1} e^{-\delta\tau} + \frac{1}{1 - a^2} e^{-(\delta + \gamma_p)\tau} \right) + W_I e^{-(\delta + \gamma_p)\tau}, \quad (6.3)$$

where $a = (\delta + \gamma_p) / \delta$. The first term on the right-hand side of Eq. 6.3 accounts for extrinsic noise and the last term corresponds to filtered intrinsic noise. The scalars, W_E and W_I , set the contributions from the extrinsic and intrinsic noise sources, respectively, and they must sum to unity for the normalized ACF. For the negatively autoregulated

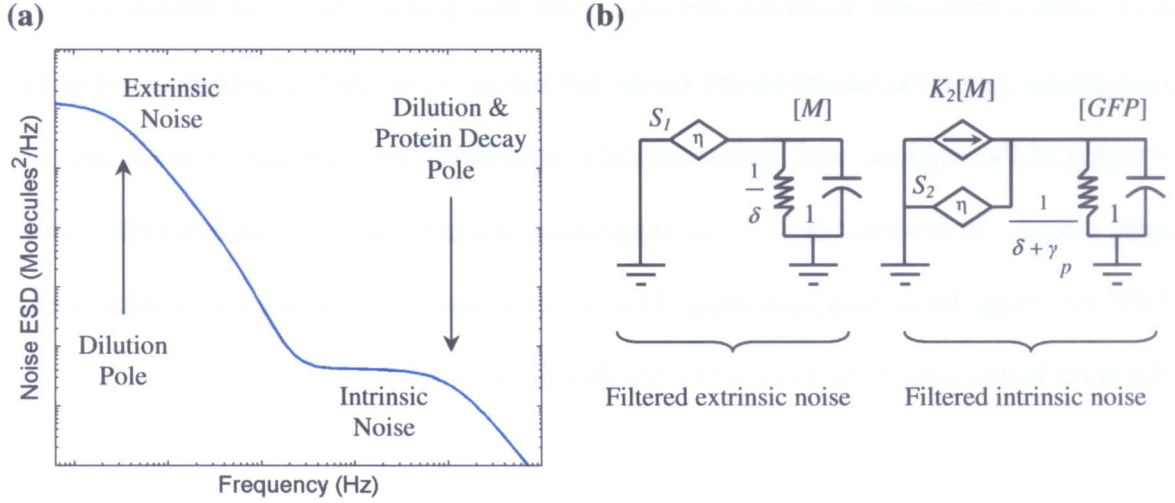


Figure 6.10. Extrinsic and intrinsic contributions to GFP noise. (a) Energy spectral density of GFP noise from the noise model. (b) Equivalent circuit for the noise model.

gene circuit, the dilution and protein decay pole shifts to higher frequency by $1+|T|$ [14].

The theoretical ACF then becomes

$$\Phi_{theory}(\tau) = W_E \left(\frac{c^2}{c^2 - 1} e^{-\delta\tau} + \frac{1}{1 - c^2} e^{-(\delta + \gamma_p)(1+|T|)\tau} \right) + W_I e^{-(\delta + \gamma_p)(1+|T|)\tau}, \quad (6.4)$$

where $c = (\delta + \gamma_p)(1+|T|) / \delta$.

Again, noise bandwidth was defined as $1/\tau_{1/2}$, where $\tau_{1/2}$ was the half-correlation time found through solution of Eq. 6.4. W_E and W_I were set equal to 0.65 and 0.35, respectively, as determined by previous estimates [85]. Using $\gamma_p = 110$ min for GFP-asv [76], theoretical noise bandwidths were calculated to derive the trend lines shown in Figure 6.8. $|T| = 4$ was used to model the strength of negative autoregulation for TetR

and yields a reasonable fit to the measured TetR data points. All of the composite ASV data points, however, fall below the model prediction. One likely explanation is that the weights of the extrinsic and intrinsic noise contributions are different for the pGFP-asv gene circuit. In Section 6.2.2, it was suggested that the protein degradation pathway for GFP-asv might have been saturated. This effect would also contribute to a decrease in the noise bandwidth of the measured ASV data points.

6.5 Estimation of Model Parameters

Sets of normalized ACFs for individual cells in each experiment exhibited rather large variance. As seen in Figure 6.11, this situation is far from ideal when attempting to fit a model to measured data points. $\Phi_{theory}(\tau)$ is shown here using the parameter values

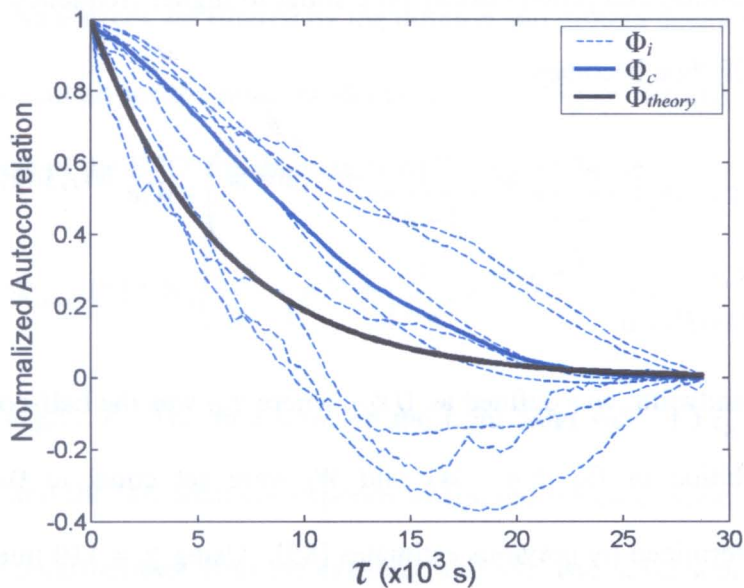


Figure 6.11. Individual cell ($\Phi_i(\tau)$), composite ($\Phi_c(\tau)$), and theoretical ($\Phi_{theory}(\tau)$) normalized ACFs of GFP noise for the ASV22 experiment.

given in Section 6.4 and with $\delta = \ln(2)/T_d$, where $T_d = 68$ min for ASV22 (Figure 6.8). The large variation in Φ_i is due to the fact that a limited number of time samples was acquired for a small population of cells. As a result, there is too much uncertainty to be able to specify model parameters with a strong level of confidence. This problem can be solved easily though; all that is needed is more samples and a lot more time.

Chapter 7

Conclusions and Future Work

Cells are complex systems made up of delicately balanced, interconnected gene circuits. Exploring and understanding the structure, function, and dynamic behavior of these circuits is a tremendous challenge. By coupling analysis and simulation with experimentation, this dissertation work has contributed substantially to the study of gene circuits. Deeper understanding of gene circuit behavior has been attained through refinement of analytical models and the development of a novel technique for mining information about underlying genetic processes from the frequency content of measured gene circuit noise.

Using available biological knowledge, more complete models were developed for unregulated and autoregulated gene circuits. These models included improvements such as transcription and translation time delays, reversible protein dimerization, and frequency-domain (FD) effects from the dynamics of reversible binding at the gene operator site. FD Langevin noise analysis applied to the models yielded relatively simple equations that uncovered relationships between model parameters and circuit behavior.

Derived transfer functions provided details of how signals were processed as they propagated through gene circuits. Despite limitations of the FD approach, careful application yielded valid results for many regions of gene circuit operation that were in excellent agreement with exact stochastic simulation, even at very low molecular populations.

Results from the FD noise analysis showed that the bandwidth of the gene circuit was determined by the protein decay rate, which was the slowest chemical reaction among the involved genetic processes. The analysis also revealed the composition of the total noise in the output protein population. Under certain conditions (i.e., moderate to high burst rates), the low-frequency noise in the protein levels was determined almost entirely by the noise of transcription and mRNA decay, as the process of translation amplified this noise. Under the same conditions, noise in synthesis and decay of protein made a negligible impact on the energy spectral distribution of the dimer protein noise. Such findings could possibly be used to improve the efficiency of stochastic simulation. That is, simulators could be made more efficient by simplifying or eliminating reactions that are known to make little or no difference in the outcome of the simulation.

The developed models also showed that it may be important to include time delays associated with transcription and translation in autoregulated gene circuits operating under certain conditions. For unregulated gene circuits, these delays simply extend the time that must elapse before active protein molecules are readily available. However, in negatively autoregulated gene circuits these delays can cause sizeable amounts of overshoot in the populations of molecular species during transitions in gene circuit

operating conditions. This is not only wasteful of limited resources but may also have toxic or lethal effects on the cell.

The data collection and processing methods developed in this work have made significant contributions to the experimental toolset that exists for studying the behavior of gene circuits. Protocols designed for laser confocal microscopy enabled observation and *in vivo* measurement of real-time gene expression in discrete cells. Software programs written in Matlab automated some of the image processing steps required for data extraction and obtained measurements of cell doubling times and mean fluorescence of cell populations.

Several opportunities exist for improving the process of obtaining fluorescence data for many cells. During many experiments, fluorescence of cells decreased over time. Although the exact mechanism is unknown, one possibility is that the environment on the microscope slide was gradually deprived of oxygen, a requirement for production of fully mature, fluorescent GFP molecules [86]. Perhaps modification to the packaging of biological samples would greatly benefit the cells. Next, correction of the microscope stage height needs to be automated so that the user does not have to be present for every acquired time sample. Finally, the tracking of individual cells through movie frames should be automated by image processing software because tracking cells manually consumes an enormous amount of time.

The experimental methods developed in this dissertation work were used to acquire the first reported measurements of frequency content in gene circuit noise. Experimental designs were guided by results of preliminary analysis. Unregulated gene circuits were

constructed with different protein decay rates to verify that gene circuit bandwidth was limited by the protein half-life. Autocorrelation functions of measured noise data were used to estimate the noise bandwidth for each gene circuit. It was found that the protein half-life did have an affect on the noise bandwidth. However, measurements revealed that bandwidth was more directly related to dilution of protein by cell growth, as cell doubling time was generally less than the protein half-life. Some disagreement between theory and measured results then motivated additional experiments to explore the cause of the unexpected increase in noise bandwidth for the wild-type (WT) GFP gene circuit. Negatively autoregulated gene circuits were also constructed, and measured frequency content of their noise provided experimental verification that noise was shifted to higher frequency by the negative autoregulation. The fact that the noise bandwidth for WT GFP was higher than expected suggested the possible existence of a condition-dependent negative feedback path in the WT gene circuit. This subtlety may have gone undetected by other traditional measurement techniques, demonstrating that noise does have probative value for studying gene circuit behavior.

This dissertation work provides a foundation for the potential development of a non-invasive technique for measuring gene circuit parameters in living cells. As proven in this research, gene circuits shape the frequency spectrum of their noise. Accurate measurements of this spectral distribution would allow confident fitting of models and the extraction of the gene circuit parameters. Eventually, experiments will get even more sophisticated: spectral analysis of input and output signals will be used to deduce transfer functions for whole genetic systems. This is all just a matter of time.

References

- [1] L. Bousse, "Whole cell biosensors," *Sens. and Actuat. B*, vol. 34, pp. 270-275, 1996.
- [2] J. W. Costerton, Z. Lewandowski, D. E. Caldwell, D. R. Korber, and H. M. Lappin-Scott, "Microbial Biofilms," *Annu. Rev. Microbiol.*, vol. 49, pp. 711-745, 1995.
- [3] H. C. Berg and D. A. Brown, "Chemotaxis in *Escherichia-coli* analyzed by 3-dimensional tracking," *Nature*, vol. 239, pp. 500, 1972.
- [4] M. Farina, D. Esquivel, and H. Lins de Barros, "Magnetic iron-sulfur crystals from a magnetotactic microorganism," *Nature*, vol. 343, pp. 256-258, 1990.
- [5] H. Kitano, "Systems biology: a brief overview," *Science*, vol. 295, pp. 1662-1664, 2002.
- [6] T. Ideker, V. Thorsson, J. A. Ranish, R. Christmas, J. Buhler, J. K. Eng, R. Bumgarner, D. R. Goodlett, R. Aebersold, and L. Hood, "Integrated genomic and proteomic analyses of a systematically perturbed metabolic network," *Science*, vol. 292, pp. 929-934, 2001.
- [7] F. J. Isaacs, J. Hasty, C. R. Cantor, and J. J. Collins, "Prediction and measurement of an autoregulatory genetic module," *Proc. Natl. Acad. Sci. USA*, vol. 100, pp. 7714-7719, 2003.
- [8] Y. Benenson, T. Paz-Elizur, R. Adar, E. Keinan, Z. Livneh, and E. Shapiro, "Programmable and autonomous computing machine made of biomolecules," *Nature*, vol. 414, pp. 430-434, 2001.

- [9] Y. Benenson, B. Gil, U. Ben-Dor, R. Adar, and E. Shapiro, "An autonomous molecular computer for logical control of gene expression," *Nature*, vol. 429, pp. 423-429, 2004.
- [10] J. M. McCollum, G. D. Peterson, C. D. Cox, and M. L. Simpson, "Accelerating gene regulatory network modeling using grid-based simulation," *Simulation*, vol. 80, pp. 231-241, 2004.
- [11] C. Sander, "Genomic medicine and the future of health care," *Science*, vol. 287, pp. 1977-1978, 2000.
- [12] H. De Jong, "Modeling and simulation of genetic regulatory systems: a literature review," *J. Comput. Biol.*, vol. 9, pp. 67-103, 2002.
- [13] D. T. Gillespie, "The chemical Langevin equation," *J. Chem. Phys.*, vol. 113, pp. 297-306, 2000.
- [14] M. L. Simpson, C. D. Cox, and G. S. Sayler, "Frequency domain analysis of noise in autoregulated gene circuits," *Proc. Natl. Acad. Sci. USA*, vol. 100, pp. 4551-4556, 2003.
- [15] C. D. Cox, G. D. Peterson, M. Allen, J. M. Lancaster, J. M. McCollum, D. Austin, L. Yan, G. S. Sayler, and M. L. Simpson, "Analysis of noise in quorum sensing," *Omics*, vol. 7, pp. 317-334, 2003.
- [16] M. L. Simpson, C. D. Cox, and G. S. Sayler, "Frequency domain chemical Langevin analysis of stochasticity in gene transcriptional regulation," *J. Theor. Biol.*, vol. 229, pp. 383-394, 2004.

- [17] H. H. McAdams and A. Arkin, "Stochastic mechanisms in gene expression," *Proc. Natl. Acad. Sci. USA*, vol. 94, pp. 814-819, 1997.
- [18] A. Arkin, J. Ross, and H. H. McAdams, "Stochastic kinetic analysis of developmental pathway bifurcation in phage lambda-infected *Escherichia coli* cells," *Genetics*, vol. 149, pp. 1633-1648, 1998.
- [19] D. T. Gillespie, "Exact stochastic simulation of coupled chemical reactions," *J. Phys. Chem.*, vol. 81, pp. 2340-2361, 1977.
- [20] M. A. Gibson and J. Bruck, "Efficient exact stochastic simulation of chemical systems with many species and many channels," *J. Phys. Chem. A*, vol. 104, pp. 1876-1889, 2000.
- [21] D. T. Gillespie, "Approximate accelerated stochastic simulation of chemically reacting systems," *J. Chem. Phys.*, vol. 115, pp. 1716-1733, 2001.
- [22] J. Hasty, D. McMillen, and J. J. Collins, "Engineered gene circuits," *Nature*, vol. 420, pp. 224-230, 2002.
- [23] T. S. Gardner, C. R. Cantor, and J. J. Collins, "Construction of a genetic toggle switch in *Escherichia coli*," *Nature*, vol. 403, pp. 339-342, 2000.
- [24] M. L. Simpson, G. S. Sayler, J. T. Fleming, and B. Applegate, "Whole-cell biocomputing," *Trends in Biotech.*, vol. 19, pp. 317-323, 2001.
- [25] M. B. Elowitz and S. Leibler, "A synthetic oscillatory network of transcriptional regulators," *Nature*, vol. 403, pp. 335-338, 2000.
- [26] M. Thattai and A. van Oudenaarden, "Intrinsic noise in gene regulatory networks," *Proc. Natl. Acad. Sci. USA*, vol. 98, pp. 8614-8619, 2001.

- [27] T. B. Kepler and T. C. Elston, "Stochasticity in transcriptional regulation: origins, consequences, and mathematical representations," *Biophys. J.*, vol. 81, pp. 3116-3136, 2001.
- [28] P. S. Swain, M. B. Elowitz, and E. D. Siggia, "Intrinsic and extrinsic contributions to stochasticity in gene expression," *Proc. Natl. Acad. Sci. USA*, vol. 99, pp. 12795-12800, 2002.
- [29] C. V. Rao, D. M. Wolf, and A. P. Arkin, "Control, exploitation and tolerance of intracellular noise," *Nature*, vol. 420, pp. 231-237, 2002.
- [30] J. Paulsson, "Summing up the noise in gene networks," *Nature*, vol. 427, pp. 415-418, 2004.
- [31] A. Becskei and L. Serrano, "Engineering stability in gene networks by autoregulation," *Nature*, vol. 405, pp. 590-593, 2000.
- [32] E. M. Ozbudak, M. Thattai, I. Kurtser, A. D. Grossman, and A. van Oudenaarden, "Regulation of noise in the expression of a single gene," *Nat. Genet.*, vol. 31, pp. 69-73, 2002.
- [33] M. B. Elowitz, A. J. Levine, E. D. Siggia, and P. S. Swain, "Stochastic gene expression in a single cell," *Science*, vol. 297, pp. 1183-1186, 2002.
- [34] W. J. Blake, M. Kaern, C. R. Cantor, and J. J. Collins, "Noise in eukaryotic gene expression," *Nature*, vol. 422, pp. 633-637, 2003.
- [35] J. M. Raser and E. K. O'Shea, "Control of stochasticity in eukaryotic gene expression," *Science*, vol. 304, pp. 1811-1814, 2004.

- [36] C. C. Guet, M. B. Elowitz, W. Hsing, and S. Leibler, "Combinatorial synthesis of genetic networks," *Science*, vol. 296, pp. 1466-1470, 2002.
- [37] D. McMillen, N. Kopell, J. Hasty, and J. J. Collins, "Synchronizing genetic relaxation oscillators by intercell signaling," *Proc. Natl. Acad. Sci. USA*, vol. 99, pp. 679-684, 2002.
- [38] J. Hasty, J. Pradines, M. Dolnik, and J. J. Collins, "Noise-based switches and amplifiers for gene expression," *Proc. Natl. Acad. Sci. USA*, vol. 97, pp. 2075-2080, 2000.
- [39] M. Samoilov, A. Arkin, and J. Ross, "Signal processing by simple chemical systems," *J. Phys. Chem. A*, vol. 106, pp. 10205-10221, 2002.
- [40] F. R. Blattner, G. Plunkett III, C. A. Bloch, N. T. Perna, V. Burland, M. Riley, J. Collado-Vides, J. D. Glasner, C. K. Rode, G. F. Mayhew, et al., "The complete genome sequence of *Escherichia coli* K-12," *Science*, vol. 277, pp. 1453-1462, 1997.
- [41] J. Yu, S. Hu, J. Wang, G. Wong, S. Li, B. Liu, Y. Deng, L. Dai, Y. Zhou, X. Zhang, et al., "A draft sequence of the rice genome (*Oryza sativa* L. ssp. *indica*)," *Science*, vol. 296, pp. 79-92, 2002.
- [42] J. C. Venter, M. D. Adams, E. W. Myers, P. W. Li, R. J. Mural, G. G. Sutton, H. O. Smith, M. Yandell, C. A. Evans, R. A. Holt, et al., "The sequence of the human genome," *Science*, vol. 291, pp. 1304-1351, 2001.
- [43] Signal Transduction Knowledge Environment, <http://stke.sciencemag.org/cm/>

- [44] T. S. Gardner, D. di Bernardo, D. Lorenz, and J. J. Collins, "Inferring genetic networks and identifying compound mode of action via expression profiling," *Science*, vol. 301, pp. 102-105, 2003.
- [45] J. Tegner, M. K. S. Yeung, J. Hasty, and J. J. Collins, "Reverse engineering gene networks: integrating genetic perturbations with dynamical modeling," *Proc. Natl. Acad. Sci. USA*, vol. 100, pp. 5944-5949, 2003.
- [46] M. L. Simpson, C. D. Cox, G. D. Peterson, and G. S. Sayler, "Engineering in the biological substrate: information processing in genetic circuits," *Proc. IEEE*, vol. 92, pp. 848-863, 2004.
- [47] N. Barkai and S. Leibler, "Robustness in simple biochemical networks," *Nature*, vol. 387, pp. 913-917, 1997.
- [48] M. E. Csete and J. C. Doyle, "Reverse engineering of biological complexity," *Science*, vol. 295, pp. 1664-1669, 2002.
- [49] J. Watson and F. Crick, "Molecular structure of nucleic acids. A structure for deoxyribose nucleic acid," *Nature*, vol. 171, pp. 737, 1953.
- [50] B. E. Funnell and G. J. Phillips, *Plasmid Biology*. Washington, D.C.: ASM Press, 2004.
- [51] R. W. Simons and M. Grunberg-Manago, *RNA Structure and Function*. Cold Spring Harbor, NY: Cold Spring Harbor Laboratory Press, 1997.
- [52] B. Hardesty and G. Kramer, *Structure, Function, and Genetics of Ribosomes*. New York, NY: Springer-Verlag, 1986.

- [53] L. Snyder and W. Champness, *Molecular Genetics of Bacteria*. Washington, D. C.: ASM Press, 1997.
- [54] J. Zyskind and D. Smith, "DNA-replication, the bacterial-cell cycle and cell-growth," *Cell*, vol. 69, pp. 5-8, 1992.
- [55] A. Herman-Antosiewicz, M. Obuchowski, and G. Wegrzyn, "A plasmid cloning vector with precisely regulatable copy number in *Escherichia coli*," *Mol. Biotech.*, vol. 17, pp. 193-199, 2001.
- [56] N. Eynard, M. Rols, V. Ganeva, B. Galutzov, N. Sabri, and J. Teissie, "Electrotransformation pathways of procaryotic and eucaryotic cells: recent developments," *Bioelectrochem. and Bioenergetics*, vol. 44, pp. 103-110, 1997.
- [57] M. Dagert and S. Ehrlich, "Prolonged incubation in calcium-chloride improves the competence of *Escherichia coli*-cells," *Gene*, vol. 6, 1979.
- [58] P. Actor, *Antibiotic Inhibition of Bacterial Cell Surface Assembly and Function*. Washington, D.C.: American Society for Microbiology, 1988.
- [59] M. Tomasz, R. Lipman, G. L. Verdine, and K. Nakanishi, "Reassignment of the guanine-binding mode of reduced mitomycin-c," *Biochemistry*, vol. 25, pp. 4337-4344, 1986.
- [60] A. V. Oppenheim, A. S. Willsky, and S. H. Nawab, *Signals and Systems*, vol. 2nd. Upper Saddle River, NJ: Prentice Hall, 1997.
- [61] R. Jaeger, *Microelectronic Circuit Design*. Boston, MA: McGraw-Hill, 1997.
- [62] J. Nilsson and S. Riedel, *Electric Circuits*, 5th ed. Reading, MA: Addison-Wesley, 1996.

- [63] K. Ogata, *Modern Control Engineering*. Englewood Cliffs, NJ: Prentice-Hall, 1970.
- [64] P. R. Gray and R. G. Meyer, *Analysis and Design of Analog Integrated Circuits*, 3rd ed. New York, NY: John Wiley & Sons, 1993.
- [65] M. J. Roberts, *Signals and Systems*. New York, NY: McGraw-Hill, 2004.
- [66] S. M. Kay, *Modern Spectral Estimation: Theory and Application*. Englewood Cliffs, NJ: Prentice Hall, 1988.
- [67] P. H. von Hippel, "An integrated model of the transcription complex in elongation, termination, and editing," *Science*, vol. 281, pp. 660-665, 1998.
- [68] R. Wagner, *Transcription regulation in prokaryotes*. Oxford: Oxford University Press, 2000.
- [69] M. Ptashne, *A Genetic Switch: Phage Lambda and Higher Organisms*, 2nd ed. Cambridge, MA: Cell Press and Blackwell Scientific Publications, 1992.
- [70] B. J. Meyer, R. Maurer, and M. Ptashne, *J. Mol. Biol.*, vol. 139, pp. 163-194, 1980.
- [71] G. N. Stephanopoulos, A. A. Aristidou, and J. Nielsen, *Metabolic Engineering*. New York, NY: Academic Press, 1998.
- [72] D. Kennell and H. Riezman, "Transcription and translation initiation frequencies of the *Escherichia coli lac* operon," *J. Mol. Biol.*, vol. 114, pp. 1-21, 1977.
- [73] N. Rosenfeld, M. B. Elowitz, and U. Alon, "Negative autoregulation speeds the response times of transcription networks," *J. Mol. Biol.*, vol. 323, pp. 785-793, 2002.

- [74] D. Thieffry, A. M. Huerta, E. Perez-Rueda, and J. Collado-Vides, "From specific gene regulation to genomic networks: a global analysis of transcriptional regulation in *Escherichia coli*," *BioEssays*, vol. 20, pp. 433-440, 1998.
- [75] Open source code for ESS is available at <http://biocomp.ece.utk.edu>
- [76] J. B. Anderson, C. Sternberg, L. K. Poulsen, S. P. Bjorn, M. Givskov, and S. Molin, "New Unstable Variants of Green Fluorescent Protein for Studies of Transient Gene Expression in Bacteria," *Appl. Envir. Microbiol.*, vol. 64, pp. 2240-2246, 1998.
- [77] R. Lutz and H. Bujard, "Independent and tight regulation of transcriptional units in *Escherichia coli* via the LacR/O, the TetR/O and AraC/I1-I2 regulatory elements," *Nucleic Acids Res.*, vol. 25, pp. 1203-1210, 1997.
- [78] J. Sambrook, E. F. Fritsch, and T. Maniatis, *Molecular Cloning, A Laboratory Manual*, 2nd ed. Cold Spring Harbor, NY: Cold Spring Harbor Laboratory Press, 1989.
- [79] C. J. R. Sheppard and T. Wilson, "The theory of the direct-view confocal microscope," *Journal of Microscopy*, vol. 124, pp. 107-117, 1981.
- [80] H. T. M. van der Voort, G. J. Brakenhoff, J. A. C. Valkenburg, and N. Nanninga, "Design and use of a computer controlled confocal microscope for biological applications," *Scanning*, vol. 7, pp. 66-78, 1985.
- [81] <http://www.ceb.utk.edu/facilities.html>
- [82] A. Periasamy, *Methods in Cellular Imaging*. New York, NY: Oxford University Press, 2001.

- [83] ImageJ is available at <http://rsb.info.nih.gov/ij/>
- [84] J. S. Bendat and A. G. Piersol, *Data: Analysis and Measurement Procedures*, 3rd ed. New York, NY: John Wiley & Sons, 2000.
- [85] N. Rosenfeld, J. W. Young, U. Alon, P. S. Swain, and M. B. Elowitz, "Gene Regulation at the Single-Cell Level," *Science*, vol. 307, pp. 1962-1965, 2005.
- [86] R. Y. Tsien, "The green fluorescent protein," *Annu. Rev. Biochem.*, vol. 67, pp. 509-544, 1998.

Appendices

Appendix A

```
% Cmog.m
%
% Complete Model of Gene (unregulated)
% This Matlab file generates ESD plots
% for an unregulated gene circuit
%
% Written by: Derek Austin

% Rate constants
Kin = 0.001; % DNA -> C
Kc = 0.1; % C -> T + DNA
KmM = 0.1; % T -> mRNA (delay for Tau-R)
M = 10; % number of steps in transcription
Tr = M/KmM % mRNA synthesis delay
Gr = 0.0058; % mRNA -> * (2 min half-life)
Kp = 0.12; % mRNA -> mC
KmC = 0.1; % mC -> mT + mRNA
KtN = 0.1; % mT -> P (delay for Tau-P)
N = 10; % number of steps in translation
Tp = N/KtN % protein synthesis delay
Gp = 0.0002; % P -> * (60 min half-life)
Kf = 0.0005; % 2P -> Di (double in ESS)
Kr = 0.6; % Di -> 2P
burst = Kp/Gr % burst rate

% Calculate average specie populations
Op = roots([Kin Kc -Kc]); % solve for mean unbound operator
ans = find(Op>0); % find positive root
Op_avg = Op(ans) % mean unbound operator
C_avg = Op_avg*Kin/Kc
T_avg = C_avg*Kc/KmM
mRNA_avg = KmM*T_avg/Gr % mean mRNA population
mC_avg = mRNA_avg*Kp/KmC
mT_avg = mC_avg*KmC/KtN
P_avg = mT_avg*KtN/Gp % mean protein population
Di_avg = P_avg^2*Kf/Kr % mean dimer population

% ESS results
Op_avg = 0.9901
mRNA_avg = 0.1666
P_avg = 99.78
Di_avg = 9.575

% Pole frequencies
Fr = Gr / (2*pi) % mRNA pole
Fp = Gp / (1+(2*P_avg*Kf)/Kr) / (2*pi) % protein pole
Fd = Kr*(1+(2*P_avg*Kf)/Kr) / (2*pi) % dimer pole
Fz = Gp / (2*pi) % dimerization zero
```

```

##### Transfer Functions #####

start = -5.2; % start frequency 10^#
endpt = 0; % end frequency 10^#
f = logspace(start, endpt, 50); % create frequency space

% From mRNA to Dimer
Ao = (2*P_avg*Kf)*Kp/(Kr*Gr*Gp);
H_4_2 = Ao./(1+i*f/Fr)./(1+i*f/Fp)./(1+i*f/Fd)...
        .*exp(-i*2*pi*(Tr+Tp).*f);

% Single-sided ESD of Protein noise due to mRNA synthesis and decay
Sr = 2*2*Kin;
So_4_2 = Sr .* H_4_2 .* conj(H_4_2);

% Transfer function from translation noise to Dimer
Ao = (2*P_avg*Kf)/(Kr*Gp);
H_4_3 = Ao./(1+i*f/Fp)./(1+i*f/Fd).*exp(-i*2*pi*Tp.*f);

% Single-sided ESD of Protein noise due to Protein synthesis and decay
Sp = 2*Kp*mRNA_avg;
So_4_3 = Sp .* H_4_3 .* conj(H_4_3);

% From Dimer noise to Dimer
Ao = 1/Kr;
H_4_3b = -Ao.*(1+i*f/Fz) ./ (1+i*f/Fp)./(1+i*f/Fd);

% Single-sided ESD of Dimer noise due to Protein synthesis
Sd = 4*Kr*Di_avg;
So_4_3b = Sd .* H_4_3b .* conj(H_4_3b);

% Sum up all noise
S_4 = So_4_2 + So_4_3 + So_4_3b;

% Plot all ESDs on one graph
figure(1)
loglog(f,So_4_2,'g:', f,So_4_3,'r:', f,So_4_3b,'m:', f, S_4,'b');
legend('S_4_2', 'S_4_3', 'S_4_3b', 'S_4');
axis([10^start 10^endpt 1e0 1e7])
xlabel('Frequency (Hz)')
ylabel('Noise ESD (Molecules^2/Hz)')

```

Appendix B

```

% Cmonag.m
%
% Complete Model of Neg. Autoregulated Gene
% This Matlab file generates Bode and ESD plots
% for an autoregulated gene circuit
%
% Written by: Derek Austin

% Rate constants
Kin = 0.003; % DNA -> C (fully induced rate)
Kc = 0.1; % C -> T + DNA
KmM = 0.1; % T -> mRNA (delay for Tau-R)
M = 10; % number of steps in transcription
Tr = M/KmM % mRNA synthesis delay
Gr = 0.0058; % mRNA -> * (2 min half-life)
Kp = 0.12; % mRNA -> mC
Kmc = 0.1; % mC -> mT + mRNA
KtN = 0.1; % mT -> P (delay for Tau-P)
N = 10; % number of steps in translation
Tp = N/KtN % protein synthesis delay
Gp = 0.0002; % P -> * (60 min half-life)
Kf = 0.0005; % 2P -> Di (double in ESS)
Kr = 0.6; % Di -> 2P
Kb = 0.02; % Di + DNA -> bDNA
Ku = 0.1; % bDNA -> Di + DNA
Kba = 0; % basal transcription rate
burst = Kp/Gr % burst rate

% Op = Ku / (Ku + Kb*Di) % avg % time operator unbound
% mRNA = Op * Kin / Gr % mRNA @ steady-state
% P = mRNA * Kp / Gp % Protein @ steady-state
% Di = P^2 * Kf / Kr % Dimer @ steady-state

% Calculate Dimer population by solving the 4 equations above
A = Kb^2;
B = 2*Ku*Kb;
C = Ku^2;
D = -(Ku*Kin*Kp/Gr/Gp)^2*Kf/Kr;
Di = [A B C D];
Di = roots(Di); % roots for dimer equation
Di = Di(find(imag(Di)==0)) % find real root
Tb = 1/Ku; % avg time operator bound
Tu = 1/(Kb*Di); % avg time operator unbound
Op_avg = Tu / (Tu + Tb) % avg % time operator unbound
free_DNA = Op_avg
C_avg = Kin*Op_avg/Kc
T_avg = Kc*C_avg/KmM
% mRNA_avg = Op_avg*Kin/Gr % mean mRNA population
mRNA_avg = KmM*T_avg/Gr

```

```

mC_avg = Kp*mRNA_avg/Kmc
mT_avg = Kmc*mC_avg/KtN
% P_avg = mRNA_avg*Kp/Gp           % mean protein population
P_avg = KtN*mT_avg/Gp
Di_avg = P_avg^2*Kf/Kr           % mean dimer population
bDNA_avg = 1 - C_avg - Op_avg

% ESS results
% Op_avg = 0.351
% mRNA_avg = 0.181;
% P_avg = 108;
% Di_avg = 10.3;

% Pole frequencies
Fr = Gr / (2*pi)                 % mRNA pole
Fp = Gp / (1+(2*P_avg*Kf)/Kr) / (2*pi) % protein pole
Fd = Kr*(1+(2*P_avg*Kf)/Kr) / (2*pi) % dimer pole
Fz = Gp / (2*pi)                 % dimerization zero
Fb = (Ku+Kb*Di_avg+Kb) / (2*pi) % operator pole

%%%%%%%%%% Transfer Functions %%%%%%%%%%%

start = -5.2;                    % start frequency 10^-6
endpt = 0;                       % end frequency 10^2
f = logspace(start, endpt, 200); % create frequency space

% From mRNA to Dimer: A_4_2(f)
Ao = (2*P_avg*Kf)*Kp/(Kr*Gr*Gp);
A = Ao./(1+i*f/Fr)./(1+i*f/Fp)./(1+i*f/Fd).*exp(-i*2*pi*(Tr+Tp).*f);

% Feedback: B_2_4(f)
Bo = Kin*Ku*Kb/(Ku+Kb*Di_avg)/(Ku+Kb*Di_avg+Kb);
B = Bo./(1+i*f/Fb);

% Loop transmission: T(f)
To = Ao*Bo
T = -A.*B;

% Calculate phase margin of loop T
ind = find(abs(T)>=1);
if length(ind)>0
    tmp = T(ind(length(ind)));
    Pmargin = angle(tmp)*180/pi
end

% Bode plot of T(f)
figure(1), semilogx(f, 20*log10(abs(T)), 'b-')
title('Bode Plot of Loop Transmission: T(f)')
hold, semilogx(f, angle(T)*180/pi, 'r:')
xlabel('Frequency (Hz)')
ylabel('Mag (dB) & Phase (Degrees)')
legend('Magnitude', 'Phase', 3)
grid on, hold off

```

```

% Transfer function from mRNA to Dimer
H_4_2 = A./(1-T);

% Bode plot of H_4_2(f)
figure(2), semilogx(f, 20*log10(abs(H_4_2)), 'b-')
title('Bode Plot of H_4_,_2(f)')
hold, semilogx(f, angle(H_4_2)*180/pi, 'r:')
xlabel('Frequency (Hz)')
ylabel('Mag (dB) & Phase (Degrees)')
legend('Magnitude', 'Phase', 3)
grid on, hold off

% Single-sided PSD of Dimer noise due to mRNA synthesis
Sn = 4*(Kba + Op_avg*(Kin-Kba));
Bn = 4*(Kin-Kba)^2*(Op_avg-Op_avg^2)/(Ku+Kb*Di_avg);
Bn = Bn./(1+(f/Fb).^2);
Sr = Sn + Bn;
So_4_2 = Sr .* H_4_2 .* conj(H_4_2);

% From Protein to Dimer: A_4_3(f)
Ao = (2*P_avg*Kf)/(Kr*Gp);
A = Ao./(1+i*f/Fp)./(1+i*f/Fd).*exp(-i*2*pi*Tp.*f);

% Transfer function from Protein to Dimer
H_4_3 = A./(1-T);

% Single-sided PSD of Dimer noise due to Protein synthesis
Sp = 4*Kp*mRNA_avg;
So_4_3 = Sp .* H_4_3 .* conj(H_4_3);

% From Dimer to Dimer: A_4_3b(f)
Ao = 1/Kr;
A = -Ao.*(1+i*f/Fz) ./ (1+i*f/Fp)./(1+i*f/Fd);

% Transfer function from Protein to Dimer
H_4_3b = A./(1-T);

% Single-sided PSD of Dimer noise due to Protein synthesis
Sd = 4*Kr*Di_avg;
So_4_3b = Sd .* H_4_3b .* conj(H_4_3b);

% Sum up all PSDs
S_4 = So_4_2 + So_4_3 + So_4_3b;

% Plot all ESDs on one graph
figure(3)
loglog(f,So_4_2,'g:', f,So_4_3,'r:', f,So_4_3b,'m:',f,S_4,'b')
axis([10^start 10^endpt 1e0 1e6])
xlabel('Frequency (Hz)')
ylabel('Noise ESD (Molecules^2/Hz)')
legend('S_4_,_2', 'S_4_,_3', 'S_4_,_3_b', 'S_4')

```

Appendix C

```
% Biopsd.m
%
% Loads input file with column vectors
% for time, mRNA, protein, and dimer.
% Asks how many data points to remove from beginning.
% Calculates steady-state mean, var, and noise figure.
% Calculates noise PSD of the three species using Pwelch algorithm
% Plots noise PSD of the three species.
%
% Written by Derek Austin

filename = input('Name of input file:', 's');
numpts = input('Number of data points in input vectors:');
remove = input('Number of data points to remove from beginning:');
winsize = input('Enter window size for PSD:');
tmp = length(filename)-3;
tmpstr = filename(1:tmp);
outfile = strcat('Biopsd_',tmpstr,'mat');
tmpstr = sprintf('Save PSD outputs to %s ? (y/n):', outfile);
savepsds = input(tmpstr, 's');

% Load data from file
range = [1 0 numpts 0];
time = dlmread(filename, '\t', range);
range = [1 1 numpts 1];
mRNA = dlmread(filename, '\t', range);
range = [1 2 numpts 2];
P = dlmread(filename, '\t', range);
range = [1 3 numpts 3];
Di = dlmread(filename, '\t', range);

% Plot time series of each species
figure(1), plot(time, mRNA);
title('mRNA Population');
xlabel('Time (s)');
ylabel('Molecules');

figure(2), plot(time, P);
title('Protein Population');
xlabel('Time (s)');
ylabel('Molecules');

figure(3), plot(time, Di);
title('Dimer Population');
xlabel('Time (s)');
ylabel('Molecules');
```

```

% Process data
Ts = time(2) - time(1);           % sampling period
Fs = 1/Ts;                       % sampling rate

% Remove transient signal
mRNA_ss = mRNA(remove+1:numpts);
mRNA_ss_mean = mean(mRNA_ss)
% Remove DC signal
mRNA_ss = mRNA_ss - mRNA_ss_mean;
mRNA_ss_var = var(mRNA_ss)
mRNA_ss_nf = mRNA_ss / mRNA_ss_mean
% [P, f] = psd(data, window_size, sample_freq, window_type, detrending)
[Psd_mRNA, freq_mRNA] = psd(mRNA_ss, winsize, Fs, '', 'none');
Psd_mRNA = 2*Ts*Psd_mRNA;
figure(4), loglog(freq_mRNA, Psd_mRNA);
title('PSD of mRNA Noise');
xlabel('Frequency (Hz)')
ylabel('Power (Molecules^2/Hz)')

% Remove transient signal
P_ss = P(remove+1:numpts);
P_ss_mean = mean(P_ss)
% Remove DC signal
P_ss = P_ss - P_ss_mean;
P_ss_var = var(P_ss)
P_ss_nf = P_ss / P_ss_mean
[Psd_P, freq_P] = psd(P_ss, winsize, Fs, '', 'none');
Psd_P = 2*Ts*Psd_P;
figure(5), loglog(freq_P, Psd_P);
title('PSD of Protein Noise');
xlabel('Frequency (Hz)')
ylabel('Power (Molecules^2/Hz)')

% Remove transient signal
Di_ss = Di(remove+1:numpts);
Di_ss_mean = mean(Di_ss)
% Remove DC signal
Di_ss = Di_ss - Di_ss_mean;
Di_ss_var = var(Di_ss)
Di_ss_nf = Di_ss / Di_ss_mean
[Psd_Di, freq_Di] = psd(Di_ss, winsize, Fs, '', 'none');
Psd_Di = 2*Ts*Psd_Di;
figure(6), loglog(freq_Di, Psd_Di);
title('PSD of Dimer Noise');
xlabel('Frequency (Hz)')
ylabel('Power (Molecules^2/Hz)')

if savepsds=='y'
    tmpstr = sprintf('save %s mRNA_ss_mean freq_mRNA Psd_mRNA...
    P_ss_mean freq_P Psd_P Di_ss_mean freq_Di Psd_Di', outfile);
    eval(tmpstr);
end

```

Appendix D

```
% Focus_all.m
%
% Loads sequence of stacks of Leica TIF images.
% Uses integral of total light to autofocus.
% Copies brightest images, which should have best focus,
% into specified directory.
%
% Written by Derek Austin

clear
numimgs = input('Enter number of images in a single time stack:');
if numimgs>99
    error('Aborted -- Too many images in stack.')
end
numstacks = input('Enter number of stacks in sequence:');
if numstacks>99
    error('Aborted -- Too many stacks.')
end
[infile, pathname] = uigetfile('*.tif','Select First Input File');
if infile==0
    error('Aborted -- No input file selected.')
end
cd(pathname)
outputdir = input('Specify output directory:', 's');

% Loop thru stacks
for j = 1:numstacks

    % Loop thru a single stack
    imgnums = 0:(numimgs-1);
    imgsums = zeros(1,numimgs);
    stop = findstr('z000',infile);
    basename = infile(1:stop);
    filelen = length(infile);
    ext = infile(filelen-8:filelen);

    for i = 1:numimgs
        infile
        input = imread(infile,'tif');
        imgsums(i) = sum(sum(input));
        % Read next image
        stmp = num2str(i);
        if i<10
            infile = strcat(basename, '00', stmp, ext);
        else
            infile = strcat(basename, '0', stmp, ext);
        end
    end
end

end
```



```

% Reset z#
infile = strcat(basename, '000', ext);

% Show plot of focus curve
plot(imgnums, imgsums, 'b')
xlabel('Image Number')
ylabel('Total Light')

% Fit to polynomial and pick max point
[P,S] = polyfit(imgnums, imgsums, 5);
Y = polyval(P,imgnums);
hold on; plot(imgnums, Y, 'g'); hold off
best = max(Y);
best = find(Y==best)-1;
if length(best)>1
    best = best(1);
end
sprintf('Best Image is #: %d', best)
pause(0.5);

% Copy best image to output directory
stmp = num2str(best);
if best<10
    outfile = strcat(basename, '00', stmp, ext);
else
    outfile = strcat(basename, '0', stmp, ext);
end
copyfile(outfile, outputdir);

% Adjust filename for next stack
if j<=10
    tmpstr = strcat('_t0', num2str(j-1));
    stop = findstr(tmpstr, infile);
    basename = infile(1:stop);
    filelen = length(infile);
    ext = infile(filelen-13:filelen);
    if j<10
        tmpstr = strcat('t0', num2str(j));
    else
        tmpstr = strcat('t', num2str(j));
    end
else
    tmpstr = strcat('_t', num2str(j-1));
    stop = findstr(tmpstr, infile);
    basename = infile(1:stop);
    filelen = length(infile);
    ext = infile(filelen-13:filelen);
    tmpstr = strcat('t', num2str(j));
end
infile = strcat(basename, tmpstr, ext);

```

end

Appendix E

```
% Estimatedts.m
%
% Estimates doubling times for cells
% Loads tab-delimited fluorescence data file for all cells
% Assumes first line of input file is labels
% Assumes first column of data is time in seconds
% Looks at Area vs Time and fits exponentials to each cell cycle
% Returns a 2-D array of doubling times per cell
% Length of output array is equal to max number of cell divisions
%
% Written by: Derek Austin

[infile, pathname] = uigetfile('*.','Select Input File');
if infile==0
    error('Aborted -- No input file selected.')
end
cd(pathname);

numcells = input('Number of cells in input file:');
numpts = input('Number of data points (N) in input vectors:');
minblocksize = input('Enter min # of points to use per curve fit
(default=5):');
if isempty(minblocksize)
    minblocksize = 5;
end

% Initialize input/output arrays
areas = zeros(numpts, numcells);
dts = zeros(12, numcells); % assumes there are <12 cell cycles

% Load data from input file
range = [1 0 numpts 0];
time = dlmread(infile, '\t', range);
for i = 1:numcells
    range = [1 i numpts i];
    areas(:,i) = dlmread(infile, '\t', range);
end

for i = 1:numcells

    % plot input data vs time
    if i==1
        figure(1), plot(time, areas(:,i),'b')
    else
        figure(gcf+1), plot(time, areas(:,i),'b')
    end
end
```

```

cellcycle = 1;
blocksize = 1;

for j = 1:numpts-1
    if (areas(j+1,i) > 0.8*areas(j,i))      % look for discontinuity
        blocksize = blocksize + 1;
    elseif blocksize >= minblocksize      % enforce min # of pts/block
        tempa = areas(j-blocksize+1:j,i);
        [a,b] = expfit(time(1:blocksize), tempa); % fit exp curve
        tempt = time(j-blocksize+1:j);
        % append exponential fit to the data plot
        hold on
        plot(tempt, exp(b)*exp(a*time(1:blocksize)), 'r')
        dts(cellcycle, i) = log(2)/a/60;      % doubling time (min)
        cellcycle = cellcycle+1;
        blocksize = 1;
    else
        blocksize = 1;                      % reset
    end
end

% analyze end of data

j = j + 1;
if (blocksize >= minblocksize)            % check for min # pts
    tempa = areas(j-blocksize+1:j,i);
    [a,b] = expfit(time(1:blocksize), tempa);
    tempt = time(j-blocksize+1:j);
    % append exponential fit to the data plot
    hold on
    plot(tempt, exp(b)*exp(a*time(1:blocksize)), 'r')
    dts(cellcycle, i) = log(2)/a/60;      % doubling time (min)
    cellcycle = cellcycle+1;
    blocksize = 1;
end

end

dts          % show doubling times of all cells in array
beep        % finished

save Doublingtimes.tab dts -ascii -tabs

% Get mean of non-zero values
[m,n] = size(dts);
dts = reshape(dts, 1, m*n);
dtsum = 0;
numpts = 0;
for i = 1:m*n
    if dts(i) ~= 0
        dtsum = dtsum + dts(i);
        numpts = numpts + 1;
    end
end
end

```

```

dtavg = dtsum/numpts

% Get std dev of non-zero values
dtvar = 0;
nzvals = zeros(1,numpts);
j = 1;
for i = 1:m*n
    if dts(i) ~= 0
        dtvar = dtvar + (dts(i) - dtavg)^2;
        nzvals(j) = dts(i);
        j = j + 1;
    end
end
dtstd = sqrt(dtvar/(numpts-1))

% Plot histogram of doubling times
figure(gcf+1), hist(nzvals)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [a,b] = expfit(x,y)
% [a,b] = expfit(x,y)
% fits data to  $e^{(ax+b)} = e^b * e^{ax}$ 

fit = polyfit(x, log(y), 1);
a = fit(1);
b = fit(2);

```

Appendix F

```
% Measure_background.m
%
% Loops thru TIF images in working directory
% Images must have filename: #.tif
% Looks at 100x100 user-defined region
% "outputs" array contains mean; std dev; min; max
%
% Written by Derek Austin

startfile = input('Enter file number for first image to analyze:');
numfiles = input('Enter number of image files to analyze:');
corner = input('Choose corner: 1=upper-left, 2=lower-left, 3=upper-
right, 4=lower-right:');

% Initialize output array
outputs = zeros(numfiles, 4);
len = 100; % length & width of area to analyze

% Loop thru images and integrate
for k = startfile:(numfiles-1+startfile)

    filename = sprintf('%d.tif', k)
    [img,map] = imread(filename, 'tiff');
    [n,m] = size(img); % image size
    if corner==1
        dd = double(img(1:len, 1:len)); % sample data
    elseif corner==2
        dd = double(img(n-len+1:n, 1:len)); % sample data
    elseif corner==3
        dd = double(img(1:len, n-len+1:n)); % sample data
    else
        dd = double(img(n-len+1:n, n-len+1:n)); % sample data
    end

    dd = reshape(dd, len*len, 1); % make column vector
    outputs((k-startfile+1), 1) = mean(dd);
    outputs((k-startfile+1), 2) = std(dd);
    outputs((k-startfile+1), 3) = min(dd);
    outputs((k-startfile+1), 4) = max(dd);

end

% done looping thru all image files

clear img; % free some memory
```

Appendix G

```
% Threshold_images.m
%
% Loops thru TIF images in working directory
% Images must have filename: #.tif
% Integrates grayscale values in each image
% Output array is "imgsums"
% Total image areas (mxn) are in array "imgareas"
%
% Integrates area considered to be only cells (using thresholds)
% Array of 'thresholds' can already be defined
% Output array is "cellareas"
%
% Written by Derek Austin

numfiles = input('Enter number of image files in directory:');
gotthresh = exist('thresholds','var');
if gotthresh~=1
    thresh = input('Enter threshold to classify cells from background
    (enter -1 to use histogram:');
    thresholds = zeros(numfiles, 1);
else
    thresh = 1; % needed for logic below
end

% Initialize output array
imgsums = zeros(numfiles, 1);
cellareas = zeros(numfiles, 1);
imgareas = zeros(numfiles, 1);

% Loop thru images and integrate

for k = 1:numfiles

    filename = sprintf('%d.tif', k)
    [img,cmap] = imread(filename, 'tiff');
    [m,n] = size(img); % get image size
    imgareas(k) = m*n;
    imgsums(k) = sum(sum(img)); % integrate image
    dd = double(img); % convert data type
    %figure(1), image(dd); % plot input image
    %colormap(cmap)
    %title('Input Image')
```

```

% automatically set threshold if user entered -1 for threshold
% this feature does not work well for fixed # of bins

if (thresh<0)
    d = reshape(dd, 1, m*n);           % change to 1-D array
    bins = 16;
    [h, centers] = hist(d, bins);      % histogram pixel values
    clear d;                          % free some memory
    der = diff(h);                    % derivative of histogram
    %figure(2), plot(der)
    %axis([0 bins -10 10])

    % Find derivative's zero-crossing
    for i = 2:(bins-2)
        if (der(i-1)*der(i+1) < 0) break
        end
    end

    thresholds(k) = centers(i);       % background-cell thresholds

elseif gotthresh~=1
    thresholds(k) = thresh;          % constant user-defined threshold
end

threshold = thresholds(k)           % user feedback on screen

% Threshold the input image and sum up pixels >= threshold
for i = 1:m
    for j = 1:n
        if (dd(i,j) >= thresholds(k))
            cellareas(k) = cellareas(k) + 1;
        end
    end
end
end

end

% done looping thru all image files

clear img;                          % free some memory
beep

```

Appendix H

```
% Bioautocorr.m
%
% Calculates autocorrelation functions (ACFs) for cells
% Asks how many cells are in tab-delimited input file
% Loads input file containing noise vectors for cells
% Assumes first line of input file is labels
% Assumes first column of data is time (not used)
% Can subtract local or global mean from each noise vector
% Outputs biased or unbiased ACF of each cell for Tau>0 only
% Saves autocorrelations to file: outputs.tab
% Note: this function requires Matlab's signal processing toolbox
%
% Written by Derek Austin

[infile, pathname] = uigetfile('*.','Select Input File');
if infile==0
    error('Aborted -- No input file selected.')
end
cd(pathname);

numcells = input('Number of cells in input file:');
numpts = input('Number of data points (N) in input vectors:');
globaldetrend =
    input('Subtract global mean from noise data? (y/n):', 's');
if globaldetrend == 'n'
    localdetrend = input('Subtract individual means from each...
                          noise data? (y/n);', 's');
else
    localdetrend = 'n';
end

flag = input('Select scaling: n=none, b=biased (*1/N), u=unbiased...
             (*1/(N-abs(lags))):', 's');
if flag == 'b'
    cflag = 'biased';
elseif flag == 'u'
    cflag = 'unbiased';
else
    cflag = 'none';
end

% Initialize input/output arrays
noisedata = zeros(numpts, numcells);
doublelength = 2*numpts-1;
outputs = zeros(doublelength, numcells);
```



```

% Load data from file
for i = 1:numcells
    range = [1 i numpts i];
    noisedata(:,i) = dlmread(infile, '\t', range);
end

% Detrending - if selected by user

% Subtract global mean from each vector
if globaldetrend == 'y'
    avgnoise = mean(mean(noisedata));
    noisedata = noisedata - avgnoise;
end

% Subtract local mean from each vector
if localdetrend == 'y'
    for i = 1:numcells
        noisedata(:,i) = noisedata(:,i) - mean(noisedata(:,i));
    end
end

% Calculate autocorrelations
for i = 1:numcells
    outputs(:,i) = xcorr(noisedata(:,i), cflag);
end

% Truncate output array for Tau>0
outputs = outputs(numpts:doublelength, :);

save outputs.tab outputs -ascii -tabs

```

Vita

Derek Austin was born in Oak Ridge, Tennessee, on May 12, 1977, as the son of Debra and Larry Austin. He graduated from Oak Ridge High School in 1995 and then entered Clemson University, where his studies focused on electronics and wireless digital communications. In 1999, he received his Bachelor of Science degree and was honored with the Samuel R. Rhodes award for top electrical engineering graduate from Clemson. As the recipient of a National Science Foundation Graduate Research Fellowship, Derek attended the University of Tennessee (UT) in Knoxville, while conducting his graduate research at Oak Ridge National Laboratory (ORNL) under the direction of Dr. Michael Simpson. He received the Master of Science degree in electrical engineering in 2001. From 2001 to 2003, Derek instructed two junior-level courses in electronic circuit design at UT. In 2005, he received the Doctor of Philosophy degree in electrical engineering from UT.

Presently, Derek is continuing systems biology research with the Molecular-scale Engineering and Nanoscale Technologies Research Group at ORNL. He also works as a consultant for CTI Molecular Imaging in Knoxville, Tennessee. His other hobbies include carpentry, restoring cars, skiing, golf, and tennis. He is a member of the IEEE, ASM, Tau Beta Pi, and Phi Kappa Phi.