

---

---

# Biomolecular Feedback Systems

---

Domitilla Del Vecchio  
MIT

Richard M. Murray  
Caltech

DRAFT v0.4a, January 1, 2011  
© California Institute of Technology  
All rights reserved.

This manuscript is for review purposes only and may not be reproduced, in whole or in part, without written consent from the authors.

---

---

# **Chapter 1**

## **Introductory Concepts**

This chapter provides a brief introduction to concepts from systems biology, tools from control theory, and approaches to modeling, analysis and design of biomolecular feedback systems. We begin with a discussion of the role of modeling, analysis and feedback in biological systems, followed by an overview of basic concepts from cell biology, focusing on the dynamics of protein production and control. This is followed by a short review of key concepts and tools from control and dynamical systems theory, intended to provide insight into the main methodology described in the text. Finally, we give a brief introduction to the field of synthetic biology, which is the primary topic of the latter half of the text.

### **1.1 Systems Biology: Modeling, Analysis and the Role of Feedback**

At a variety of levels of organization—from molecular to cellular to organismal—biology is becoming more accessible to approaches that are commonly used in engineering: mathematical modeling, systems theory, computation and abstract approaches to synthesis. Conversely, the accelerating pace of discovery in biological science is suggesting new design principles that may have important practical applications in man-made systems. This synergy at the interface of biology and engineering offers unprecedented opportunities to meet challenges in both areas. The guiding principles of feedback and control are central to many of the key questions in biological engineering and can play an enabling role in understanding the complexity of biological systems.

In this section we summarize our view on the role that modeling and analysis should (eventually) play in the study and understanding of biological systems, and discuss some of the ways in which an understanding of feedback principles in biology can help us better understand and design complex biomolecular circuits. There are a wide variety of biological phenomena that provide a rich source of examples for control, including gene regulation and signal transduction; hormonal, immunological, and cardiovascular feedback mechanisms; muscular control and locomotion; active sensing, vision, and proprioception; attention and consciousness; and population dynamics and epidemics. Each of these (and many more) provide opportunities to figure out what works, how it works and what can be done to affect it. Our focus here is at the molecular scale, but the principles and approach that we

describe can also be applied at larger time and length scales.

### **Modeling and analysis**

Over the past several decades, there have been huge advances in modeling capabilities for biological systems that have provided new insights into the complex interactions of the molecular-scale processes that implement life. Reduced-order modeling has become commonplace as a mechanism for describing and documenting experimental results and high-dimensional stochastic models can now be simulated in reasonable periods of time to explore underlying stochastic effects. Coupled with our ability to collect large amounts of data from flow cytometry, micro-array analysis, single-cell microscopy and other modern experimental techniques, our understanding of biomolecular processes is advancing at a rapid pace.

Unfortunately, although models are becoming much more common in biological studies, they are still far from playing the central role in explaining complex biological phenomena. Although there are exceptions, the predominant use of models is to “document” experimental results: a hypothesis is proposed and tested using careful experiments, and then a model is developed to match the experimental results and help demonstrate that the proposed mechanisms can lead to the observed behavior. This necessarily limits our ability to explain complex phenomena to those for which controlled experimental evidence of the desired phenomena can be obtained.

This situation is much different than what is standard practice in the physical sciences and engineering. In those disciplines, experiments are routinely used to help build models for individual components at a variety of levels of detail, and then these component-level models are interconnected to obtain a system-level model. This system-level model, carefully built to capture the appropriate level of detail for a given question or hypothesis, is used to explain, predict and systematically analyze the behaviors of a system. Because of the ways in which models are viewed, it becomes possible to prove (or invalidate) a hypothesis through analysis of the model, and the fidelity of the models is such that decisions can be made based on them. Indeed, in many areas of modern engineering—including electronics, aeronautics, robotics and chemical processing, to name a few—models play a primary role in the understanding of the underlying physics and/or chemistry, and these models are used in predictive ways to explore design tradeoffs and failure scenarios.

A key element in the successful application of modeling in engineering disciplines is the use of *reduced-order models* that capture the underlying dynamics of the system without necessarily modeling every detail of the underlying mechanisms. The generation of these reduced-order models, either directly from data or through analytical or computational methods, is critical in the effective application of modeling since modeling of the detailed mechanisms produces high fidelity

models that are too complicated to use with existing tools for analysis and design. One area in which the development of reduced order models is fairly advanced is in control theory, where input/output models such as transfer functions [1], describing functions [31], Volterra series [41] and behavioral models [57] are used to capture structured representations of dynamics at the appropriate level of fidelity for the task at hand.

While developing predictive models and corresponding analysis tools for biology is much more difficult, it is perhaps even more important that biology make use of models, particularly reduced-order models, as a central element of understanding. Biological systems are by their nature extremely complex and can behave in counter-intuitive ways. Only by capturing the many interacting aspects of the system in a formal model can we ensure that we are reasoning properly about its behavior, especially in the presence of uncertainty. To do this will require substantial effort in building models that capture the relevant dynamics at the proper scales (depending on the question being asked) as well as building an analytical framework for answering questions of biological relevance.

The good news is that a variety of new techniques, ranging from experiments to computation to theory, are enabling us to explore new approaches to modeling that attempt to address some of these challenges. In this text we focus on the use of a relevant classes of reduced-order models that can be used to capture many phenomena of biological relevance.

### **Input/output formalisms for biomolecular modeling**

A key challenge in developing models for any class of problems is the selection of an appropriate mathematical framework for the models. Among the features that we believe are important for a wide variety of biological systems are capturing the temporal response of a biomolecular system to various inputs and understanding how the underlying dynamic behavior leads to a given phenotypes. The models should reflect the subsystem structure of the underlying dynamical system to allow prediction of results, but need not necessarily be mechanistically accurate at a detailed biochemical level. We are particularly interested in those problems that include a number of molecular “subsystems” that interact with each other, and so our models should support a level of modularity (with the additional advantage of allowing multiple groups to develop detailed models for each module that can be combined to form more complex models of the interacting components). Since we are likely to be building models based on high-throughput experiments, it is also key that the models capture the measurable outputs of the systems.

For many of the systems that we are interested in, a good starting point is to use reduced-order models consisting of nonlinear differential equations, possible with some time delay. In this setting, the model of a given component  $i$  in a multi-

component system might be modeled using a differential equation of the form

$$\begin{aligned} \dot{x}^i &= A^i x^i + N^i(x^i, L_j^i y^{*j}, \theta) + B^i u^i + F^i w, \\ y^i &= C^i x^i + H^i v \quad y^{*i}(t) = y^i(t - \tau^i). \end{aligned} \quad (1.1)$$

The internal state of the subsystem is captured by the state  $x^i \in \mathbb{R}^{n_i}$ , which might capture the concentrations of various species and complexes as well as other internal variables required to describe the dynamics. The “outputs” of the system, which describe those species (or other quantities) that interact with other subsystems in the cell is captured by the variable  $y^i \in \mathbb{R}^{p_i}$ . The internal dynamics consist of a set of linear dynamics ( $A^i x^i$ ) as well as nonlinear terms that depend both on the internal state and the state of other subsystems ( $N^i(\cdot)$ ), where  $\theta$  is a set of parameters that represent the context of the system (described in more detail below). We also allow for the possibility of time delays (due to folding, transport or other processes) and write  $y^{*i}$  for the “functional” output seen by other subsystems.

The coupling between subsystems is captured using a weighted graph, whose elements are represented by the coefficients  $L_j^i$  of an interconnection matrix  $L$ . In the simplest version of the model, we simply combine different outputs from other modules in some linear combination to obtain the “input”  $L_j^i y^{*j}$  (summation over repeated indices is assumed). More general interconnections are possible, including allowing multiple outputs from different subsystems to interact in nonlinear ways (such as one often sees on combinatorial promoters in gene regulatory networks).

Finally, in addition to the internal dynamics and nonlinear coupling, we separately keep track of external inputs to the subsystem ( $B^i u^i$ ), stochastic disturbances ( $F^i w^i$ ) and measurement noise ( $H^i v^i$ ). We treat the external inputs  $u^i$  as deterministic variables (representing inducer concentrations, nutrient levels, temperature, etc) and the disturbances and noise  $w^i$  and  $v^i$  as random processes (representing extrinsic and intrinsic stochasticity). If desired, the mappings from the various inputs to the states and outputs, represented by the matrices  $B$ ,  $F$  and  $H$  can also depend on the system state  $x$  (resulting in additional nonlinearities).

This particular structure is useful because it captures a large number of modeling frameworks in a single formalism. In particular, mass action kinetics and chemical reaction networks can be represented by equating the stoichiometry matrix with the interconnection matrix  $L$  and using the nonlinear terms to capture the fluxes, with  $\theta$  representing the rate constants. We can also represent typical reduced-order models for transcriptional regulatory networks by letting the nonlinear functions  $N^i$  represent various types of Hill functions and including the effects of mRNA/protein production, degradation and dilution through the linear dynamics. These two classes of systems can also be combined, allowing a very expressive set of dynamics that is capable of capturing many relevant phenomena of interest in molecular biology.

Figure 1.1 shows a graphical representation of this structure applied to a set of  $M$  subsystems, where for simplicity, we omit the stochastic disturbances and mea-

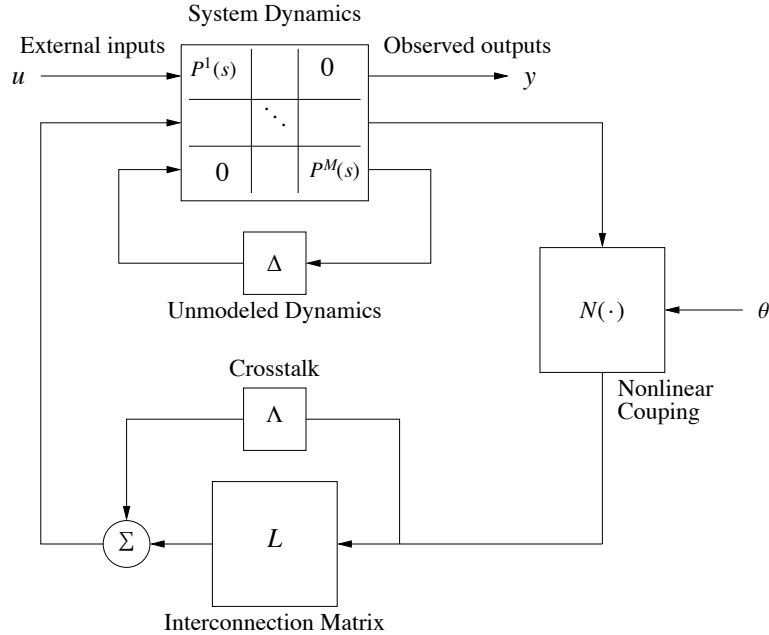


Figure 1.1: Modeling framework. The dynamics consist of a set of linear dynamics, represented by the multi-input, multi-output transfer function  $P(s)$ , a static nonlinear map  $N$  and an interconnection matrix  $L$ . Uncertainty is represented as unmodeled dynamics  $\Delta$ , crosstalk  $\Lambda$  and system context  $\theta$ . The inputs and outputs to the system are denoted by  $u$  and  $y$ .

surement noise. The linear dynamics of the system are captured via the frequency response (represented in the diagram by its Laplace transform,  $P(s)$ ). The interconnection matrix  $L$  is a matrix that takes outputs from the individual subsystems as outputs and provides linear combinations of these variables as potential inputs to the nonlinear maps represented by  $N$ . This graphical representation makes more evident the role of feedback through the interconnection matrix  $L$ .

In addition to the nominal dynamics described in equation (1.1), two other features are present in Figure 1.1. The first is the uncertainty operator  $\Delta$ , attached to the linear dynamics block. This operator represents both parametric uncertainty in the dynamics as well as unmodeled dynamics that have known (timescale dependent) bounds. Tools for understanding this class of uncertainty are available for both linear and nonlinear control systems and allow stability and performance analyses in the presence of uncertainty. A similar term  $\Lambda$  is included in the interconnection matrix and represents “crosstalk” between subsystems. While existing tools in distributed control systems do not formally handle crosstalk, we believe that it will be important to capture its effects and that it will be possible to use tools similar to those developed in control theory to analyze them.

One of the appealing features of this particular structure is that variants of it

are well studied and characterized in the control and dynamical systems literature. For example, the effect of the nonlinearities can be studied using the method of harmonic balance [44] or the related technique of describing functions (see Section 3.4). Describing function analysis allows prediction of stability boundaries and the onset of limit cycles, as well as some characterization of robustness. Similarly, in the absence of the nonlinearities and with simplifying assumptions on the linear dynamics, the effect of the interconnection topology can be captured by investigating the location of the eigenvalues of the graph Laplacian  $L$  [25].

Despite being a well-studied class of systems, there are still many open questions with this framework, especially in the context of biomolecular systems. For example, a rigorous theory of the effects of crosstalk, the role of context on the nonlinear elements, and combining the effects of interconnection, uncertainty and nonlinearity is just emerging. Adding stochastic effects, either through the disturbance and noise terms, initial conditions or in a more fundamental way, is also largely unexplored. And the critical need for methods for performing model reduction in a way that respects of the structure of the subsystems has only recently begun to be explored. Nonetheless, many of these research directions are being pursued and we attempt to provide some insights in this text into the underlying techniques that are available.

### Dynamic behavior and phenotype

One of the key needs in developing a more systematic approach to the use of models in biology is to become more rigorous about the various behaviors that are important for biological systems. One of the key concepts that needs to be formalized is the notion of “phenotype”. This term is often associated with the existence of an equilibrium point in a reduced-order model for a system, but clearly more complex (non-equilibrium) behaviors can occur and the “phenotypic response” of a system to an input may not be well-modeled by a steady operating condition. Even more problematic is determining which regulatory structures are “active” in a given phenotype (versus those for which there is a regulatory pathway that is saturated and hence not active).

In the context of the modeling framework described in equation (1.1) and Figure 1.1, it is possible to consider a working definition of phenotype in terms of the patterns of the dynamics that are present. In the simplest case, consisting of operation near equilibrium points, we can look at the effective gain of the different nonlinearities as a measure of which regulatory pathways are “active” in a given state. Consider, for example, labeling each nonlinearity in a system as being either *on*, *off* or *active*. A nonlinearity that is on or off represents one in which changes of the input produce very small deviations in the output, such as those that occur at very high or low concentrations in interactions modeled by a Hill function. An active nonlinearity is one in which there is a proportional response to changes in the

input, with the slope of the nonlinearity giving the effective gain. In this setting, the phenotype of the system would consist of both a description of the nominal concentrations of the measurable species ( $y$ ) as well as the state of each nonlinearity (on, off, active).

For more complex phenotypes, where the subsystems are not at a steady operating point, one can consider the temporal patterns that are exhibited at various points in Figure 1.1. This could correspond to traditional modal patterns such as those that are obtained via either principle component analysis or balanced truncation (the latter being a generalization of the former), or temporal patterns of regulation represented in the nonlinearities. Extending these ideas to consider changes in context and changes in input combinations is harder still, but the structure of the proposed representation presents several starting points for exploration.

Additional types of analysis that can be applied to systems of this form include sensitivity analysis (dependence of solution properties on selected parameters), uncertainty analysis (impact of disturbances, unknown parameters and unmodeled dynamics), bifurcation analysis (changes in phenotype as a function of input levels, context or parameters) and probabilistic analysis (distributions of states as a function of distributions of parameters, initial conditions or inputs). In each of these cases, there is a need to extend existing tools to exploit the particular structure of the problems we consider, as well as modify the techniques to provide relevance to biological questions.

## **Stochastic behavior**

### **The role of feedback**

One may view life in a cell as a huge “wireless” network of interactions among proteins, DNA, and smaller molecules involved in signaling and energy transfer. As a large system, the external inputs to a cell include physical signals (UV radiation, temperature) as well as chemical signals (drugs, hormones, nutrients). Its outputs include chemicals that affect other cells. Each cell can be thought of, in turn, as composed of a large number of subsystems involved in cell growth, maintenance, division and death. A typical diagram describing this complex set of interactions is shown in Figure 1.2.

The study of cell networks leads to the formulation of a large number of questions, some of which we have already alluded to above. For example, what is special about the information-processing capabilities, or input/output behaviors, of such biological networks? What “modules” appear repeatedly in cellular signaling cascades, and what are their system-theoretic properties? Inverse or “reverse engineering” issues include the estimation of system parameters (such as reaction constants) as well as the estimation of state variables (concentration of protein, RNA, and other chemical substances) from input/output experiments.

One can also attempt to better understand the temporal properties of the various

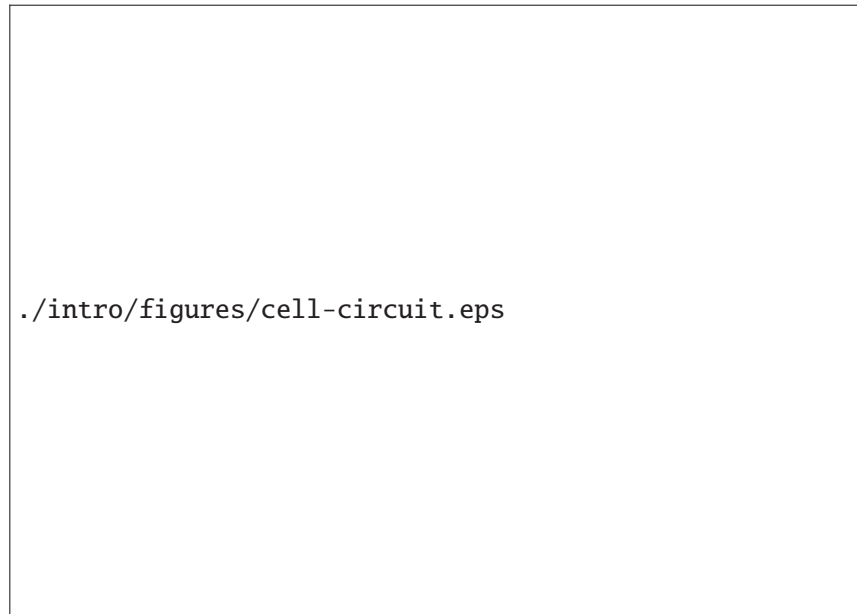


Figure 1.2: The wiring diagram of the growth signaling circuitry of the mammalian cell [34].

cascades and feedback loops that appear in cellular signaling networks. Dynamical properties such as stability and existence of oscillations in such networks are of interest, and techniques from control theory such as the calculation of robustness margins will likely play a central role in the future. At a more speculative (but increasingly realistic) level, one wishes to study the possibility of using control strategies (both open and closed loop) for therapeutic purposes, such as drug dosage scheduling.

From a theoretical perspective, feedback serves to minimize uncertainty and increase accuracy in the presence of noise. The cellular environment is extremely noisy in many ways, while at the same time variations in levels of certain chemicals (such as transcriptional regulators) may be lethal to the cell. Feedback loops are omnipresent in the cell and help regulate the appropriate variations. It is estimated, for example, that in *E. coli* about 40% of transcription factors self-regulate. One may ask whether the role of these feedback loops is indeed that of reducing variability, as expected from principles of feedback theory. Recent work tested this hypothesis in the context of tetracycline repressor protein (TetR) [11]. An experiment was designed in which feedback loops in TetR production were modified by genetic engineering techniques, and the increase in variability of gene expression was correlated with lower feedback “gains,” verifying the role of feedback in reducing the effects of uncertainty. Modern experimental techniques will afford the opportunity for testing experimentally (and quantitatively) other theoretical predic-

tions, and this may be expected to be an active area of study at the intersection of control theory and molecular biology.

Another illustration of the interface between feedback theory and modern molecular biology is provided by recent work on chemotaxis in bacterial motion. *E. coli* moves, propelled by flagella, in response to gradients of chemical attractants or repellents, performing two basic types of motions: *tumbles* (erratic turns, with little net displacement) and *runs*. In this process, *E. coli* carries out a stochastic gradient search strategy: when sensing increased concentrations it stops tumbling (and keeps running), but when it detects low gradients it resumes tumbling motions (one might say that the bacterium goes into “search mode”).

The chemotactic signaling system, which detects chemicals and directs motor actions, behaves roughly as follows: after a transient nonzero signal (“stop tumbling, run toward food”), issued in response to a change in concentration, the system adapts and its signal to the motor system converges to zero (“OK, tumble”). This adaptation happens for any constant nutrient level, even over large ranges of scale and system parameters, and may be interpreted as robust (structurally stable) rejection of constant disturbances. The internal model principle of control theory implies (under appropriate technical conditions) that there must be an embedded integral controller whenever robust constant disturbance rejection is achieved. Recent models and experiments succeeded in finding, indeed, this embedded structure [10, 73].

This is only one of the many possible uses of control theoretic knowledge in reverse engineering of cellular behavior. Some of the deepest parts of the theory concern the necessary existence of embedded control structures, and in this manner one may expect the theory to suggest appropriate mechanisms and validation experiments for them.

## 1.2 Dynamics and Control in the Cell

The molecular processes inside a cell determine its behavior and are responsible for metabolizing nutrients, generating motion, enabling procreation and carrying out the other functions of the organism. In multi-cellular organisms, different types of cells work together to enable more complex functions. In this section we briefly describe the role of dynamics and control within a cell and discuss the basic processes that govern its behavior and its interactions with its environment (including other cells). We assume knowledge of the basics of cell biology at the level provided in Appendix A; a much more detailed introduction to the biology of the cell and some of the processes described here can be found in standard textbooks on cell biology such as Alberts *et al.* [2] or Phillips *et al.* [56]. (Readers who are familiar with the material at the level described in these latter references can skip this section without any loss of continuity.)

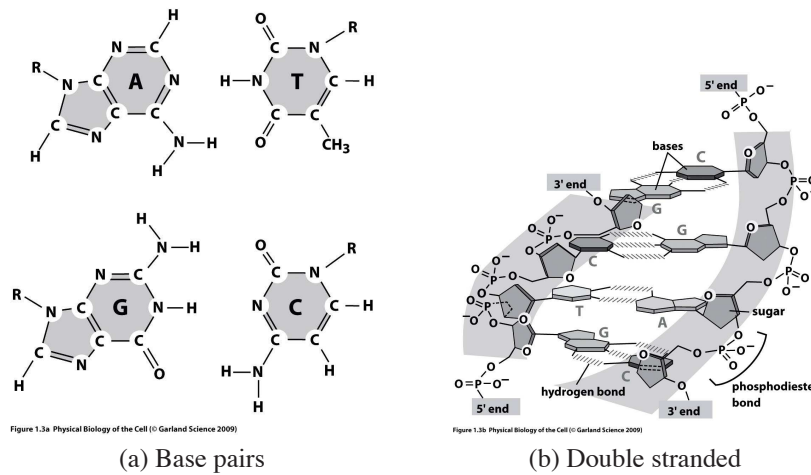


Figure 1.3: Molecular structure of DNA. (a) Individual bases (nucleotides) that make up DNA: adenine (A), cytosine (C), guanine (G) and thymine (T). (b) Double stranded DNA formed from individual nucleotides, with A binding to T and C binding to G. Each strand contains a 5' and 3' end, determined by the locations of the carbons where the next nucleotide binds. Figure from Phillips, Kondev and Theriot [56]; used with permission of Garland Science.

### The central dogma: production of proteins

The genetic material inside a cell, encoded in its DNA, governs the response of a cell to various conditions. DNA is organized into collections of genes, with each gene encoding a corresponding protein that performs a set of functions in the cell. The activation and repression of genes are determined through a series of complex interactions that give rise to a remarkable set of circuits that perform the functions required for life, ranging from basic metabolism to locomotion to procreation. Genetic circuits that occur in nature are robust to external disturbances and can function in a variety of conditions. To understand how these processes occur (and some of the dynamics that govern their behavior), it will be useful to present a relatively detailed description of the underlying biochemistry involved in the production of proteins.

DNA is a double stranded molecule with the “direction” of each strand specified by looking at the geometry of the sugars that make up its backbone (see Figure 1.3). The complementary strands of DNA are composed of a sequence of nucleotides that consist of a sugar molecule (deoxyribose) bound to one of 4 bases: adenine (A), cytosine (C), guanine (G) and thymine (T). The coding strand (by convention the top row of a DNA sequence when it is written in text form) is specified from the 5' end of the DNA to the 3' end of the DNA. (As described briefly in Appendix A, 5' and 3' refer to carbon locations on the deoxyribose backbone that are involved in linking together the nucleotides that make up DNA.) The DNA that encodes

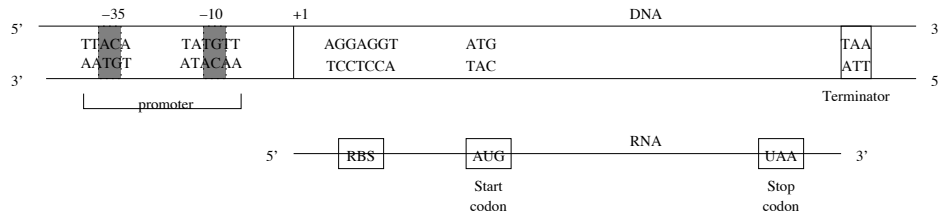


Figure 1.4: Geometric structure of DNA. The layout of the DNA is shown at the top. RNA polymerase binds to the promoter region of the DNA and transcribes the DNA starting at the +1 side and continuing to the termination site.

proteins consists of a promoter region, regulator regions (described in more detail below), a coding region and a termination region (see Figure 1.4).

RNA polymerase enzymes are present in the nucleus (for eukaryotes) or cytoplasm (for prokaryotes) and must localize and bind to the promoter region of the DNA template. Once bound, the RNA polymerase “opens” the double stranded DNA to expose the nucleotides that make up the sequence, as shown in Figure 1.5. This reversible reaction, called *isomerization*, is said to transform the RNA polymerase and DNA from a *closed complex* to an *open complex*. After the open complex is formed, RNA polymerase begins to travel down the DNA strand and constructs an mRNA sequence that matches the 5' to 3' sequence of the DNA to which it is bound. By convention, we number the first base pair that is transcribed as ‘+1’ and the base pair prior to that (which is not transcribed) is labeled as ‘-1’. The promoter region is often shown with the -10 and -35 regions indicated, since these regions contain the nucleotide sequences to which the RNA polymerase enzyme binds (the locations vary in different cell types, but these two numbers are typically used).

The RNA strand that is produced by RNA polymerase is also a sequence of nucleotides with a sugar backbone. The sugar for RNA is ribose instead of deoxyribose and mRNA typically exists as a single stranded molecule. Another difference is that the base thymine (T) is replaced by uracil (U) in RNA sequences. RNA polymerase produces RNA one base pair at a time, as it moves from in the 5' to 3' direction along the DNA coding strand. RNA polymerase stops transcribing DNA when it reaches a *termination region* (or *terminator*) on the DNA. This termination region consists of a sequence that causes the RNA polymerase to unbind from the DNA. The sequence is not conserved across species and in many cells the termination sequence is sometimes “leaky”, so that transcription will occasionally occur across the terminator (we will see examples of this in the  $\lambda$  phage circuitry described in Chapter 5).

Once the mRNA is produced, it must be translated into a protein. This process is slightly different in prokaryotes and eukaryotes. In prokaryotes, there is a region of the mRNA in which the ribosome (a molecular complex consisting of of both



Figure 1.5: Production of messenger RNA from DNA. RNA polymerase, along with other accessory factors, binds to the promoter region of the DNA and then “opens” the DNA to begin transcription (initiation). As RNA polymerase moves down the DNA, producing an RNA transcript (elongation), which is later translated into a protein. The process ends when the RNA polymerase reaches the terminator (termination). Reproduced from Courey [16]; permission pending.

proteins and RNA) binds. This region, called the *ribosome binding site (RBS)*, has some variability between different cell species and between different genes in a given cell. The Shine-Delgarno sequence, AGGAGG, is the consensus sequence for the RBS. (A consensus sequence is a pattern of nucleotides that implements a given function across multiple organisms; it is not exactly conserved, so some variations in the sequence will be present from one organism to another.)

In eukaryotes, the RNA must undergo several additional steps before it is translated. The RNA sequence that has been created by RNA polymerase consists of *introns* that must be spliced out of the RNA (by a molecular complex called the spliceosome), leaving only the *exons*, which contain the coding sequence for the protein. The term “*pre-mRNA*” is often used to distinguish between the raw tran-

script and the spliced mRNA sequence, which is called “*mature RNA*”. In addition to splicing, the mRNA is also modified to contain a *poly(A)* (polyadenine) *tail*, consisting of a long sequence of adenine (A) nucleotides on the 3’ end of the mRNA. This processed sequence is then transported out of the nucleus into the cytoplasm, where the ribosomes can bind to it.

Unlike prokaryotes, eukaryotes do not have a well defined ribosome binding sequence and hence the process of the binding of the ribosome to the mRNA is more complicated. The *Kozak sequence* A/GCCACCAAUGG is the rough equivalent of the ribosome binding site, where the underlined AUG is the start codon (described below). However, mRNA lacking the Kozak sequence can also be translated.

Once the ribosome is bound to the mRNA, it begins the process of translation. Proteins consist of a sequence of amino acids, with each amino acid specified by a codon that is used by the ribosome in the process of translation. Each codon consists of three base pairs and corresponds to one of the 20 amino acids or a “stop” codon. The genetic code mapping between codons and amino acids is shown in Table A.1. The ribosome translates each codon into the corresponding amino acid using transfer RNA (tRNA) to integrate the appropriate amino acid (which binds to the tRNA) into the polypeptide chain, as shown in Figure 1.6. The start codon (AUG) specifies the location at which translation begins, as well as coding for the amino acid methionine (a modified form is used in prokaryotes). All subsequent codons are translated by the ribosome into the corresponding amino acid until it reaches one of the stop codons (typically UAA, UAG and UGA).

The sequence of amino acids produced by the ribosome is a polypeptide chain that folds on itself to form a protein. The process of folding is complicated and involves a variety of chemical interactions that are not completely understood. Additional post-translational processing of the protein can also occur at this stage, until a folded and functional protein is produced. It is this molecule that is able to bind to other species in the cell and perform the chemical reactions that underly the behavior of the organism.

Each of the processes involved in transcription, translation and folding of the protein takes time and affects the dynamics of the cell. Table 1.1 shows the rates of some of the key processes involved in the production of proteins. It is important to note that each of these steps is highly stochastic, with molecules binding together based on some propensity that depends on the binding energy but also the other molecules present in the cell. In addition, although we have described everything as a sequential process, each of the steps of transcription, translation and folding are happening simultaneously. In fact, there can be multiple RNA polymerases that are bound to the DNA, each producing a transcript. In prokaryotes, as soon as the ribosome binding site has been transcribed, the ribosome can bind and begin translation. It is also possible to have multiple ribosomes bound to a single piece of mRNA. Hence the overall process can be extremely stochastic and asynchronous.

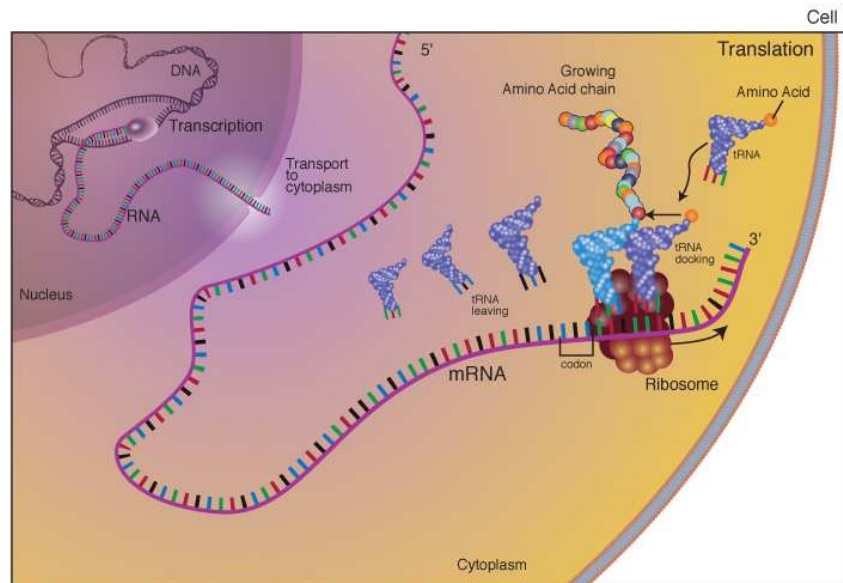


Figure 1.6: Translation is the process of translating the sequence of a messenger RNA (mRNA) molecule to a sequence of amino acids during protein synthesis. The genetic code describes the relationship between the sequence of base pairs in a gene and the corresponding amino acid sequence that it encodes. In the cell cytoplasm, the ribosome reads the sequence of the mRNA in groups of three bases to assemble the protein. Figure and caption courtesy the National Human Genome Research Institute.

### Transcriptional regulation of protein production

There are a variety of mechanisms in the cell to regulate the production of proteins. These regulatory mechanisms can occur at various points in the overall process that produces the protein. *Transcriptional regulation* refers to regulatory mechanisms that control whether or not a gene is transcribed.

Table 1.1: Rates of core processes involved in the creation of proteins from DNA in *E. coli*.

Process	Characteristic rate	Source
mRNA production	10–30 bp/sec	Vogel and Jensen
Protein production	10–30 aa/sec	PKT08
Protein folding	???	
mRNA half life	~ 100 sec	YM03
Cell division time	~ 3000 sec	???
Protein half life	~ $5 \times 10^4$ sec	YM03
Protein diffusion along DNA	up to $10^4$ bp/sec	

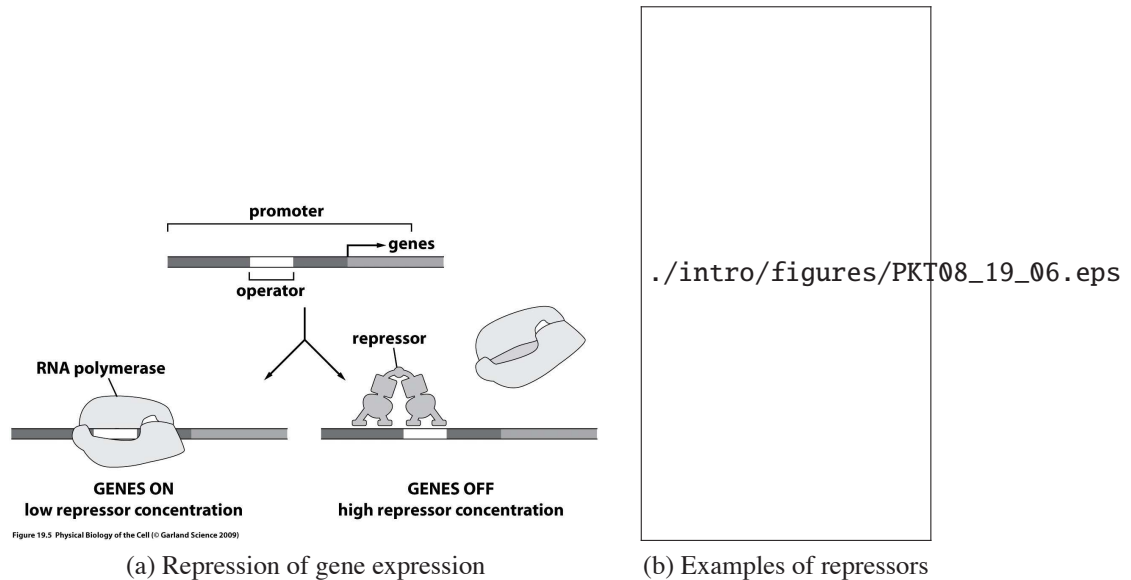


Figure 1.7: Repression of gene expression. Figure from Phillips, Kondev and Theriot [56]; used with permission of Garland Science.

The simplest forms of transcriptional regulation are repression and activation, which are controlled through *transcription factors*. In the case of repression, the presence of a transcription factor (often a protein that binds near the promoter) turns off the transcription of the gene and this type of regulation is often called negative regulation or “down regulation”. In the case of activation (or positive regulation), transcription is enhanced when an activator protein binds to the promoter site (facilitating binding of the RNA polymerase).

A common mechanism for repression is that a protein binds to a region of DNA near the promoter and blocks RNA polymerase from binding. The region of DNA in which the repressor protein binds is called an *operator region* (see Figure 1.7a). If the operator region overlaps the promoter, then the presence of a protein at the promoter “blocks” the DNA at that location and transcription cannot initiate, as illustrated in Figure 1.7a. Repressor proteins often bind to DNA as dimers or pairs of dimers (effectively tetramers). Figure 1.7b shows some examples of repressors bound to DNA.

A related mechanism for repression is *DNA looping*. In this setting, two repressor complexes (often dimers) bind in different locations on the DNA and then bind to each other. This can create a loop in the DNA and block the ability of RNA polymerase to bind to the promoter, thus inhibiting transcription. Figure 1.8 shows an example of this type of repression, in the *lac* operon. (An *operon* is a set of genes that is under control of a single promoter; this is discussed in more detail below.)

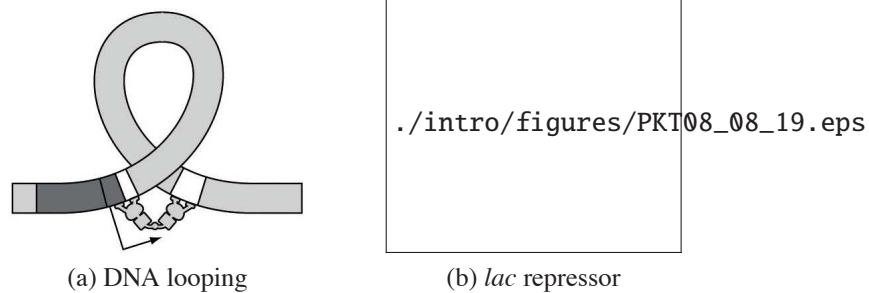


Figure 1.8: Repression via DNA looping. Figure from Phillips, Kondev and Theriot [56]; used with permission of Garland Science.

A feature that is present in some types of repressor proteins is the existence of an *inducer molecule* that combines with the protein to either activate or inactivate its repression function. A *positive inducer* is a molecule that must be present in order for repression to occur. A *negative inducer* is one in which the presence of the inducer molecule blocks repression, either by changing the shape of the repressor protein or by blocking active sites on the repressor protein that would normally bind to the DNA. Figure 1.9a summarizes the various possibilities. Common examples of repressor-inducer pairs include *lacI* and lactose (or IPTG), *tetR* and ATc, and tryptophan repressor and tryptophan. Lactose/IPTG and ATc are both negative inducers, so their presence causes the otherwise repressed gene to be expressed, while tryptophan is a positive inducer.

The process of activation of a gene requires that an activator protein be present in order for transcription to occur. In this case, the protein must work to either recruit or enable RNA polymerase to begin transcription.

The simplest form of activation involves a protein binding to the DNA near the promoter in such a way that the combination of the activator and the promoter sequence bind RNA polymerase. One of the most well-studied examples is the *catabolite activator protein (CAP)*—also sometimes called the *cAMP receptor protein (CRP)*—shown in Figure 1.10. Like repressors, many activators have inducers, which can act in either a positive or negative fashion (see Figure 1.9b). For example, cyclic AMP (cAMP) acts as a positive inducer for CAP.

Another mechanism for activation of transcription, specific to prokaryotes, is the use of *sigma factors*. Sigma factors are part of a modular set of proteins that bind to RNA polymerase and form the molecular complex that performs transcription. Different sigma factors enable RNA polymerase to bind to different promoters, so the sigma factor acts as a type of activating signal for transcription. Table 1.2 lists some of the common sigma factors in bacteria. One of the uses of sigma factors is to produce certain proteins only under special conditions, such as when the cell undergoes *heat shock* (discussed in more detail in Chapter 5). Another use is to



Figure 1.9: Effects of inducers. Reproduced from Alberts et al. [2]; permission pending.

control the timing of the expression of certain genes, as illustrated in Figure 1.11.

In addition to repressors and activators, many genetic circuits also make use of *combinatorial promoters* that can act as either repressors or activators for genes. This allows genes to be switched on and off based on more complex conditions, represented by the concentrations of two or more activators or repressors.

Figure 1.12 shows one of the classic examples, a promoter for the *lac* system. In the *lac* system, the expression of genes for metabolizing lactose are under the control of a single (combinatorial) promoter. CAP, which is positively induced by cAMP, acts as an activator and LacI (also called “lac repressor”), which is negatively induced by lactose, acts as a repressor. In addition, the inducer cAMP is

Table 1.2: Sigma factors in *E. coli* [2].

Sigma factor	Promoters recognized
$\sigma^{70}$	most genes
$\sigma^{32}$	genes associated with heat shock
$\sigma^{28}$	genes involved in stationary phase and stress response
$\sigma^{28}$	genes involved in motility and chemotaxis
$\sigma^{24}$	genes dealing with misfolded proteins in the periplasm

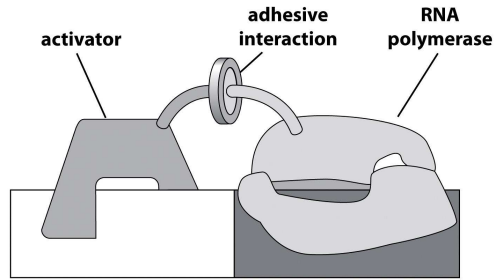
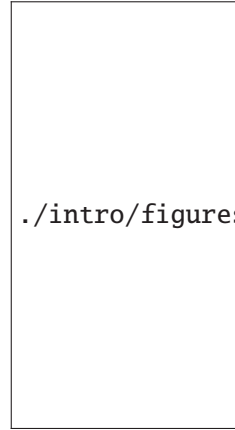


Figure 1&amp;7 Physical Biology of the Cell © Garland Science 2009

(a) Activation mechanism



(b) Examples of activators

Figure 1.10: Activation of gene expression. Figure from Phillips, Kondev and Theriot [56]; used with permission of Garland Science.

expressed only when glucose levels are low. The resulting behavior is that the proteins for metabolizing lactose are expressed only in conditions where there is no glucose (so CAP is active) *and* lactose is present.

More complicated combinatorial promoters can also be used to control transcription in two different directions, a example that is found in some viruses.

A final method of activation in prokaryotes is the use of *antitermination*. The basic mechanism involves a protein that binds to DNA and deactivates a site that would normally serve as a termination site for RNA polymerase. Additional genes are located downstream from the termination site, but without a promoter region. Thus, in the presence of the anti-terminator protein, these genes are not expressed (or expressed with low probability). However, when the antitermination protein is present, the RNA polymerase maintains (or regains) its contact with the DNA

./intro/figures/MBoC09\_07\_43.eps

Figure 1.11: Use of sigma factors to controlling the timing of expression. Reproduced from Alberts et al. [2]; permission pending.

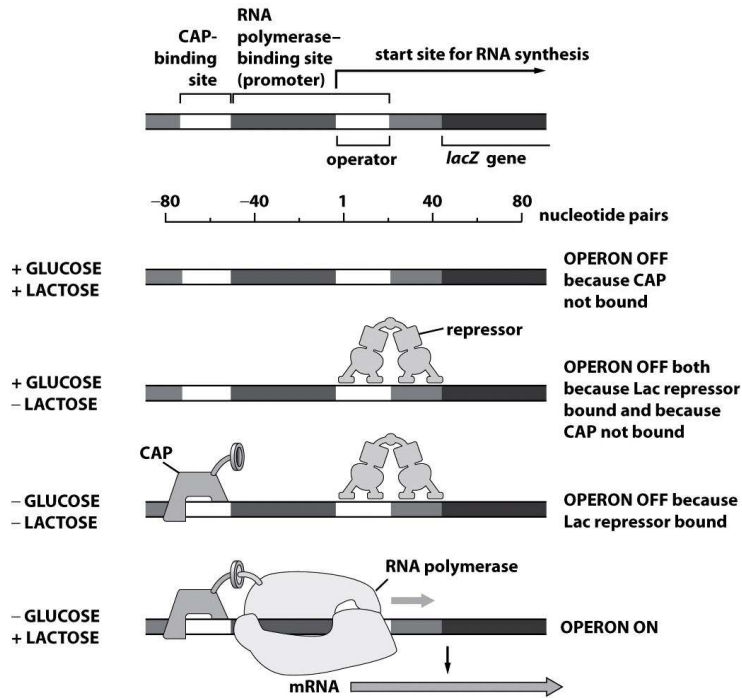


Figure 4.15 Physical Biology of the Cell (© Garland Science 2009)

Figure 1.12: Combinatorial logic for the *lac* operator. Figure from Phillips, Kondev and Theriot [56]; used with permission of Garland Science.

and expression of the downstream genes is enhanced. In this way, antitermination allows downstream genes to be regulated by repressing “premature” termination. An example of an antitermination protein is the protein N in phage  $\lambda$ , which binds to a region of DNA labeled Nut (for N utilization), as shown in Figure 1.13 and discussed in more detail in Section 5.3.

./intro/figures/GNM93-antitermination.eps

Figure 1.13: Antitermination. Reproduced from [33]; permission pending.

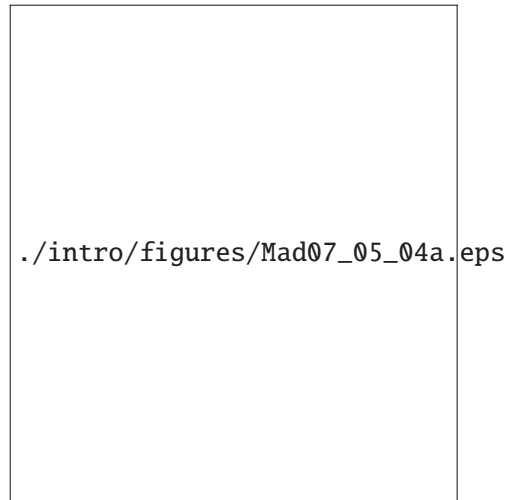


Figure 1.14: Phosphorylation of a protein via a kinase. Reproduced from Madhani [48]; permission pending.

### Post-transcriptional regulation of protein production

In addition to regulation that controls transcription of DNA into mRNA, a variety of mechanisms are available for controlling expression after mRNA is produced. These include control of splicing and transport from the nucleus (in eukaryotes), the use of various secondary structure patterns in mRNA that can interfere with ribosomal binding or cleave the mRNA into multiple pieces, and targeted degradation of mRNA. Once the polypeptide chain is formed, additional mechanisms are available that regulate the folding of the protein as well as its shape and activity level. We briefly describe some of the major mechanisms here.

Material to be written.

One of the most common types of post-transcriptional regulation is through the *phosphorylation* of proteins. Phosphorylation is an enzymatic process in which a phosphate group is added to a protein and the resulting conformation of the protein changes, usually from an inactive configuration to an active one. The enzyme that adds the phosphate group is called a *kinase* (or sometimes a *phosphotransferase*) and it operates by transferring a phosphate group from a bound ATP molecule to the protein, leaving behind ADP and the phosphorylated protein. *Dephosphorylation* is a complementary enzymatic process that can remove a phosphate group from a protein. The enzyme that performs dephosphorylation is called a *phosphatase*. Figure 1.14 shows the process of phosphorylation in more detail.

Phosphorylation is often used as a regulatory mechanism, with the phosphorylated version of the protein being the active conformation. Since phosphorylation and dephosphorylation can occur much more quickly than protein production and

degradation, it is used in my biological circuits in which a rapid response is required. One common motif is that a signaling protein will bind to a ligand and the resulting allosteric change allows the signaling protein to serve as a kinase. The newly active kinase then phosphorylates a second protein, which modulates other functions in the cell. Phosphorylation cascades can also be used to amplify the effect of the original signal; we will describe this in more detail in Section 2.5.

Kinases in cells are usually very specific to a given protein, allowing detailed signaling networks to be constructed. Phosphatases, on the other hand, are much less specific, and a given phosphatase species may desphosphorylate many different types of proteins. The combined action of kinases and phosphatases is important in signaling since the only way to deactivate a phosphorylated protein is by removing the phosphate group. Thus phosphatases are constantly “turning off” proteins, and the protein is activated only when sufficient kinase activity is present.

Phosphorylation of a protein occurs by the addition of a charged phosphate ( $\text{PO}_4$ ) group to the serine (Ser), threonine (Thr) or tyrosine (Tyr) amino acids. Similar covalent modifications can occur by the attachment of other chemical groups to select amino acids. *Methylation* occurs when a methyl group ( $\text{CH}_3$ ) is added to lysine (Lys) and is used for modulation of receptor activity and in modifying histones that are used in chromatin structures. *Acetylation* occurs when an acetyl group ( $\text{COCH}_3$ ) is added to lysine and is also used to modify histones. *Ubiquitination* refers to the addition of a small protein, ubiquitin, to lysine; the addition of a polyubiquitin chain to a protein targets it for degradation.

### 1.3 Control and Dynamical Systems Tools [AM08]

In this section we present a brief introduction to some of the key concepts from control and dynamical systems that are relevant for the study of biological systems. More details on the application of specific concepts listed here to biomolecular systems is provided in the main body of the text. Readers who are familiar with introductory concepts in dynamical systems and control, at the level described in Åström and Murray [1] for example, can skip this section.

#### Dynamics, feedback and control

A *dynamical system* is a system whose behavior changes over time, often in response to external stimulation or forcing. The term *feedback* refers to a situation in which two (or more) dynamical systems are connected together such that each system influences the other and their dynamics are thus strongly coupled. Simple causal reasoning about a feedback system is difficult because the first system influences the second and the second system influences the first, leading to a circular argument. This makes reasoning based on cause and effect tricky, and it is necessary to analyze the system as a whole. A consequence of this is that the behavior

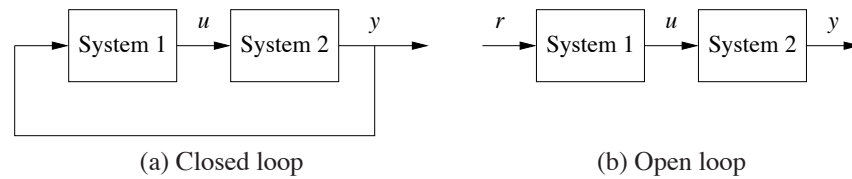


Figure 1.15: Open and closed loop systems. (a) The output of system 1 is used as the input of system 2, and the output of system 2 becomes the input of system 1, creating a closed loop system. (b) The interconnection between system 2 and system 1 is removed, and the system is said to be open loop.

of feedback systems is often counterintuitive, and it is therefore necessary to resort to formal methods to understand them.

Figure 1.15 illustrates in block diagram form the idea of feedback. We often use the terms *open loop* and *closed loop* when referring to such systems. A system is said to be a closed loop system if the systems are interconnected in a cycle, as shown in Figure 1.15a. If we break the interconnection, we refer to the configuration as an open loop system, as shown in Figure 1.15b.

A major source of examples of feedback systems is biology. Biological systems make use of feedback in an extraordinary number of ways, on scales ranging from molecules to cells to organisms to ecosystems. One example is the regulation of glucose in the bloodstream through the production of insulin and glucagon by the pancreas. The body attempts to maintain a constant concentration of glucose, which is used by the body's cells to produce energy. When glucose levels rise (after eating a meal, for example), the hormone insulin is released and causes the body to store excess glucose in the liver. When glucose levels are low, the pancreas secretes the hormone glucagon, which has the opposite effect. Referring to Figure 1.15, we can view the liver as system 1 and the pancreas as system 2. The output from the liver is the glucose concentration in the blood, and the output from the pancreas is the amount of insulin or glucagon produced. The interplay between insulin and glucagon secretions throughout the day helps to keep the blood-glucose concentration constant, at about 90 mg per 100 mL of blood.

Feedback has many interesting properties that can be exploited in designing systems. As in the case of glucose regulation, feedback can make a system resilient toward external influences. It can also be used to create linear behavior out of non-linear components, a common approach in electronics. More generally, feedback allows a system to be insensitive both to external disturbances and to variations in its individual elements.

Feedback has potential disadvantages as well. It can create dynamic instabilities in a system, causing oscillations or even runaway behavior. Another drawback, especially in engineering systems, is that feedback can introduce unwanted sensor noise into the system, requiring careful filtering of signals. It is for these reasons

that a substantial portion of the study of feedback systems is devoted to developing an understanding of dynamics and a mastery of techniques in dynamical systems.

Feedback systems are ubiquitous in both natural and engineered systems. Control systems maintain the environment, lighting and power in our buildings and factories; they regulate the operation of our cars, consumer electronics and manufacturing processes; they enable our transportation and communications systems; and they are critical elements in our military and space systems. For the most part they are hidden from view, buried within the code of embedded microprocessors, executing their functions accurately and reliably. Feedback has also made it possible to increase dramatically the precision of instruments such as atomic force microscopes (AFMs) and telescopes.

In nature, homeostasis in biological systems maintains thermal, chemical and biological conditions through feedback. At the other end of the size scale, global climate dynamics depend on the feedback interactions between the atmosphere, the oceans, the land and the sun. Ecosystems are filled with examples of feedback due to the complex interactions between animal and plant life. Even the dynamics of economies are based on the feedback between individuals and corporations through markets and the exchange of goods and services.

The mathematical study of the behavior of feedback systems is an area known as *control theory*. The term control has many meanings and often varies between communities. In engineering applications, we typically define control to be the use of algorithms and feedback in engineered systems. Thus, control includes such examples as feedback loops in electronic amplifiers, setpoint controllers in chemical and materials processing, “fly-by-wire” systems on aircraft and even router protocols that control traffic flow on the Internet. Emerging applications include high-confidence software systems, autonomous vehicles and robots, real-time resource management systems and biologically engineered systems. At its core, control is an *information science* and includes the use of information in both analog and digital representations.

A modern engineering control system senses the operation of a system, compares it against the desired behavior, computes corrective actions based on a model of the system’s response to external inputs and actuates the system to effect the desired change. This basic *feedback loop* of sensing, computation and actuation is the central concept in control. The key issues in designing control logic are ensuring that the dynamics of the closed loop system are stable (bounded disturbances give bounded errors) and that they have additional desired behavior (good disturbance attenuation, fast responsiveness to changes in operating point, etc). These properties are established using a variety of modeling and analysis techniques that capture the essential dynamics of the system and permit the exploration of possible behaviors in the presence of uncertainty, noise and component failure.

A typical example of a control system is shown in Figure 1.16. The basic elements of sensing, computation and actuation are clearly seen. In modern control

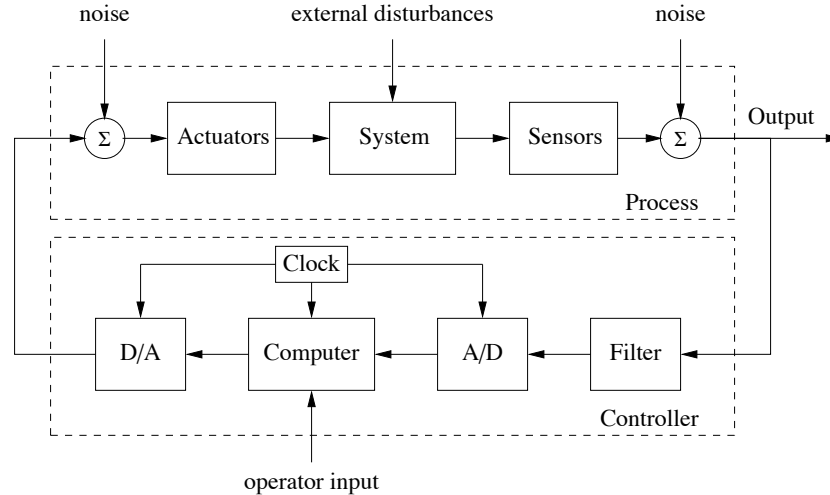


Figure 1.16: Components of a computer-controlled system. The upper dashed box represents the process dynamics, which include the sensors and actuators in addition to the dynamical system being controlled. Noise and external disturbances can perturb the dynamics of the process. The controller is shown in the lower dashed box. It consists of a filter and analog-to-digital (A/D) and digital-to-analog (D/A) converters, as well as a computer that implements the control algorithm. A system clock controls the operation of the controller, synchronizing the A/D, D/A and computing processes. The operator input is also fed to the computer as an external input.

systems, computation is typically implemented on a digital computer, requiring the use of analog-to-digital (A/D) and digital-to-analog (D/A) converters. Uncertainty enters the system through noise in sensing and actuation subsystems, external disturbances that affect the underlying system operation and uncertain dynamics in the system (parameter errors, unmodeled effects, etc). The algorithm that computes the control action as a function of the sensor values is often called a *control law*. The system can be influenced externally by an operator who introduces *command signals* to the system.

Control engineering relies on and shares tools from physics (dynamics and modeling), computer science (information and software) and operations research (optimization, probability theory and game theory), but it is also different from these subjects in both insights and approach.

Perhaps the strongest area of overlap between control and other disciplines is in the modeling of physical systems, which is common across all areas of engineering and science. One of the fundamental differences between control-oriented modeling and modeling in other disciplines is the way in which interactions between subsystems are represented. Control relies on a type of input/output modeling that allows many new insights into the behavior of systems, such as disturbance attenuation and stable interconnection. Model reduction, where a simpler (lower-fidelity)

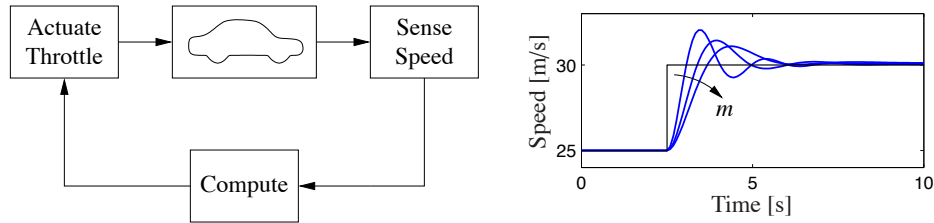


Figure 1.17: A feedback system for controlling the speed of a vehicle. In the block diagram on the left, the speed of the vehicle is measured and compared to the desired speed within the “Compute” block. Based on the difference in the actual and desired speeds, the throttle (or brake) is used to modify the force applied to the vehicle by the engine, drivetrain and wheels. The figure on the right shows the response of the control system to a commanded change in speed from 25 m/s to 30 m/s. The three different curves correspond to differing masses of the vehicle, between 1000 and 3000 kg, demonstrating the robustness of the closed loop system to a very large change in the vehicle characteristics.

description of the dynamics is derived from a high-fidelity model, is also naturally described in an input/output framework. Perhaps most importantly, modeling in a control context allows the design of *robust* interconnections between subsystems, a feature that is crucial in the operation of all large engineered systems.

### Feedback properties

Feedback is a powerful idea that is used extensively in natural and technological systems. The principle of feedback is simple: implement correcting actions based on the difference between desired and actual performance. In engineering, feedback has been rediscovered and patented many times in many different contexts. The use of feedback has often resulted in vast improvements in system capability, and these improvements have sometimes been revolutionary, as discussed above. The reason for this is that feedback has some truly remarkable properties, which we discuss briefly here.

*Robustness to Uncertainty.* One of the key uses of feedback is to provide robustness to uncertainty. By measuring the difference between the sensed value of a regulated signal and its desired value, we can supply a corrective action. If the system undergoes some change that affects the regulated signal, then we sense this change and try to force the system back to the desired operating point. This is precisely the effect that Watt exploited in his use of the centrifugal governor on steam engines.

As an example of this principle, consider the simple feedback system shown in Figure 1.17. In this system, the speed of a vehicle is controlled by adjusting the amount of gas flowing to the engine. Simple *proportional-integral* (PI) feedback is used to make the amount of gas depend on both the error between the current

and the desired speed and the integral of that error. The plot on the right shows the results of this feedback for a step change in the desired speed and a variety of different masses for the car, which might result from having a different number of passengers or towing a trailer. Notice that independent of the mass (which varies by a factor of 3!), the steady-state speed of the vehicle always approaches the desired speed and achieves that speed within approximately 5 s. Thus the performance of the system is robust with respect to this uncertainty.

Another early example of the use of feedback to provide robustness is the negative feedback amplifier. When telephone communications were developed, amplifiers were used to compensate for signal attenuation in long lines. A vacuum tube was a component that could be used to build amplifiers. Distortion caused by the nonlinear characteristics of the tube amplifier together with amplifier drift were obstacles that prevented the development of line amplifiers for a long time. A major breakthrough was the invention of the feedback amplifier in 1927 by Harold S. Black, an electrical engineer at Bell Telephone Laboratories. Black used *negative feedback*, which reduces the gain but makes the amplifier insensitive to variations in tube characteristics. This invention made it possible to build stable amplifiers with linear characteristics despite the nonlinearities of the vacuum tube amplifier.

*Design of Dynamics.* Another use of feedback is to change the dynamics of a system. Through feedback, we can alter the behavior of a system to meet the needs of an application: systems that are unstable can be stabilized, systems that are sluggish can be made responsive and systems that have drifting operating points can be held constant. Control theory provides a rich collection of techniques to analyze the stability and dynamic response of complex systems and to place bounds on the behavior of such systems by analyzing the gains of linear and nonlinear operators that describe their components.

An example of the use of control in the design of dynamics comes from the area of flight control. The following quote, from a lecture presented by Wilbur Wright to the Western Society of Engineers in 1901 [50], illustrates the role of control in the development of the airplane:

Men already know how to construct wings or airplanes, which when driven through the air at sufficient speed, will not only sustain the weight of the wings themselves, but also that of the engine, and of the engineer as well. Men also know how to build engines and screws of sufficient lightness and power to drive these planes at sustaining speed ... Inability to balance and steer still confronts students of the flying problem ... When this one feature has been worked out, the age of flying will have arrived, for all other difficulties are of minor importance.

The Wright brothers thus realized that control was a key issue to enable flight. They resolved the compromise between stability and maneuverability by building

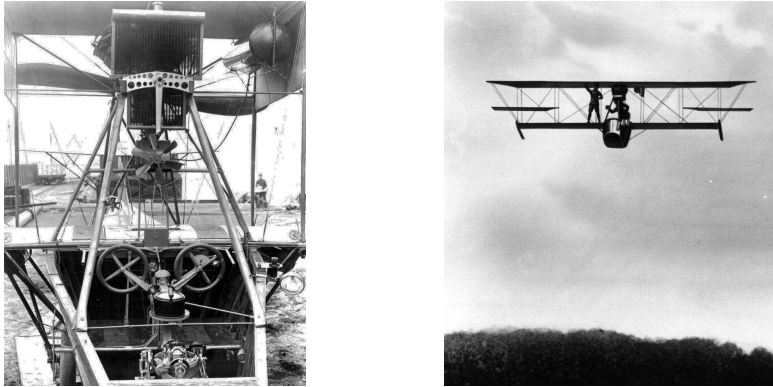


Figure 1.18: Aircraft autopilot system. The Sperry autopilot (left) contained a set of four gyros coupled to a set of air valves that controlled the wing surfaces. The 1912 Curtiss used an autopilot to stabilize the roll, pitch and yaw of the aircraft and was able to maintain level flight as a mechanic walked on the wing (right) [39].

an airplane, the Wright Flyer, that was unstable but maneuverable. The Flyer had a rudder in the front of the airplane, which made the plane very maneuverable. A disadvantage was the necessity for the pilot to keep adjusting the rudder to fly the plane: if the pilot let go of the stick, the plane would crash. Other early aviators tried to build stable airplanes. These would have been easier to fly, but because of their poor maneuverability they could not be brought up into the air. By using their insight and skillful experiments the Wright brothers made the first successful flight at Kitty Hawk in 1903.

Since it was quite tiresome to fly an unstable aircraft, there was strong motivation to find a mechanism that would stabilize an aircraft. Such a device, invented by Sperry, was based on the concept of feedback. Sperry used a gyro-stabilized pendulum to provide an indication of the vertical. He then arranged a feedback mechanism that would pull the stick to make the plane go up if it was pointing down, and vice versa. The Sperry autopilot was the first use of feedback in aeronautical engineering, and Sperry won a prize in a competition for the safest airplane in Paris in 1914. Figure 1.18 shows the Curtiss seaplane and the Sperry autopilot. The autopilot is a good example of how feedback can be used to stabilize an unstable system and hence “design the dynamics” of the aircraft.

One of the other advantages of designing the dynamics of a device is that it allows for increased modularity in the overall system design. By using feedback to create a system whose response matches a desired profile, we can hide the complexity and variability that may be present inside a subsystem. This allows us to create more complex systems by not having to simultaneously tune the responses of a large number of interacting components. This was one of the advantages of Black’s use of negative feedback in vacuum tube amplifiers: the resulting device

had a well-defined linear input/output response that did not depend on the individual characteristics of the vacuum tubes being used.

*Drawbacks of Feedback.* While feedback has many advantages, it also has some drawbacks. Chief among these is the possibility of instability if the system is not designed properly. We are all familiar with the undesirable effects of feedback when the amplification on a microphone is turned up too high in a room. This is an example of feedback instability, something that we obviously want to avoid. This is tricky because we must design the system not only to be stable under nominal conditions but also to remain stable under all possible perturbations of the dynamics.

In addition to the potential for instability, feedback inherently couples different parts of a system. One common problem is that feedback often injects measurement noise into the system. Measurements must be carefully filtered so that the actuation and process dynamics do not respond to them, while at the same time ensuring that the measurement signal from the sensor is properly coupled into the closed loop dynamics (so that the proper levels of performance are achieved).

Another potential drawback of control is the complexity of embedding a control system in a product. While the cost of sensing, computation and actuation has decreased dramatically in the past few decades, the fact remains that control systems are often complicated, and hence one must carefully balance the costs and benefits. An early engineering example of this is the use of microprocessor-based feedback systems in automobiles. The use of microprocessors in automotive applications began in the early 1970s and was driven by increasingly strict emissions standards, which could be met only through electronic controls. Early systems were expensive and failed more often than desired, leading to frequent customer dissatisfaction. It was only through aggressive improvements in technology that the performance, reliability and cost of these systems allowed them to be used in a transparent fashion. Even today, the complexity of these systems is such that it is difficult for an individual car owner to fix problems.

*Feedforward.* Feedback is reactive: there must be an error before corrective actions are taken. However, in some circumstances it is possible to measure a disturbance before it enters the system, and this information can then be used to take corrective action before the disturbance has influenced the system. The effect of the disturbance is thus reduced by measuring it and generating a control signal that counteracts it. This way of controlling a system is called *feedforward*. Feedforward is particularly useful in shaping the response to command signals because command signals are always available. Since feedforward attempts to match two signals, it requires good process models; otherwise the corrections may have the wrong size or may be badly timed.

The ideas of feedback and feedforward are very general and appear in many different fields. In economics, feedback and feedforward are analogous to a market-

based economy versus a planned economy. In business, a feedforward strategy corresponds to running a company based on extensive strategic planning, while a feedback strategy corresponds to a reactive approach. In biology, feedforward has been suggested as an essential element for motion control in humans that is tuned during training. Experience indicates that it is often advantageous to combine feedback and feedforward, and the correct balance requires insight and understanding of their respective properties.

*Positive Feedback.* In most of control theory, the emphasis is on the role of *negative feedback*, in which we attempt to regulate the system by reacting to disturbances in a way that decreases the effect of those disturbances. In some systems, particularly biological systems, *positive feedback* can play an important role. In a system with positive feedback, the increase in some variable or signal leads to a situation in which that quantity is further increased through its dynamics. This has a destabilizing effect and is usually accompanied by a saturation that limits the growth of the quantity. Although often considered undesirable, this behavior is used in biological (and engineering) systems to obtain a very fast response to a condition or signal.

One example of the use of positive feedback is to create switching behavior, in which a system maintains a given state until some input crosses a threshold. Hysteresis is often present so that noisy inputs near the threshold do not cause the system to jitter. This type of behavior is called *bistability* and is often associated with memory devices.

### Simple forms of feedback

The idea of feedback to make corrective actions based on the difference between the desired and the actual values of a quantity can be implemented in many different ways. The benefits of feedback can be obtained by very simple feedback laws such as on-off control, proportional control and proportional-integral-derivative control. In this section we provide a brief preview of some of these topics to provide a basis of understanding for their use in the chapters that follows.

*On-Off Control.* A simple feedback mechanism can be described as follows:

$$u = \begin{cases} u_{\max} & \text{if } e > 0 \\ u_{\min} & \text{if } e < 0, \end{cases} \quad (1.2)$$

where the *control error*  $e = r - y$  is the difference between the reference signal (or command signal)  $r$  and the output of the system  $y$  and  $u$  is the actuation command. Figure 1.19a shows the relation between error and control. This control law implies that maximum corrective action is always used.

The feedback in equation (1.2) is called *on-off control*. One of its chief advantages is that it is simple and there are no parameters to choose. On-off control often

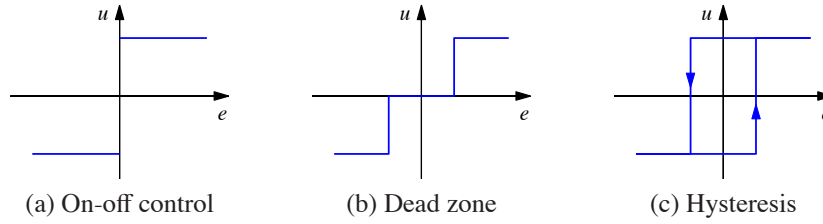


Figure 1.19: Input/output characteristics of on-off controllers. Each plot shows the input on the horizontal axis and the corresponding output on the vertical axis. Ideal on-off control is shown in (a), with modifications for a dead zone (b) or hysteresis (c). Note that for on-off control with hysteresis, the output depends on the value of past inputs.

succeeds in keeping the process variable close to the reference, such as the use of a simple thermostat to maintain the temperature of a room. It typically results in a system where the controlled variables oscillate, which is often acceptable if the oscillation is sufficiently small.

Notice that in equation (1.2) the control variable is not defined when the error is zero. It is common to make modifications by introducing either a dead zone or hysteresis (see Figure 1.19b and 1.19c).

*PID Control.* The reason why on-off control often gives rise to oscillations is that the system overreacts since a small change in the error makes the actuated variable change over the full range. This effect is avoided in *proportional control*, where the characteristic of the controller is proportional to the control error for small errors. This can be achieved with the control law

$$u = \begin{cases} u_{\max} & \text{if } e \geq e_{\max} \\ k_p e & \text{if } e_{\min} < e < e_{\max} \\ u_{\min} & \text{if } e \leq e_{\min}, \end{cases} \quad (1.3)$$

where  $k_p$  is the controller gain,  $e_{\min} = u_{\min}/k_p$  and  $e_{\max} = u_{\max}/k_p$ . The interval  $(e_{\min}, e_{\max})$  is called the *proportional band* because the behavior of the controller is linear when the error is in this interval:

$$u = k_p(r - y) = k_p e \quad \text{if } e_{\min} \leq e \leq e_{\max}. \quad (1.4)$$

While a vast improvement over on-off control, proportional control has the drawback that the process variable often deviates from its reference value. In particular, if some level of control signal is required for the system to maintain a desired value, then we must have  $e \neq 0$  in order to generate the requisite input.

This can be avoided by making the control action proportional to the integral of the error:

$$u(t) = k_i \int_0^t e(\tau) d\tau. \quad (1.5)$$

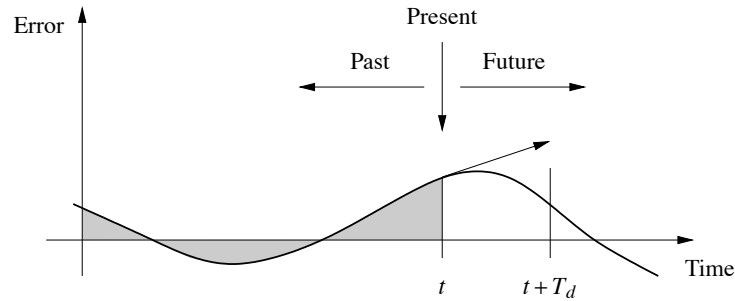


Figure 1.20: Action of a PID controller. At time  $t$ , the proportional term depends on the instantaneous value of the error. The integral portion of the feedback is based on the integral of the error up to time  $t$  (shaded portion). The derivative term provides an estimate of the growth or decay of the error over time by looking at the rate of change of the error.  $T_d$  represents the approximate amount of time in which the error is projected forward (see text).

This control form is called *integral control*, and  $k_i$  is the integral gain. It can be shown through simple arguments that a controller with integral action has zero steady-state error. The catch is that there may not always be a steady state because the system may be oscillating.

An additional refinement is to provide the controller with an anticipative ability by using a prediction of the error. A simple prediction is given by the linear extrapolation

$$e(t + T_d) \approx e(t) + T_d \frac{de(t)}{dt},$$

which predicts the error  $T_d$  time units ahead. Combining proportional, integral and derivative control, we obtain a controller that can be expressed mathematically as

$$u(t) = k_p e(t) + k_i \int_0^t e(\tau) d\tau + k_d \frac{de(t)}{dt}. \quad (1.6)$$

The control action is thus a sum of three terms: the past as represented by the integral of the error, the present as represented by the proportional term and the future as represented by a linear extrapolation of the error (the derivative term). This form of feedback is called a *proportional-integral-derivative (PID) controller* and its action is illustrated in Figure 1.20.

A PID controller is very useful and is capable of solving a wide range of control problems. More than 95% of all industrial control problems are solved by PID control, although many of these controllers are actually *proportional-integral (PI) controllers* because derivative action is often not included [21]. There are also more advanced controllers, which differ from PID controllers by using more sophisticated methods for prediction.

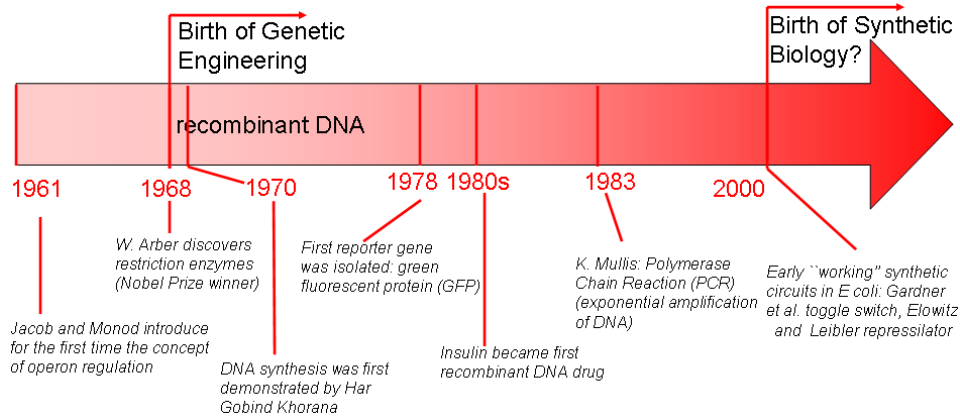


Figure 1.21: Milestones in the history of synthetic biology.

## 1.4 From Systems to Synthetic Biology

The rapidly growing field of synthetic biology seeks to use biological principles and processes to build useful engineering devices and systems. Applications of synthetic biology range from materials production (drugs, biofuels) to biological sensing and diagnostics (chemical detection, medical diagnostics) to biological machines (bioremediation, nanoscale robotics). Like many other fields at the time of their infancy (electronics, software, networks), it is not yet clear where synthetic biology will have its greatest impact. However, recent advances such as the ability to “boot up” a chemically synthesized genome [27] demonstrate the ability to synthesize systems that offer the possibility of creating devices with substantial functionality. At the same time, the tools and processes available to design systems of this complexity are much more primitive, and *de novo* synthetic circuits typically use a tiny fraction of the number of genetic elements of even the smallest microorganisms.

Several scientific and technological developments over the past four decades have set the stage for the design and fabrication of early synthetic biomolecular circuits (see Figure 1.21). An early milestone in the history of synthetic biology can be traced back to the discovery of mathematical logic in gene regulation. In their 1961 paper, Jacob and Monod introduced for the first time the idea of gene expression regulation through transcriptional feedback [42]. Only a few years later (1969), *restriction enzymes* that cut double-stranded DNA at specific recognition sites were discovered by Arber and co-workers [4]. These enzymes were a major enabler of recombinant DNA technology, in which genes from one organism are extracted and spliced into the chromosome of another. One of the most celebrated products of this technology was the large scale production of insulin by employing *E. coli* bacteria as a cell factory [72].

Another key innovation was the development of the polymerase chain reaction (PCR), devised in the 1980s, which allows exponential amplification of small amounts of DNA and can be used to obtain sufficient quantities for use in a variety of molecular biology laboratory protocols where higher concentrations of DNA are required. Using PCR, it is possible to “copy” genes and other DNA sequences out of their host organisms.

The developments of recombinant DNA technology, PCR and artificial synthesis of DNA provided the ability to “cut and paste” natural or synthetic promoters and genes in almost any fashion. This cut and paste procedure is called *cloning* and consists of four primary steps: *fragmentation*, *ligation*, *transfection* and *screening*. The DNA of interest is first isolated using restriction enzymes and/or PCR amplification. Then, a ligation procedure is employed in which the amplified fragment is inserted into a vector. The vector is often a piece of circular DNA, called a plasmid, that has been linearized by means of restriction enzymes that cleave it at appropriate restriction sites. The vector is then incubated with the fragment of interest with an enzyme called *DNA ligase*, producing a single piece of DNA with the target DNA inserted. The next step is to transfect (or transform) the DNA into living cells, where the natural replication mechanisms of the cell will duplicate the DNA when the cell divides. This process does not transfect all cells, and so a selection procedure is required to isolate those cells that have the desired DNA inserted in them. This is typically done by using a plasmid that gives the cell resistance to a specific antibiotic; cells grown in the presence of that antibiotic will only live if they contain the plasmid. Further selection can be done to insure that the inserted DNA is also present.

Once a circuit has been constructed, its performance must be verified and, if necessary, debugged. This is often done with the help of *fluorescent reporters*. The most famous of these is GFP, which was isolated from the jellyfish *Aequorea victoria* in 1978 by Shimomura [?]. Further work by Chalfie, Tsujii and others in the 1990s enabled the use of GFP in *E. coli* as a fluorescent reporter by inserting it into an appropriate point in an artificial circuit. By using spectrofluorometry, fluorescent microscopy or flow cytometry, it is possible to measure the amount of fluorescence in individual cells or collections of cells and characterize the performance of a circuit in the presence of inducers or other factors.

Two early examples of the application of these technologies were the *repressilator* [23] and a synthetic genetic switch [].

The repressilator is a synthetic circuit in which three proteins each repress another in a cycle. This is shown schematically in Figure 1.22a, where the three proteins are TetR,  $\lambda$  cI and LacI. The basic idea of the repressilator is that if TetR is present, then it represses the production of  $\lambda$  cI. If  $\lambda$  cI is absent, then LacI is produced (at the unregulated transcription rate), which in turn represses TetR. Once TetR is repressed, then  $\lambda$  cI is no longer repressed, and so on. If the dynamics of the circuit are designed properly, the resulting protein concentrations will oscillate,

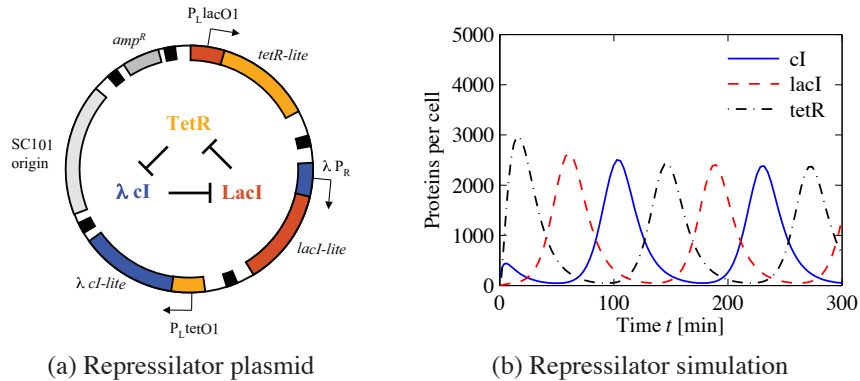


Figure 1.22: The repressilator genetic regulatory network. (a) A schematic diagram of the repressilator, showing the layout of the genes in the plasmid that holds the circuit as well as the circuit diagram (center). (b) A simulation of a simple model for the repressilator, showing the oscillation of the individual protein concentrations. (Figure courtesy M. Elowitz.)

as shown in Figure 1.22b.

The genetic switch consists of two repressors connected together in a cycle, as shown in Figure 1.23a. The intuition behind this circuit is that if the gene A is being expressed, it will repress production of B and maintain its expression level (since the protein corresponding to B will not be present to repress A). Similarly, if B is being expressed, it will repress the production of A and maintain its expression level. This circuit thus implements a type of *bistability* that can be used as a simple form of memory. Figure 1.23b shows the time traces for a system, illustrating the bistable nature of the circuit. When the initial condition starts with a concentration of protein B greater than that of A, the solution converges to the equilibrium point where B is on and A is off. If A is greater than B, then the opposite situation results.

These seemingly simple circuits took years to get to work, but showed that it was possible to synthesize a biological circuit that performed a desired function that was not originally present in a natural system. Today, commercial synthesis of DNA sequences and genes has become cheaper and faster, with a price often below \$0.30 per base pair.<sup>1</sup> The combination of inexpensive synthesis technologies, new advances in cloning techniques, and improved devices for imaging and measurement has vastly simplified the process of producing a sequence of DNA that encodes a given set of genes, operator sites, promoters and other functions, and these techniques are a routine part of undergraduate courses in molecular and synthetic biology.

As illustrated by the examples above, current techniques in synthetic biology have demonstrated the ability to program biological function by designing DNA sequences that implement simple circuits. Most current devices make use of tran-

<sup>1</sup>As of this writing; divide by a factor of two for every two years after the publication date.

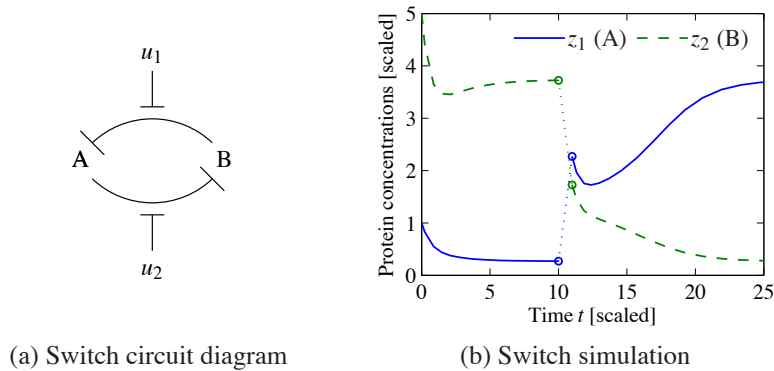


Figure 1.23: Stability of a genetic switch. The circuit diagram in (a) represents two proteins that are each repressing the production of the other. The inputs  $u_1$  and  $u_2$  interfere with this repression, allowing the circuit dynamics to be modified. The simulation in (b) shows the time response of the system starting from two different initial conditions. The initial portion of the curve corresponds to protein B having higher concentration than A, and converges to an equilibrium where A is off and B is on. At time  $t = 10$ , the concentrations are perturbed, moving the concentrations into a region of the state space where solutions converge to the equilibrium point with the A on and B off.

scriptional or post-transcriptional processing, resulting in very slow timescales (response times typically measured in tens of minutes to hours). This restricts their use in systems where faster response to environmental signals is needed, such as rapid detection of a chemical signal or fast response to changes in the internal environment of the cell. In addition, existing methods for biological circuit design have limited modularity (reuse of circuit elements requires substantial redesign or tuning) and typically operate in very narrow operating regimes (e.g., a single species grown in a single type of media under carefully controlled conditions).

As an illustration of the dynamics of typical synthetic devices in use today, Figure 1.24 shows a typical response of a genetic element to an inducer molecule [14]. In this circuit, an external signal of homoserine lactone (HSL) is applied at time zero and the system reaches 10% of the steady state value in approximately 15 minutes. This response is limited in part by the time required to synthesize the output protein (GFP), including delays due to transcription, translation and folding. Since this is the response time for the underlying “actuator”, circuits that are composed of feedback interconnections of such genetic elements will typically operate at 5–10 times slower speeds. While these speeds are appropriate in many applications (e.g., regulation of steady state enzyme levels for materials production), in the context of biochemical sensors or systems that must maintain a steady operating point in more rapidly changing thermal or chemical environments, this response time is too slow to be used as an effective engineering approach.

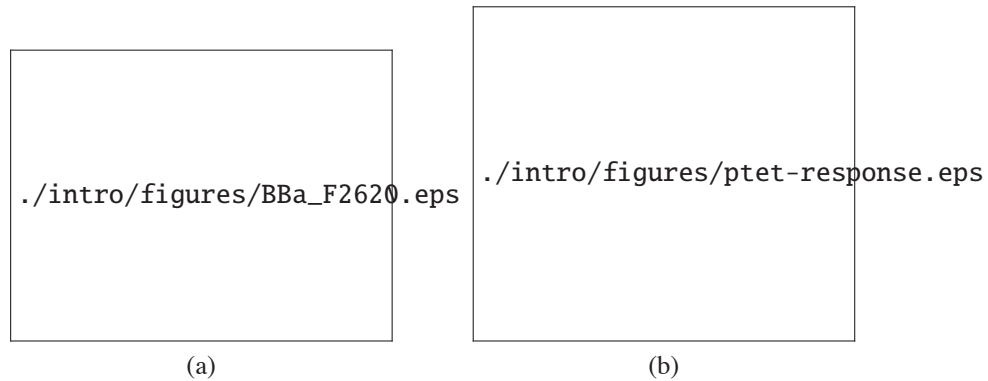


Figure 1.24: Expression of a protein using an inducible promoter [14]. (a) The circuit diagram indicates the DNA sequences that are used to construct the part (chosen from the BioBrick library). (b) The measured response of the system to a step change in the inducer level (HSL).

By comparison, the frequency response for the signaling component in *E. coli* chemotaxis is shown in Figure 1.25 [?]. Here the response of the kinase CheA is plotted in response to an exponential ramp in the ligand concentration. The response is extremely rapid, with the timescale measured in seconds. This rapid response is implemented by conformational changes in the proteins involved in the circuit, rather than regulation of transcription or other slower processes.

The field of synthetic biology has the opportunity to provide new approaches to solving engineering and scientific problems. Sample engineering applications include the development of synthetic circuits for producing biofuels, ultrasensitive chemical sensors, or production of materials with specific properties that are tuned to commercial needs. In addition to the potential impact on new biologically engineered devices, there is also the potential for impact in improved understanding of biological processes. For example, many diseases such as cancer and Parkinson's disease are closely tied to kinase dysfunction. Our analysis of robust systems of kinases and the ability to synthesize systems that support or invalidate biological hypotheses may lead to a better systems understanding of failure modes that lead to such diseases.

## 1.5 Further Reading

There are numerous survey articles and textbooks that provide more detailed introductions to the topics introduced in this chapter. In the area of systems biology, the textbook by Alon [3] provides a broad view of some of the key elements of modern systems biology. A more comprehensive set of topics is covered in the recent textbook by Klipp [?], while a more engineering-oriented treatment of modeling

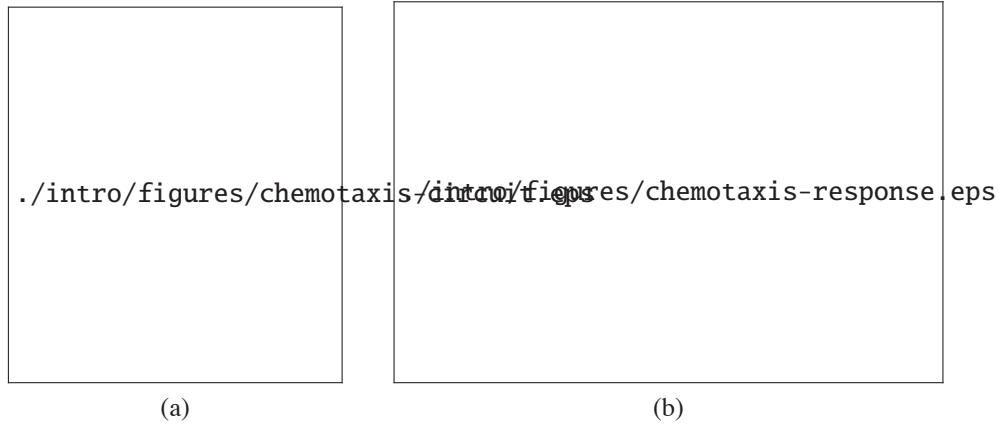


Figure 1.25: Responses of *E. coli* signaling network to exponential ramps in ligand concentration. (a) A simplified circuit diagram for chemotaxis, showing the biomolecular processes involved in regulating flagellar motion. (b) Time responses of the “sensing” subsystem (from Shimizu, Tu and Berg; *Molecular Systems Biology*, 2010), showing the response to exponential inputs.

of biological circuits can be found in the text by Myers [?]. Two other books that are particularly noteworthy are Ptashne’s book on the phage  $\lambda$  [58] and Madhani’s book on yeast [48], both of which use well-studied model systems to describe a general set of mechanisms and principles that are present in many different types of organisms.

The topics in dynamical systems and control theory that are briefly introduced here are covered in more detail in AM08 [1], to which this text is a supplement. Other books that introduce tools for modeling and analysis of dynamical systems with applications in biology include the two-volume text by J. D. Murray [52] and the recent text by and Ellner and Guckenheimer [22].

Synthetic biology is a rapidly evolving field that includes many different sub-areas of research, but few textbooks are currently available. In the specific area of biological circuit design that we focus on here, there are a number of good survey and review articles. The article by Baker *et al* [9] provides a high level description of the basic approach and opportunities. Recent survey and review papers include Voigt [?] and Khalil and Collins [?].

