Building the components for a biomolecular computer

Clint Morgan Darko Stefanovic Cristopher Moore
Department of Computer Science, University of New Mexico
Milan N. Stojanovic
Department of Medicine, Columbia University

March 1, 2004

Abstract

We propose a new technology for amorphous bio-compatible computing, using deoxyribozyme-based logic gates, and derive specific designs for a biomolecular realization of elementary components for a digital computer, including bistable memory elements (flip-flops) and oscillators. An eventual goal of this technology is the development of *in-vivo* computational devices. Thus, our system for chemical computation offers functionality similar to conventional electronic circuits with the potential for deployment inside of living cells.

1 Introduction

1.1 Background

Electronic computation has proved enormously successful in the decades since its inception. However, this success has not been without limitations. For example, explicit wired connections between devices are needed to transmit signals. In highly connected networks, this creates problems as the number of wires scales quadratically with the number of elements. Moreover, the operating environment of the electronic computer is limited by the architecture which supports it. Electronic computers cannot scale down to operate in cellular environments (in living beings), and moreover current implementations in silicon are not bio-compatible. New systems of computation are being developed to overcome inherent limitations of electronic computing. This work will focus on computational systems constructed from a DNA-based technology.

Several methods for computing with DNA have been proposed. Many of these approaches solve combinatorial problems using the ability of DNA to self-assemble in parallel. This approach was demonstrated by Adleman to solve the NP-complete Hamiltonian Path Problem [1]. In his experiments, the vertices and edges of a graph are encoded as single-stranded DNA. The strands are allowed to self-assemble such that a double-stranded complex will represent a path in the graph. Laboratory techniques are then used to filter the paths such that only a Hamiltonian is remaining. While this approach provided a demonstration of the utility of DNA-based computation, it was not a practical approach. As the size of the graph is increased the amount of DNA needed becomes prohibitively large [2]. The process requires macroscopic interfaces which adds an additional level of complexity. More importantly, this approach is not general enough to perform arbitrary computations.

In a step toward general computation Benenson et al. have developed a DNA-based "finite state automaton" [3]. This scheme uses a clever system to encode transition rules and an input sequence into a double-stranded DNA molecule. An enzyme acts on the complex to perform the state transitions without any macroscopic intervention. The automaton demonstrated was relatively simple; it consisted of a two-state machine operating on a two-letter alphabet. Still, this device is significant in that its functioning does not require outside interference.

1.2 DNA-based Boolean Gates

Our chemistry consists of a new class of enzyme-catalyzed reactions constructed from deoxyribozyme technology. These reactions can be used to transform gate, input, and substrate signals into product signals and thereby perform simple computations. This technology differs from other DNA computation schemes in that it is composable: the inputs are of the same type as the outputs. As a result, gates may be combined into various reactions networks designed to perform more complex computations. Thus, we may design chemical systems from simple boolean primitives in much the same way electronic computers are built from simple logic gates. These devices operate without macroscopic intervention in a biological environment. As a result, a goal of this technology is autonomous *in vivo* computations for diagnostic and therapeutic purposes.

Gates which perform several simple boolean computations have been developed. In the past these gates have implemented a single layer of logic, and no inter-gate communication was employed [4, 5]. Devices that function as a half-adder [6] and a tic-tac-toe playing automaton [7] have been built and tested in the laboratory. These gates have been deployed in a closed reactor, which effectively limits this technology to one-shot boolean computations.

To overcome this limitation, we explore using this chemistry in an open reactor. Moving to an open reactor adds a level of complexity to the engineering task and creates a fundamental change in the design process. This work develops a process that may be used to engineer these devices. We apply methods of dynamical systems to construct reaction networks in open reactors that implement rudimentary elements of digital chemical computation.

In order to further increase the capabilities of this technology, we investigate more complex reaction networks that make use of inter-gate communication and feedback. Feedback is used to construct flip-flops and clocking signals, which are fundamental components of digital electronic circiuts, including general-purpose computers. This work shows that adding feedback to our reaction networks allows the construction of fundamental components of a biomolecular computer—a flip-flop and an oscillator—from a chemical substrate.

This work is organized as follows: In Section 2 we introduce the chemical technology that will be used through the rest of the work and present a mathematical model to describe its operation. In Section 3 we describe the open systems in which our reactions will occur and introduce the techniques that will be used to explore these systems. In Section 4 we use these concepts to build a simple inverter and in Section 5 we present a more error-resistant inverter. In Section 6 we present a flip-flop and in Section 7 an oscillator. Finally, in Section 8 we draw some conclusions.

2 The Chemistry

Deoxyribozyme-based gates capable of boolean computation in solution have been developed [4]. The gates are constructed in deoxyribozyme technology, and are completely biocompatible. They may operate in living cells and on cellular material. Modular construction techniques are employed such that a variety of interconnections between elements are possible. This allows one to design complex chemical systems using our gates as the basic functional unit.

The four components of these deoxyribozyme systems are inputs, gates, substrates, and products. Under certain input conditions a gate is an active enzyme. The effect of input molecules on the catalytic activity of the gate defines the logic operation that the gate performs. A given gate may require the presence and/or absence of certain inputs to be active. When active, the enzymatic gate catalyzes a cleavage reaction and a substrate molecule is cleaved into two product molecules. The product molecules represent the output signal of the gate. Currently, a variety of one-, two-, and three-input gates have been developed. In this work we will use a one-input NOT gate and a two-input AND gate to engineer several types of circuits. Figure 1 shows chemical and electronic schematics of these two gates.

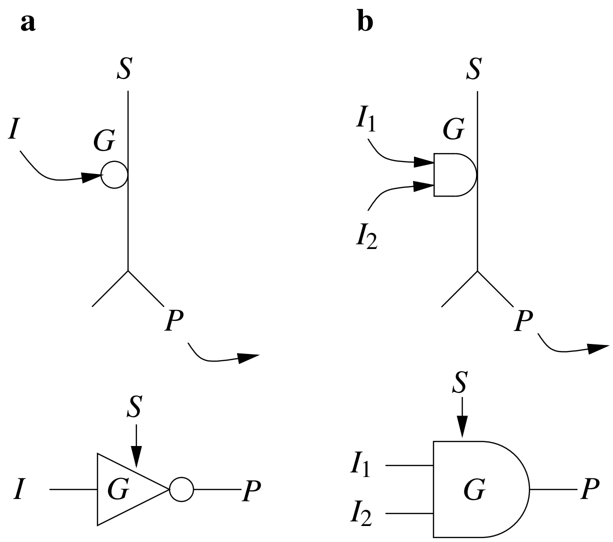

Figure 1: (a) An inverter. Top: schematic of an enzymatic NOT gate. In the absence of input I the gate G cleaves substrate S to produce product P. If I has bound to G, the gate becomes inactive and does not produce product. Bottom: schematic of the corresponding electronic logic. (b) An AND gate. Top: Schematic of enzymatic AND gate. In the presence of inputs I_1 and I_2 the gate G cleaves substrate S to produce product P. In the absence of either I_1 or I_2 the gate is inactive and does not produce product. Bottom: schematic of the corresponding electronic logic.

2.1 Signal Representation in Solution

Standard electronic circuits have the need for physical connections between spatially localized circuit elements; enzymatic gates have no such restrictions. Instead, computations are carried out in solution with communication between elements accomplished by diffusion of the oligonucleotide molecules.

Logic signals are expressed by concentrations of specific oligonucleotides. These molecules transmit information by participating in the reactions of multiple gates. The simplest example is a molecule that is a product of one gate and an input to another; serving as a substrate would suffice as well. The absence of the oligonucleotide in solution represents a low logic value, while the presence beyond a certain threshold represents a high logic value. Just as the definition of a high voltage may change in different types of electronic circuits, the specific concentrations which represent a logic high or low may change from circuit to circuit, depending on the individual needs of the system and the restrictions imposed by the underlying chemical technology.

2.2 Reaction Kinetics

We will be analyzing and specifying physical systems, so it will be worthwhile to briefly describe the units that will be used throughout this work. The units used for time will be seconds (s), for volume will be cubic meters (m^3), for amount of substance will be moles (mol), and for concentration will be molarity ($\mathrm{M} = \frac{1000\,\mathrm{mol}}{\mathrm{m}^3}$). Additionally, we will make use of standard metric prefixes. To clarify our discussion we will omit units unless presenting actual values.

The general chemical mechanism of a deoxyribozyme gate is as follows. First, input molecules bind to the designated locations on the gate molecules. The binding of an input to a gate may have an effect on the physical shape of the gate, which in turn may affect catalytic activity. Under the appropriate circumstances, the gate is an active enzyme, in which case it will bind to a substrate molecule, cleave it into two molecules of product, and separate into two molecules of product and one active gate complex. Active gates will continue to operate as long as there is substrate remaining to be cleaved.

In order to design larger circuits, we must first understand the dynamic behavior of individual logic gates. Above we discussed the chemical principles that make the gates feasible; now we examine the chemical kinetics. We set up the experiment as follows. We prepare a solution with a concentration of $G = 250\,\mathrm{nM}$ of a specific YES gate (which becomes active in the presence of input), a certain concentration I of the matching input, and a concentration $S = 2500\,\mathrm{nM}$ of the substrate cleaved by the gate. At 15-minute intervals we record the instrumentally measured fluorescence. We repeat the experiment varying I, starting with $I = S$ and repeatedly halving it.

The measured fluorescence level of a molecular species is proportional to its concentration. The specific fluorescences of the product and the substrate have been established separately and they are in a ratio of 8:1. Therefore the increase of total fluorescence is proportional to the amount of product, which allows us to convert measured fluorescence into product concentration, shown in Figure 2.

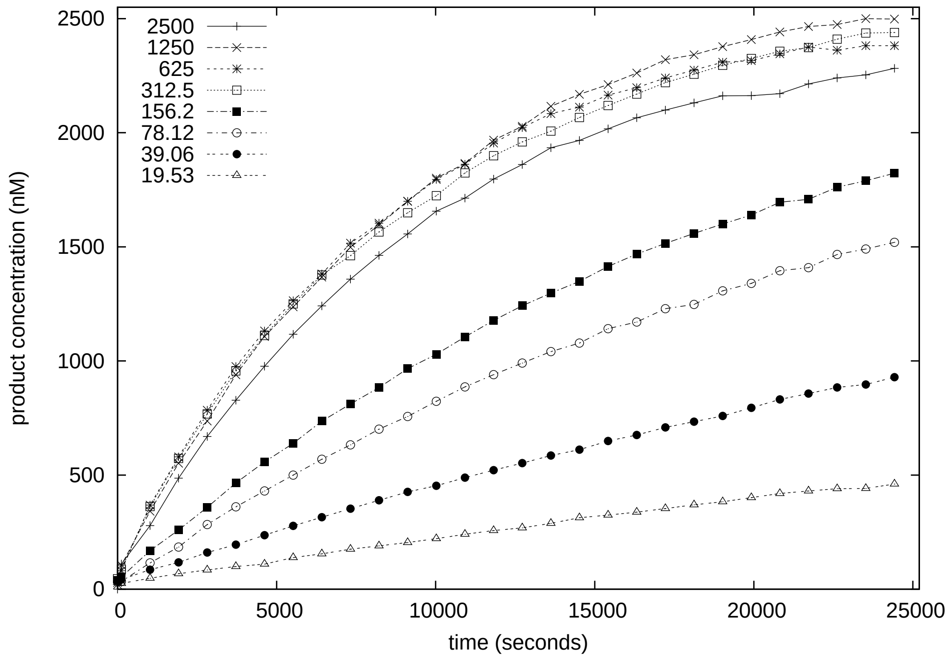

Figure 2: Experimentally measured behavior of a deoxyribozyme gate in isolation. Curves correspond to different input concentrations I (nM).

For small values of the input concentration I, product concentration[1] P rises approximately linearly with time, with slope proportional to I. For larger values of I, the growth soon reaches a plateau defined by the initial substrate concentration S: when all of the substrate has been converted to product, the reaction stops. Note also the saturating behavior: whereas the slope of $t \mapsto P$ increases with I for small I, it remains roughly constant for $I > G$. Lastly, with no input, product concentration remains 0 (not shown).

The kinetics of this reaction will be modeled as follows. First we note that the cleavage and separation of substrate molecules is the slowest of the reactions. As such it is the rate-limiting process. We will assume that bonding between gate and input molecules is instantaneous and complete. Thus, the number of active gates at a given time is a simple calculation depending solely on the number of gates and inputs. Consider a NOT gate, which is active when it has not bound with an input. The number of active NOT gates in solution may be calculated as the total number of gates less the number of inputs. No gates will be active if there are more inputs than gates.

In addition to the NOT gate, this work will use an AND gate that is activated by the presence of two identical molecules of input. To calculate the number of active AND gates G with I inputs in solution we assume random binding of inputs to gates. There are two binding sites per gate, so each site has a probability of $\frac{I}{2G}$ of binding to input. Two inputs are required to activate a gate, so the probability that an individual gate is active is $\frac{I^2}{4G^2}$. Hence the total number

[1] We only track one of the two cleavage products; their concentrations are obviously equal.

of active gates is calculated as the minimum of $\frac{I^2}{4G}$ and G.

Cleavage of substrate requires both substrate and an active gate complex. Experiments have shown that the rate of production is proportional to the concentration of both reactants. Hence, a model for the rate at which product is produced is: $\frac{dP}{dt} = \beta S G_A$, where P is the product concentration, β is the reaction rate constant, S is the substrate concentration, and G_A is the concentration of *active* gates. In the experiments shown in Figure 2, I (and G_A) was time-invariant. Therefore, the solution is an exponential decay of the substrate S. This model agrees well with observed measurements in Figure 2, and analysis of the measured data gives a rough estimate of the reaction rate constant $\beta = 5 \cdot 10^{-7}\,\mathrm{nM}^{-1}\mathrm{s}^{-1}$. This value will be used to model the chemical gates in the circuit designs presented herein.

The kinetic model is only valid for a limited range of concentrations. As the concentrations are increased beyond a certain threshold, effects start to occur which are not present in the model. In this work we will limit ourselves to gate concentrations between 0 and $10\,\mu$M and substrate concentrations between 0 and $20\,\mu$M, for which ranges this model has been verified in the laboratory.

3 The Reactor

Chemical reactors may be divided into two broad categories: closed systems where reactants are added to a solution and the reaction is allowed to proceed toward equilibrium, and open systems where reactants are continuously supplied and excess solution is removed. This work explores the benefits and design considerations associated with using our chemistry in an open reactor. Previous theoretical work has described circuits created from hypothetical enzymatic transistors in open reactors [8]. We will expand on this work by presenting several circuit designs based on the chemical technology described in the previous chapter.

3.1 Reactions in a Closed System

Reactions in a closed environment are subject to the Second Law of Thermodynamics which posits that the free energy in a closed system will continuously decrease and implies that the system will move toward an equilibrium. This does not rule out interesting behavior, such as oscillations on the way to an equilibrium [9]. However, it does imply a finite number of cycles through digital values. An even greater restriction comes from the fact that the gates as described function as integrators rather than straight boolean gates. Concentrations of product may rise (and logic signals move from low to high), but they cannot fall. This effectively limits gates in a closed system to one-shot boolean computations.

3.2 Thermodynamically Open Reactions

In order to avoid the restrictions of closed systems, this work uses a system which is thermodynamically open; material is continuously supplied and removed, as in a living cell. The circuit may be reused and produce many outputs over its lifetime–we now have recurrent chemical circuits.

3.2.1 The CSTR

While a long-term goal of this technology is the deployment inside of (thermodynamically open) living cells, the first step toward that end is testing and verification in a laboratory setting. A model open environment is the continuous-flow stirred tank reactor (CSTR). A CSTR consists of a mechanism (usually some type of pump) for delivering reactants into a reaction chamber which is stirred to maintain a uniform distribution of chemical species. An outflow removes a homogeneous sample of solution from the reactor to maintain constant volume.

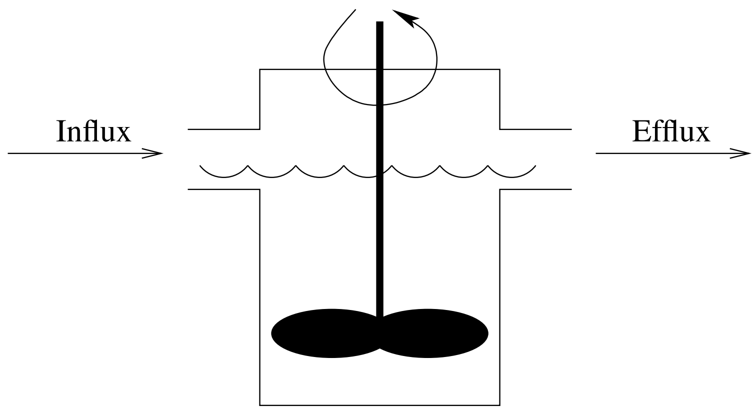

Figure 3: Schematic of a CSTR. Reactants are fed into the reactor via influx. The reaction chamber is well-stirred. Excess solution is removed through efflux.

The inputs to a CSTR may be varied in terms of concentration of input solution and in terms of flow rate into the reactor. Note that both the concentration and the *volumetric* influx of a solution can be varied while still maintaining the same total *molecular* influx rate. Thus we have the ability to manipulate total efflux while still maintaining desired concentrations of chemical species inside the reactor.

The decay rate of the reactor (k) is equal to the efflux rate (E) divided by the volume (V) of the reactor. Thus decay rate is proportional to efflux rate. As the decay rate of the reactor is increased, material will spend less time inside the reactor and the previous state will fall away more quickly. Because the reactor state can change more quickly, the circuit speed will increase. However, this increases the amount of chemical species needed to maintain the same levels of concentration and so increases cost.

In order to make a fair comparison between the circuits presented, we will require that all reactors have the same total efflux of $5 \cdot 10^{-8}\,\mathrm{m}^3\mathrm{s}^{-1}$ and volume of $5 \cdot 10^{-4}\,\mathrm{m}^3$. This results in a decay rate of $10^{-4}\,\mathrm{s}^{-1}$. While the resulting circuits will operate *very* slowly, these values were chosen as realistic design points corresponding to equipment that can be build in a chemistry laboratory at modest expense.

3.2.2 A Simple Example

To illustrate our mathematical model, let us consider a simple hypothetical reaction. Two species A and B react at a rate β to produce P. If the reaction were to occur in a closed reactor, then we could model the system with the following set of three ordinary differential equations:

$$\begin{aligned}
\frac{dA}{dt} &= -\beta AB \\
\frac{dB}{dt} &= -\beta AB \\
\frac{dP}{dt} &= \beta AB
\end{aligned}$$

where the variables A, B, and P are continuous dynamical variables that represent the concentration of the respective species.

Now suppose we wish to model this reaction in a CSTR of volume V. The concentrations of input solutions are A^c and B^c; the rates of influx for the two solutions are A^i and B^i. It follows that the *molecular* influxes of A and B are $A^m = A^c A^i$ and $B^m = B^c B^i$, and the total efflux of the system is $E = A^i + B^i$. We can describe the dynamics of the open reactor with the following set of three ordinary differential equations:

$$\begin{aligned}
\frac{dA}{dt} &= \frac{A^m}{V} - \beta AB - \frac{AE}{V} \\
\frac{dB}{dt} &= \frac{B^m}{V} - \beta AB - \frac{BE}{V} \\
\frac{dP}{dt} &= \beta AB - \frac{PE}{V}
\end{aligned}$$

The three terms describing the rate of change of A and B describe, from left to right, the influx rate, then reaction rate, and the efflux rate. Because P is not being supplied, its rate equation need only describe the reaction rate and the efflux rate.

This system of equations can be numerically integrated to simulate the behavior of the system. Additionally, we can use techniques from ordinary differential equations and non-linear systems analysis to obtain qualitative and quantitative insights to the behavior of the system. This example is simple enough that, along with initial conditions, it may be solved explicitly. However, more complex reaction terms may introduce non-linearity into the system, which usually prevents an explicit solution.

3.3 Analytical Techniques

We have shown how to transform a network of chemical reactions and a specification for a reactor into a system of differential equations. Now we need analytical tools to explore the model. Dynamical systems theory provides techniques to do exactly this.

Some systems may be solved explicitly. If a given system falls into this class, then we may use standard techniques to obtain an exact solution. However, the vast majority of dynamical systems encountered have no explicit solution [10]. Despite the fact that we cannot obtain a precise description of system, we can still assemble qualitative knowledge about its behavior.

We can learn a great deal about a dynamical system by examining the lines on which one of the variables is unchanging—the *nullclines*. If a system is described by a set of differential equations, the nullclines can be found by setting each rate equation to zero and solving for the concentrations. The lines can then be plotted in the *phase-space*–a n-dimensional space of the n dynamical variables. In a two dimensional phase-space the nullcline will represent areas where the system evolves only in one direction. The intersection of two nullclines implies a fixed point—a location in the phase space where the system is unchanging.

Once a fixed point has been identified we would like to characterize how the system behaves in the neighborhood of that point. The point could be stable—points near the fixed point are attracted to it, or it could be unstable—points near the fixed point will fall away. Other possibilities include saddle point—the system will fall away in one direction and toward in another, and a center—the system will orbit the fixed point in a limit cycle.

One method of analytically determining the stability of fixed points is to compute a linearization of the derivative near the steady state. By performing a Taylor series expansion about the point and disregarding the quadratic and higher order terms we can determine the effect of small perturbations from the fixed point. This linearization may be done in multi-dimensional systems and is known as the *Jacobian* matrix.

For example, if a two dimensional system is described by $\frac{dx}{dt} = f(x,y)$ and $\frac{dy}{dt} = g(x,y)$, then a fixed point (x^*, y^*) will occur at a simultaneous solution to $f(x,y) = 0$ and $g(x,y) = 0$. To calculate the nature of the fixed point we will compute the Jacobian as:

$$\mathrm{J} = \begin{bmatrix} \frac{\partial x}{\partial f} & \frac{\partial y}{\partial f} \\ \frac{\partial x}{\partial g} & \frac{\partial y}{\partial g} \end{bmatrix}$$

evaluated at the fixed point. By examining the eigenvalues of J we can determine whether small perturbations from (x^*, y^*) will decay back to the fixed point, fall away from the fixed point, or orbit the fixed point in a limit cycle [10, 9].

4 A Simple Inverter

4.1 Computational Characteristics

Several quantitative descriptions may be used to describe our chemical gates. This work will begin by exploring the transfer curve and propagation delay time of two chemical inverters. These properties address static and dynamic behaviors of our chemical circuits.

The transfer curve shows the output a circuit will generate for each input value. By examining how the output changes as a function of the input, we can see how sensitive the output is to fluctuations in the input. For error-resistant digital computation we would like the output to change values *only* when the input changes its digital value. Thus the output should exhibit low-gain when the input is near the digital values zero and one, and should have a high-gain transition region at some point in between. For an inverter, such an ideal characteristic is exemplified by a sharp sigmoid.

Propagation delay is the delay from the time a signal is applied to a gate until the appropriate change appears at the output [11]. We are interested in this delay time for both the high-to-low (t_{PHL}) and the low-to-high (t_{PLH}) output transitions. This transition is measured from the time when the input passes through half of the high concentration to the time when the output passes the same value. The total propagation delay is defined as the average of t_{PHL} and t_{PLH}. In our circuits the propagation delay will depend on a number of factors including reaction rate, reactor volume, efflux rate, and concentration of chemical species. In practice some of these parameters are an immutable result of the underlying technology, but some can be controlled by manipulating the environment.

4.2 A Simple Inverter

We begin by examining a simple computational device: the digital inverter. This inverter will be built from a single type of NOT gate (G) operating in an open reactor. The reactor will be supplied with a constant influx of gate and substrate molecules. In addition, input (I) to the gate will be supplied to provide an external drive. In a more complex circuit the input could be the output of another gate in the reactor. The output of this inverter is expressed by the concentration of product (P) cleaved from substrate (S). This reaction network is shown schematically in Figure 4.

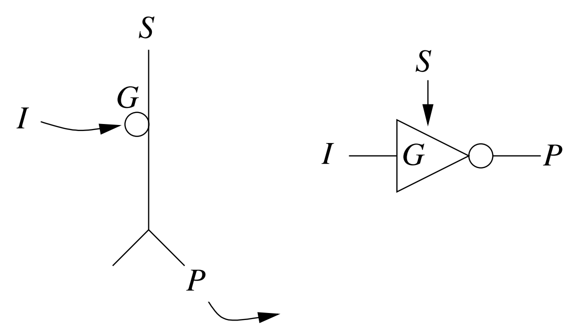

Figure 4: A schematic of the inverter reaction network. Left: The chemical gate. In the absence of input I, NOT gate G cleaves substrate S to produce product P. As the concentration of I rises in the reactor, more gates will become inhibited, and product concentration will fall. Right: Schematic of the corresponding electronic logic.

The behavior of the system can be modeled with the following system of rate equations:

$$\frac{dG}{dT} = \frac{G^m - EG(T)}{V} \tag{1}$$

$$\frac{dI}{dT} = \frac{I^m - EI(T)}{V} \tag{2}$$

$$\frac{dP}{dT} = \beta S(T)\max(0, G(T) - I(T)) - \frac{EP(T)}{V} \tag{3}$$

$$\frac{dS}{dt} = \frac{S^m}{V} - \beta S(T)\max(0, G(T) - I(T)) - \frac{ES(T)}{V} \tag{4}$$

where I^m, G^m, and S^m are the constant rates of *molar* influx of the respective chemical species,

V is the volume of the reactor, E is the rate of volume efflux, and β is the reaction rate constant. Recall that the max terms in (3)-(4) come from our assumption that the binding of input to gate molecule is both instantaneous and complete.

It is quite clear that this system can function as an inverter. If there are no inputs in the reactor, all of the gates will be active and produce product—this is the high signal. As input is added gates will become inhibited, and the product concentration will fall. As input concentration reaches gate concentration, all gates become inhibited and the product concentration falls to zero—this is the low signal.

4.3 Static Behavior

To explore the equilibrium behavior of our inverter we set the time derivatives in (1)-(4) to zero and obtain:

$$\begin{aligned} G &= \frac{G^m}{E} \\ I &= \frac{I^m}{E} \\ P &= \frac{V\beta S \max(0, G-I)}{E} \\ S &= \frac{VS^m - \beta S \max(0, G-I)}{E} \\ &= \frac{S^m}{(E + V\beta \max(0, G-I))} \end{aligned}$$

Requiring that input concentration never exceeds gate concentration allows us to simplify and eliminate the max functions. This produces the following relation between input concentration and output (product) concentration:

$$P = \frac{\beta S^m V(\frac{G^m}{E} - I)}{E^2 + \beta V E(\frac{G^m}{E} - I)} = \frac{\frac{S^m}{E}(\frac{G^m}{E} - I)}{\frac{E}{\beta V} + \frac{G^m}{E} - I}$$

Introducing rescaling parameters $\alpha = \frac{S^m}{E}$, $\gamma = \frac{G^m}{E}$, and $\delta = \frac{E}{\beta V}$ allows further simplification. Thus we arrive at the following equation for the static transfer curve of our chemical inverter:

$$P = \frac{\alpha(\gamma - I)}{\delta + (\gamma - I)}$$

This shows how the output concentration (P) changes as a function of the input concentration (I).

Further constraints must be introduced to create an inverter with well-defined signal levels. First, the concentration which is a high logic value is defined as H. Now we will require that $P = H$ when $I = 0$ and that $P = 0$ when $I = H$. These conditions yield the following constraints:

$$\gamma = H \quad \text{and} \quad \alpha = H + \delta$$

Thus, to alter the static transfer curve, we may vary the parameters δ and α, while maintaining the relationship: $\alpha = H + \delta$.

In our systems, both the reaction rate (β) and the reactor volume (V) are fixed as a result of the technology being used. This means that to alter the *static* transfer characteristics we must alter the efflux of the reactor, which will have an impact on the dynamics of the inverter as well.

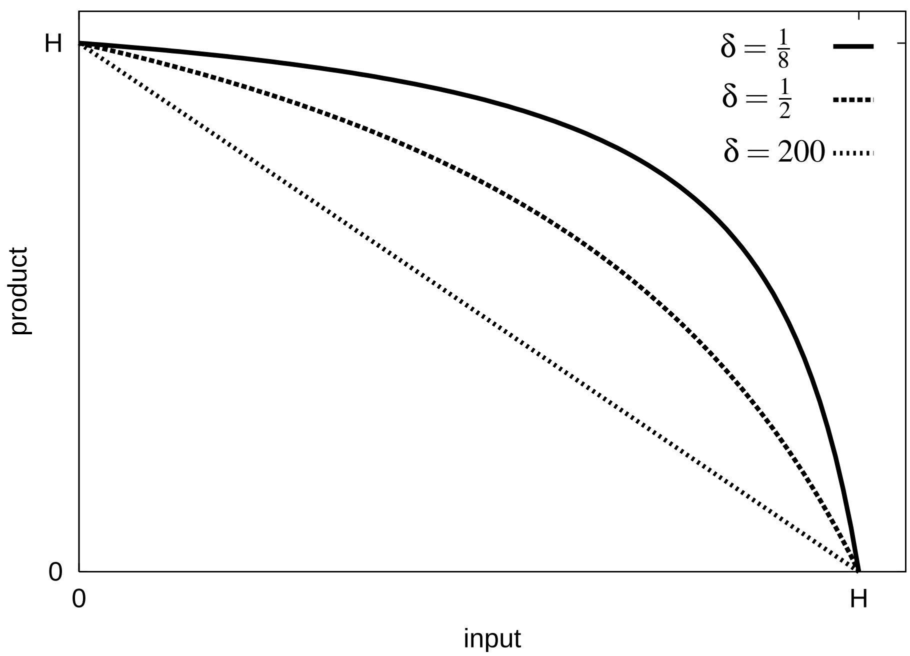

Figure 5: The static transfer curve for an inverter constructed from a NOT gate in an open system with several different values for δ: $\frac{1}{8}$, $\frac{1}{2}$, and 200. As δ increases the transfer curve approaches a straight line from high product and low input to low product and high input concentrations.

Figure 5 shows the transfer functions obtained by setting δ to a range of values. As δ is increased, the curve flattens out and becomes close to linear. As δ is decreased the curve becomes more bowed out (i.e., a large derivative).

Note that this transfer curve is far from the sigmoid shape ideal for digital computing. Any noise that moves the input concentration away from its digital value will propagate through to the output, possibly causing computational errors. The next chapter will introduce another inverter system that yields more favorable static transfer characteristics.

4.4 Dynamic Behavior

While the static behavior discussed in the previous section addresses equilibrium characteristics, this analysis neglects the dynamic behavior of the system. Ultimately, the static transfer characteristic depended on only one ratio, but the dynamic behavior is less restricted and depends on several variables. However, this solution space is narrowed by the physical restrictions of our technology discussed in Sections 2 and 3. We define a logical high value to be a concentration of 250 nM and calculate the rest of the parameters for the system.

Figure 6 shows the results of numerical integration of the inverter working under such conditions. The reactor was started with zero concentration of all chemical species. The propagation delay of the inverter is $\approx 7.9 \cdot 10^3$ s, with a $t_{PHL} \approx 12.5 \cdot 10^3$ s, and $t_{PLH} \approx 3.2 \cdot 10^3$ s.

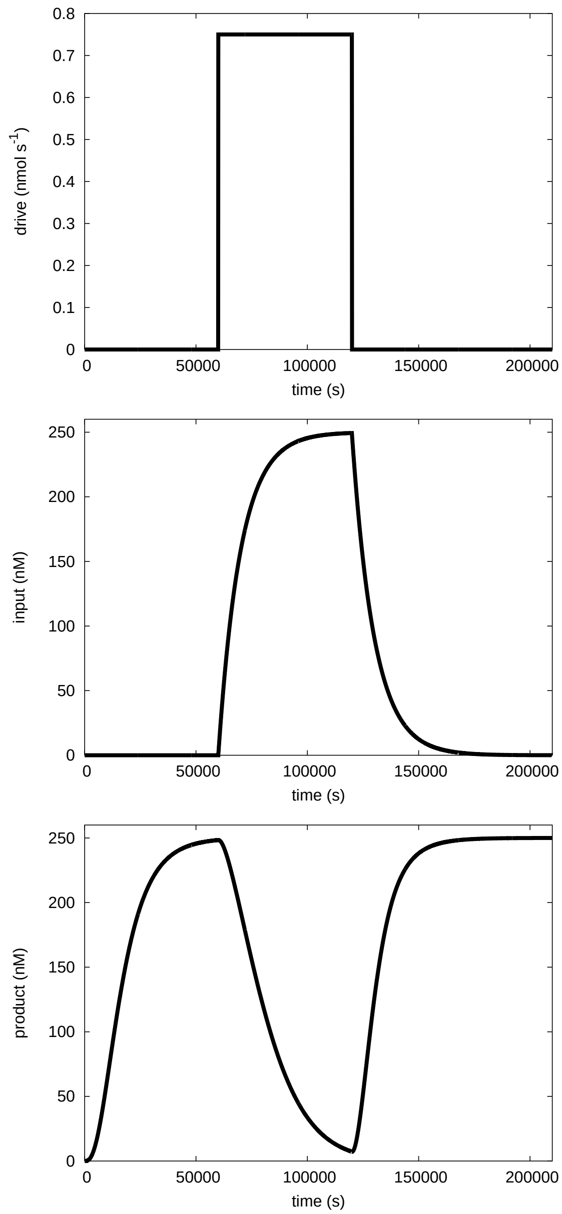

Figure 6: The top trace shows the external drive, in terms of molecular influx, which serves as input to the circuit. The middle trace shows the concentration of input I inside the reactor. The bottom trace shows the product concentration P which serves as the output of the inverter. The input concentration is moved from low, to high, and back to low at $6 \cdot 10^4$ s intervals.

5 An Improved Inverter

In an effort to improve the transfer characteristics of our chemical inverter we will expand our system to include several gates in a cascade. The network will be constructed such that multiple inputs are required to deactivate a single NOT gate. This introduces more non-linearity in the input/output relationship and creates a more favorable static transfer curve [12].

The reaction network constructed to meet these goals is shown schematically in Figure 7. It consists of a top-level AND gate that is active in the presence of two input molecules (I) to the inverter. When the top level AND gate is active it produces product molecules J. A second type of AND gate is present which, in the presence of two molecules of J, becomes active and produces product K. This product K is then used as an inhibitory input to a NOT gate, which prevents it from producing its product P, the output of the system. Thus, every NOT gate requires four inputs to become inhibited. Also note that the process of inputs inhibiting the catalytic NOT gate is no longer instantaneous, as several enzymatic reactions have to occur.

The dynamics of this system can be modeled with the following rate equations:

$$
\begin{aligned}
\frac{dG_1}{dT} &= \frac{G_1^m - EG_1(T)}{V}\\
\frac{dG_2}{dT} &= \frac{G_2^m - EG_2(T)}{V}\\
\frac{dG_3}{dT} &= \frac{G_3^m - EG_3(T)}{V}\\
\frac{dI}{dT} &= \frac{I^m - EI(T)}{V}\\
\frac{dJ}{dT} &= \beta_1 S_1(T)\min(G_1(T), \frac{I(T)^2}{4G_1(T)}) - \frac{EJ(T)}{V}\\
\frac{dK}{dT} &= \beta_2 S_2(T)\min(G_2(T), \frac{J(T)^2}{4G_2(T)}) - \frac{EK(T)}{V}\\
\frac{dP}{dT} &= \beta_3 S_3(T)\max(0, G_3(T) - K(T)) - \frac{EP(T)}{V}\\
\frac{dS_1}{dT} &= \frac{S_1^m}{V} - \beta_1 S_1(T)\min(G_1(T), \frac{I(T)^2}{4G_1(T)}) - \frac{ES_1(T)}{V}\\
\frac{dS_2}{dT} &= \frac{S_2^m}{V} - \beta_2 S_2(T)\min(G_2(T), \frac{J(T)^2}{4G_2(T)}) - \frac{ES_2(T)}{V}\\
\frac{dS_3}{dT} &= \frac{S_3^m}{V} - \beta_3 S_3(T)\max(0, G_3(T) - K(T)) - \frac{ES_3(T)}{V}
\end{aligned}
$$

where G_1^m, G_2^m, G_3^m, I^m, S_1^m, S_2^m, and S_3^m are the molecular influxes of the respective species, V is the volume of the reactor, and E is the volumetric efflux of the system.

5.1 Static Behavior

To explore the equilibrium behavior we set the time derivatives to zero and obtain:

$$
\begin{aligned}
G_1 &= \frac{G_1^m}{E} \\
G_2 &= \frac{G_2^m}{E} \\
G_3 &= \frac{G_3^m}{E} \\
I &= \frac{I^m}{E} \\
J &= \frac{V\beta_1 S_1 \min(G_1, \frac{I^2}{4G_1})}{E} \\
K &= \frac{V\beta_2 S_2 \min(G_2, \frac{J^2}{4G_2})}{E} \\
P &= \frac{V\beta_3 S_3 \max(0, G_3 - K)}{E} \\
S_1 &= \frac{S_1^m}{E + V\beta_1 \min(G_1, \frac{I^2}{4G_2})} \\
S_2 &= \frac{S_2^m}{E + V\beta_2 \min(G_2, \frac{J^2}{4G_2})} \\
S_3 &= \frac{S_3^m}{E + V\beta_3 \max(0, G_3 - K)}
\end{aligned}
$$

Now it is simply a matter of substitution to solve for P in terms of I, and we see that:

$$P(I) = \frac{V\beta_3 S_1^m \max(0, G_3 - K(I))}{E^2 + EV\beta_3 \max(0, G_3 - K(I))} \tag{5}$$

$$K(I) = \frac{V\beta_2 S_2^m \min(G_2, \frac{J(I)^2}{4G_2})}{E^2 + EV\beta_2 \min(G_2, \frac{J(I)^2}{4G_2})} \tag{6}$$

$$J(I) = \frac{V\beta_1 S_1^m \min(G_1, \frac{I^2}{gG_1})}{E^2 + EV\beta_1 \min(G_1, \frac{I^2}{4G_1})} \tag{7}$$

Thus, equation (5) describes the static transfer curve of this system in terms of $K(I)$ and $J(I)$. The shape and orientation of this curve may be adjusted by varying the gate and substrate concentrations. In order to make a valid comparison to the inverter of the previous chapter, we will use the same reaction rates, local high concentration, efflux rate, and reactor volume. We can now set the fixed parameters and explore the remainder of the parameter space to find values that produce favorable transfer characteristics. It was empirically determined that setting $G_1 = 500\,\mathrm{nM}$, $G_2 = 50\,\mathrm{nM}$, $G_3 = 100\,\mathrm{nM}$, $S_1^m = 2\,\mathrm{nmol\,s^{-1}}$, $S_2^m = 8\,\mathrm{nmol\,s^{-1}}$, and $S_3^m = 2.25\,\mathrm{nmol\,s^{-1}}$, yields the sigmoid-shaped curve shown in Figure 8. This curve exhibits desirable properties of a digital inverter: low-gain near the high and low input concentrations, and a high-gain transition region.

5.2 Dynamic Behavior

We now examine the dynamic behavior of the system described in the previous sections. The reactor is started with zero concentrations of all species. The system can now be numerically integrated, and the the results are shown in Figure 9.

In addition to having different transfer characteristics, the dynamics of this inverter system differ significantly from the single NOT inverter from the previous section. Adding two additional layers of enzymatic gates has the expected effect of increasing the propagation delay. In the system shown in Figure 9 the propagation delay is $\approx 2.7 \cdot 10^4$ s, with $t_{PLH} \approx 2.1 \cdot 10^4$ s and $t_{PLH} \approx 3.3 \cdot 10^4$ s.

The two inverters illustrate a trade-off between favorable transfer characteristics and shorter propagation delay time. Thus we have some flexibility when engineering these elements into particular circuits.

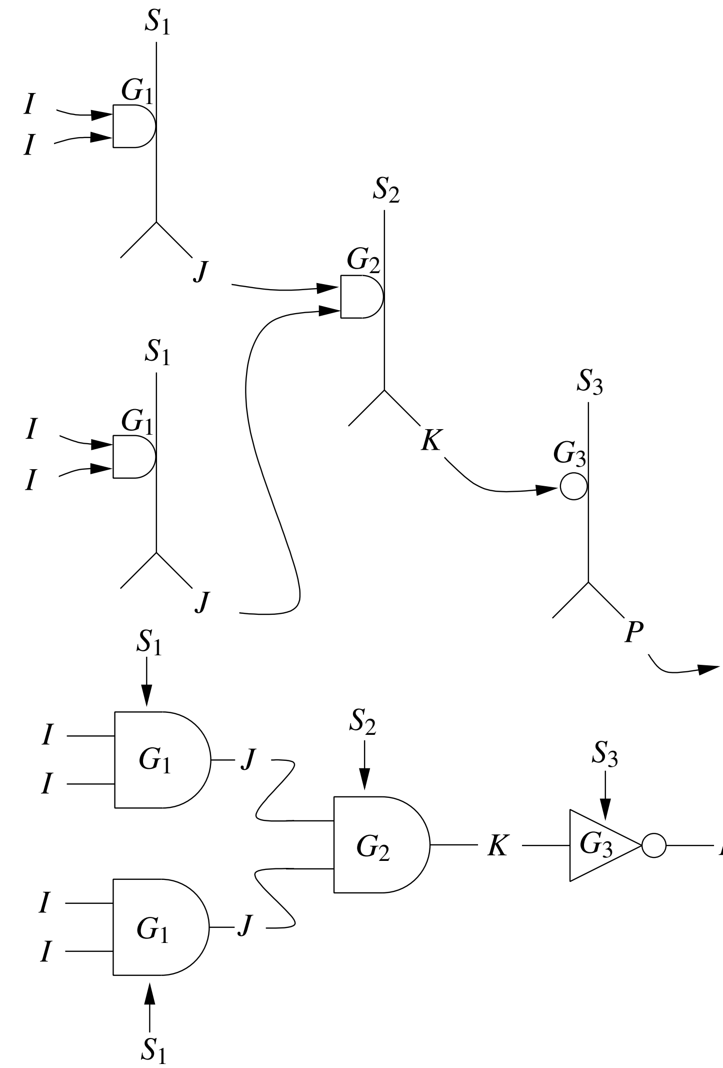

Figure 7: Schematic of the inverter reaction network. Top: The chemical gates. In the presence of two molecules of I, the AND gate G_1 becomes active and cleaves substrate S_1 to produce product J. In the presence of two molecules of J, the AND gate G_2 becomes active and cleaves substrate S_2 to produce product K. In the absence of the inhibitory input K, NOT gate G_3 cleaves substrate S_3 to produce P, the output of the inverter. Bottom: Schematic of the corresponding electronic logic.

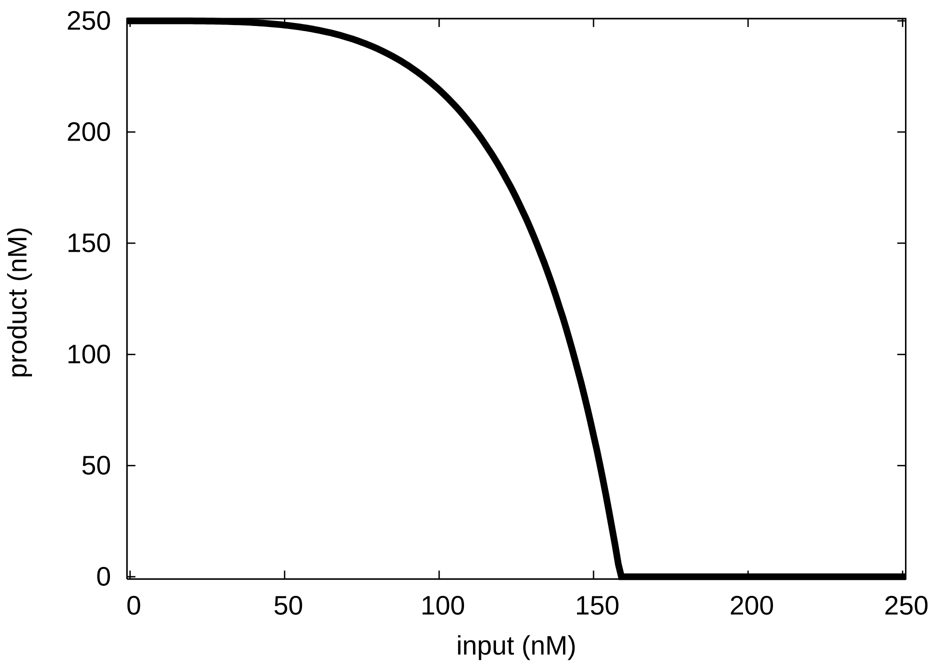

Figure 8: The transfer curve for our improved inverter. Note the sigmoid-type shape and complete resistance to error through much of the high input range. The low input range shows good error resistance as well.

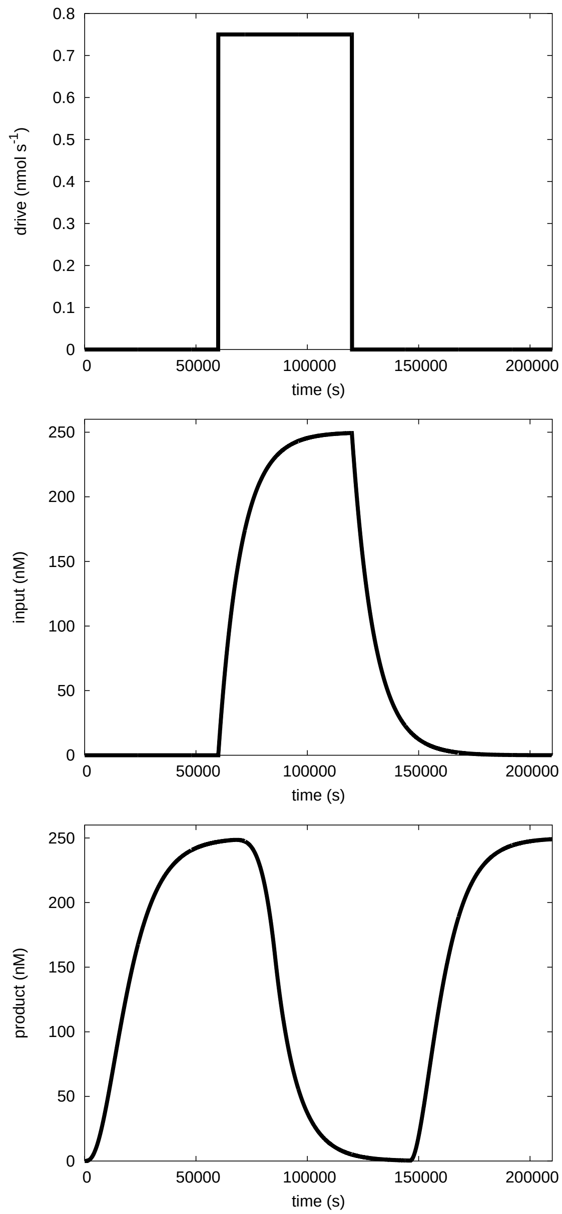

Figure 9: The top trace shows the external drive, in terms of molecular influx, which serves as input to the circuit. The middle trace shows the concentration of input I inside the reactor. The bottom trace shows the product concentration P which serves as the output of the inverter. The input concentration is moved from low, to high, and back to low at $6 \cdot 10^4$ s intervals.

6 A Chemical Flip-Flop

The inverters designed in the preceding two sections have been feed-forward circuits. Moving to an open system allows recurrent circuits to be constructed. We will now examine more complex networks of chemical reactions that make use of feedback.

Using feedback, we create a bistable system that can be used for one bit of memory—a chemical flip-flop. The flip-flop must exhibit three types of behavior: hold, set, and reset. In the hold behavior, there should be two stable concentration states, one of which represents a high output of the system, and the other represents a low output. In the set behavior, there should exist a single stable high state such that, regardless of previous state, the flip-flop will go to the high state. Similarly, the reset behavior should take the flip-flop to the low state.

A system that functions as a flip-flop can be constructed with a network of two NOT gates connected in a cycle of inhibition. This network is shown schematically in Figure 10. Gate G_1 cleaves substrate S_1 to produce product P_1, which inhibits the catalytic activity of gate G_2. Gate G_2 cleaves S_2 to produce P_2, which inhibits G_1 to complete the cycle. As with the inverters of the previous two sections, these reactions take place in an open system, where chemical species are being supplied and removed.

Output from the flip-flop is in terms of the concentration of the cleaved product P_2: a high concentration represents a logic value of one, a low concentration represents a logic value of zero. Control of the flip-flop is accomplished by varying the substrate concentrations inside the reactor. A circuit consisting of a single flip-flop may be controlled by varying the influx of substrate to the reactor. However, in a more complex circuit, these substrates would be the output of other gates.

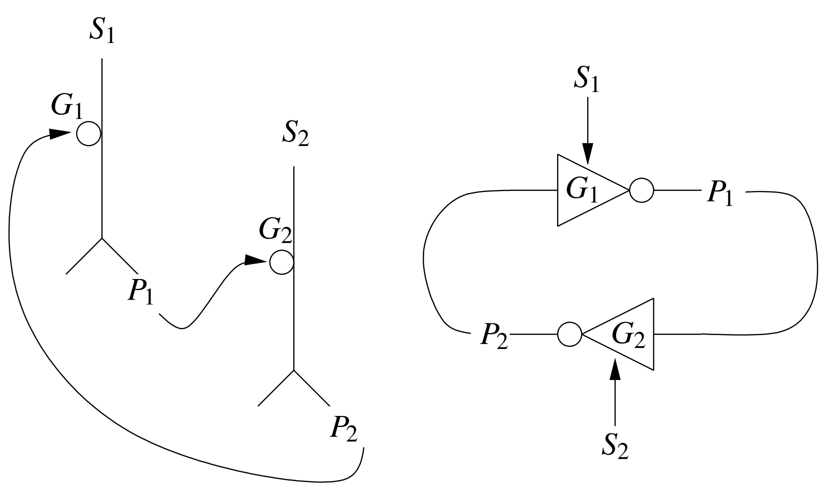

Figure 10: Schematic of the flip-flop reaction network. Left: The chemical reaction network. Gate G_1 cleaves substrate S_1 to produce P_1. Gate G_2 cleaves substrate S_2 to produce P_2. Each product serves as an inhibitory input to the opposite gate. Right: Schematic of the corresponding electronic circuit.

These reactions will proceed in an open reactor with four solutions flowing in, and one outflow to maintain constant volume. Solutions containing gates 1 and 2 will be continuously supplied. Additionally, substrates 1 and 2 will be variably supplied to provide an external control mechanism for the flip-flop. We define constants G_1^i and G_2^i to be the rates of influx of gate solutions. The concentration of species within each input solution is given by G_1^c and G_2^c. Now we may define the molecular influx of the gates as $G_1^m = G_1^i G_1^c$ and $G_2^m = G_2^i G_2^c$. The external control mechanism will be modeled by the functions $S_1^m(T)$ and $S_2^m(T)$, which describe the variable *molecular* influx of substrates at time T. Our model will assume a constant *volumetric* influx of substrate solutions given by S_1^i and S_2^i. Thus the rate of efflux of the system at time T is given by:

$$E = G_1^i + G_2^i + S_1^i + S_2^i$$

We now define $P_1(T)$, $P_2(T)$, $S_1(T)$, $S_2(T)$, $G_1(T)$, and $G_2(T)$ to be the concentrations within the reactor at time T of product 1, product 2, substrate 1, substrate 2, gate 1, and gate 2, respectively. The system dynamics are modeled by the following system of six ordinary differential equations:

$$\begin{aligned}
\frac{dG_1}{dT} &= \frac{G_1^m - EG_1(T)}{V} \\
\frac{dG_2}{dT} &= \frac{G_2^m - EG_2(T)}{V} \\
\frac{dP_1}{dT} &= \beta_1 S_1(T)\max(0, G_1(T) - P_2(T)) - \frac{EP_1(T)}{V} \\
\frac{dP_2}{dT} &= \beta_2 S_2(T)\max(0, G_2(T) - P_1(T)) - \frac{EP_2(T)}{V} \\
\frac{dS_1}{dT} &= \frac{S_1^m(T)}{V} - \beta_1 S_1(T)\max(0, G_1(T) - P_2(T)) - \frac{ES_1(T)}{V} \\
\frac{dS_2}{dT} &= \frac{S_2^m(T)}{V} - \beta_2 S_2(T)\max(0, G_2(T) - P_1(T)) - \frac{ES_2(T)}{V}
\end{aligned}$$

where β_1 and β_2 are the reaction rate constants and V is the volume of the reactor.

6.1 Analyzing the model

Now we can examine this model to determine the conditions under which it will function as a flip-flop. In order to simplify the system, the dynamic variables representing gate concentration are treated as constant—they will maintain a steady state that depends only on the efflux of the system. Substrate concentrations will be used to control the flip-flop, and we must determine how the flip-flop behaves (in terms of its outputs) in response to substrate concentrations. Hence we view the substrate concentrations as parameters of the system rather than dynamic variables. Now the behavior of the flip-flop can be understood using the following two-dimensional system:

$$\frac{dp_1}{dt} = r_1 \max(0, g_1 - p_2(t)) - kp_1(t) \tag{8}$$

$$\frac{dp_2}{dt} = r_2 \max(0, g_2 - p_1(t)) - kp_2(t) \tag{9}$$

where p_1 and p_2 are the product concentrations, r_1 and r_2 are lumped parameters representing substrate concentration and the reaction rate constant, g_1 and g_2 are the gate concentrations, and k is the decay constant.

In order to function as a flip-flop, the system must have two stable states: a *set* state with a high value of p_2, and a *reset* state with a low value of p_2. Additionally there should exist control mechanisms to switch between states, in this case by varying the substrate concentrations. High concentration of both substrates is used to hold the flip-flop state, while the absence of one is used to set or reset the flip-flop. We now examine how the system will behave for various values of the parameters r_1 and r_2.

We begin our analysis by examining the nullclines of the system. Setting equations (8) and (9) to zero yields:

$$\frac{dp_1}{dt} = 0 \quad \Rightarrow \quad p_1 = \frac{r_1 \max(0, g_1 - p_2)}{k}$$
$$\frac{dp_2}{dt} = 0 \quad \Rightarrow \quad p_2 = \frac{r_2 \max(0, g_2 - p_1)}{k}$$

The derivative on one of the nullclines will have components in a single direction, along either the p_1 or p_2 axis. This observation can be further qualified by inspecting equations (8)-(9). Notice that as we increase the value of p_1 the value of $\frac{dp_1}{dt}$ decreases, and as we decrease the value of p_1 the value of $\frac{dp_1}{dt}$ increases. This same relationship holds between p_2 and $\frac{dp_2}{dt}$. Thus a point above the p_1 nullcline will have a derivative in the negative p_1 direction and a point below the p_1 nullcline will have a derivative in the positive p_1 direction. We can use these facts to partition the phase space into regions where the signs of the derivatives are known [10].

Figure 11 shows two possible configurations of the nullclines along with the signs of the derivatives in different regions of the phase space. In the case shown in Figure 11**a** the two nullclines intersect (to create fixed-points) at points A, B, and C. The stability of these points can be investigated by examining the slopes of the surrounding regions. The intersection at point A has four adjacent regions. Two of the regions adjacent to A have derivatives pointing toward A, but two of the regions have derivatives away from A. This reveals that A is partly stable: perturbations in one direction will return back to A, while perturbations in another direction will fall away from A. Points B and C are each surrounded by regions with derivatives pointing back to them. A small perturbation will return to the original state. Hence, B and C are stable nodes.

The separatrix shown in Figure 11**a** divides the phase space into two basins of attraction. All trajectories starting above the separatrix will be attracted to point B; all trajectories starting below the separatrix will be attracted to point C. This configuration represents a hold state of the flip-flop as the system will evolve to either point B or point C. Point B is the low state of the flip-flop;the system contains a low concentration of p_2 and a high concentration of p_1. Point C is the high state of the flip-flop; the system contains a high concentration of p_2 and a low concentration of p_1.

If the nullclines do not intersect in the positive quadrant then there will be a single intersection on either the p_1 or the p_2 axis. Such a situation is shown in Figure 11**b**. The point F is a stable node with a low p_2 value and a high p_1 value. An examination of the three regions of

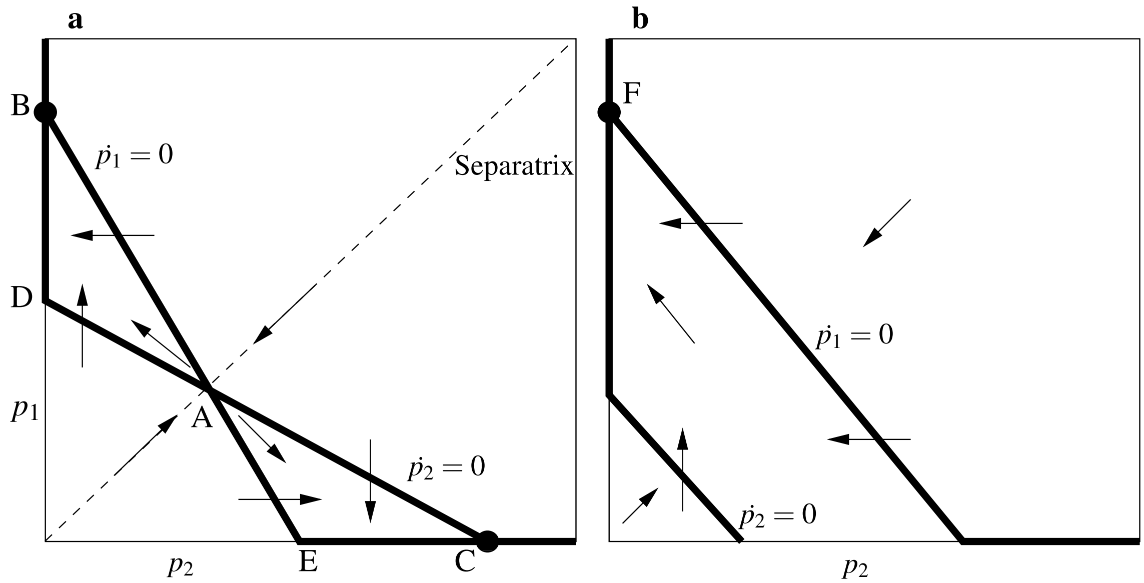

Figure 11: The geometric structure of the flip-flop equations. Dark lines indicate the nullclines, where p_1 or p_2 is unchanging. The arrows indicate the sign of the derivatives at various regions in the phase space. **a** The hold state: a bistable system with balanced substrate concentrations. Points B and C are the stable points of the system. Point A is an unstable saddle node. **b** The reset state: a mono-stable system caused by a low concentration of substrate 2. The nullclines intersect once at point F, which all locations will be attracted to.

the phase space reveals that all points will be attracted to point F. Thus this state will reset the flip-flop.

For bistability to occur the two nullclines must intersect in the orientation shown in Figure 11**a**. This requires that point B be above point D and that point C be to the right of point E, and translates into the following conditions: $kg_2 < r_1 g_1$ and $kg_1 < r_2 g_2$. If we wish to maintain symmetry, we can set $g_1 = g_2$ and $r_1 = r_2 = r$ and the conditions become $k < r$. When this condition is met the flip-flop will be in a hold state.

The flip-flop is set or reset when the two nullclines do not intersect in the positive quadrant. Then there will exist one stable fixed point at their intersection (point F in Figure 11**b**). This will be the case if either r_1 or r_2 takes the value of zero.

Suppose that the flip-flop is in the hold state with a high concentration of p_2 at point C in Figure 11**a**. As the parameter r_2 is reduced, the slope of the $\frac{dp_2}{dt}$ nullcline will decrease and the stable point C will slide down the p_2 axis toward point E. As C passes E the fixed point is annihilated, and the system will jump to the only fixed point: point F in Figure 11**b**.

After the system has gone to point F the parameter r_2 can be returned to its original value and the flip-flop will remain in its low state. This lack of reversibility as the parameter is varied is called *hysteresis* in dynamical systems [10].

6.1.1 Stability analysis

The stability of the fixed points may also be analyzed by examining the value of the Jacobian. Recall that the Jacobian captures the nature of small perturbations from a fixed point via the derivative of the perturbations. In our case, this technique cannot directly be applied to phase space points on the axes, because our system is only defined in the first quadrant, and not differentiable on the boundaries. However, we can use the Jacobian to examine the fixed point which occurs in the positive quadrant of the phase space.

Equations 8 and 9 produce the Jacobian matrix:

$$\mathrm{J} = \begin{bmatrix} -k & -r_2 \\ -r_1 & -k \end{bmatrix}$$

The eigenvalues of J are $\lambda_1 = -k - \sqrt{r_1 r_2}$ and $\lambda_1 = -k + \sqrt{r_1 r_2}$. If both of the eigenvalues are negative then the fixed-point will be stable [10]. Thus when $k^2 > r_1 r_2$ all non-zero fixed points will be stable nodes, otherwise they will be unstable. This produces the same restriction for bistability as did the analysis of the nullclines, namely that $k^2 < r_1 r_2$. Additionally, this analysis provides an exact classification of the unstable point A of the nullclines intersection: a saddle node.

6.1.2 A qualitative view of the constraints

The stability analysis has revealed constraints needed for a this system to function as a flip-flop. First, in order to maintain output symmetry we required that $g_1 = g_2$. This ensures that both the output of the flip-flop (p_2) and the negated output (p_1) have the same value for a logical high. Next, in order to maintain a symmetric separatrix along the line $p_1 = p_2$ we require that $r_1 = r_2 = r$. Thus, during the hold state of the flip-flop, the phase-space is equally divided between values which are attracted to the high state and values which are attracted to the low state.

The constraints for bistability is then $k < r$. This means that the rate at which concentrations decay in the reactor (by means of outflow) must be less than the rate at which the enzymatic gates can create product. In other words, the gates must be able to create product faster than it is being removed.

6.2 Designing a flip-flop

We are now in a position to convert these constraints on the dimensionless model to specifications for a physical system. We will use the reaction rate and reactor volume described in Sections 2 and 3. We assume that the initial concentration of all species in the reactor is zero. These conditions, and the constraints introduced from the analysis are enough to fully describe the physical system.

Figure 12 shows a numerical integration of the system over a period of $9.6 \cdot 10^4$ s. At $2.4 \cdot 10^4$ s intervals the system is moved between set, hold, reset, and hold operations.

The maximum switching speed of this flip-flop can be empirically determined by rapidly moving between set and reset states. As the switching speed is increased beyond some threshold, the flip-flop can no longer respond quickly enough. For the flip-flop designed in this

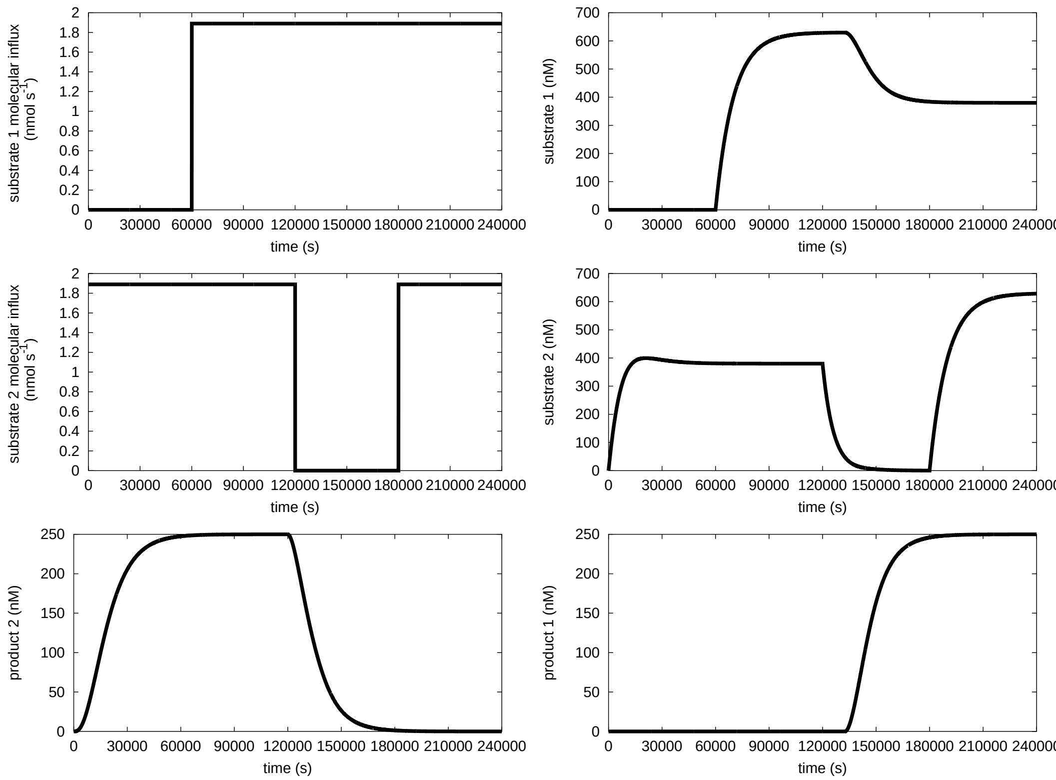

Figure 12: Exercising control over the flip-flop system. The top two traces on the left show substrate molecular influx rates. The top two traces on the right show the corresponding concentrations of substrates. The bottom two traces on the show concentrations of products 2 and 1. These represent the output and the negated output of the system respectively. The system is moved from set, to hold, to reset, and finally to hold states. This control is accomplished by controlling the influx of substrate.

section, we determined the maximum switching speed to be about $3.33\cdot 10^{-}5$ Hz, corresponding to a period of $3\cdot 10^{4}$ s. A numerical simulation of the flip-flop running at this speed is shown in Figure 13.

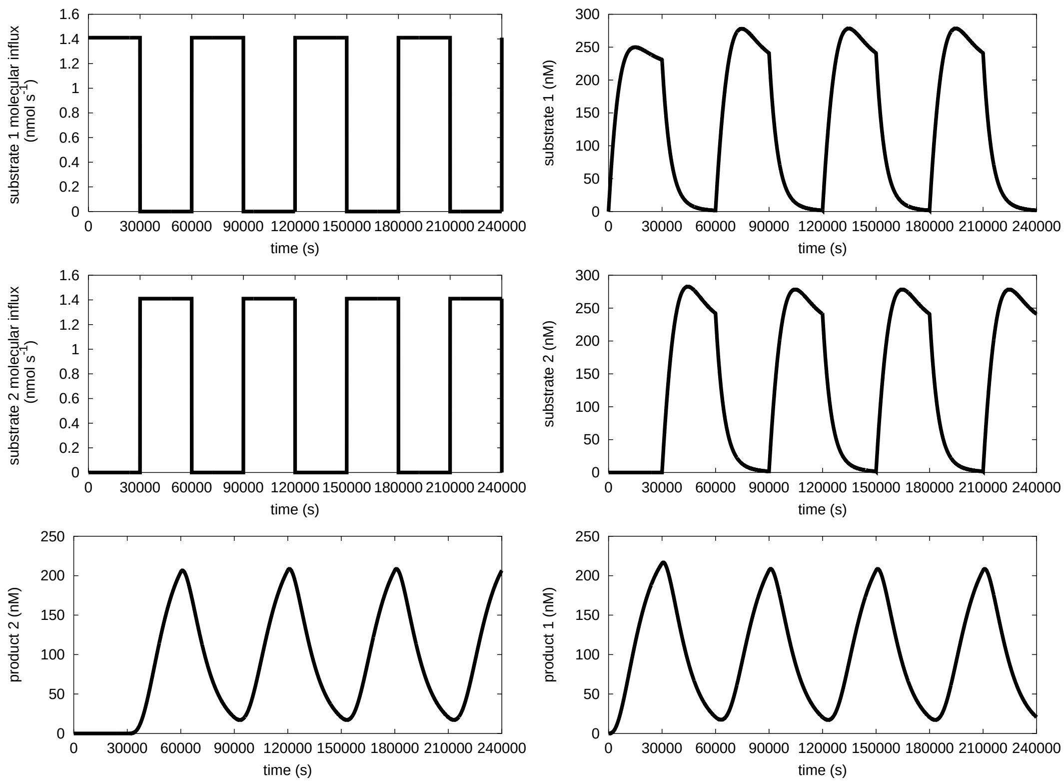

Figure 13: Flip-flop switching speed. The flip-flop is moved repeatedly from set to reset states at its maximum switching speed. The top two traces on the left show substrate molecular influx rates. The top two traces on the right show the corresponding concentrations of substrates in the reactor. The bottom two traces show concentrations of products 2 and 1. These represent the output and the negated output of the system, respectively.

7 A Chemical Oscillator

Oscillating reactions are a recent curiosity in chemistry. Such periodic behavior was once thought impossible, but in recent decades the mechanisms behind chemical oscillations have become established [9]. However, the design of chemical oscillators remains something of a black art. Recently, Elowitz and Leibler created oscillations using a network of transcriptional regulators in *E. coli* [13]. To design their system, dubbed the repressilator, they used techniques from dynamical systems similar to those used herein.

By using composable enzymatic gates we can systematically create a chemical oscillator. A network constructed from n NOT gates connected in a cycle can give rise to oscillatory behavior when n is an odd number greater than one. We will explore the simplest case where $n = 3$. The reaction network consists of three NOT gates connected in a cycle of inhibition, operating in an open reactor. This network is shown schematically in Figure 14.

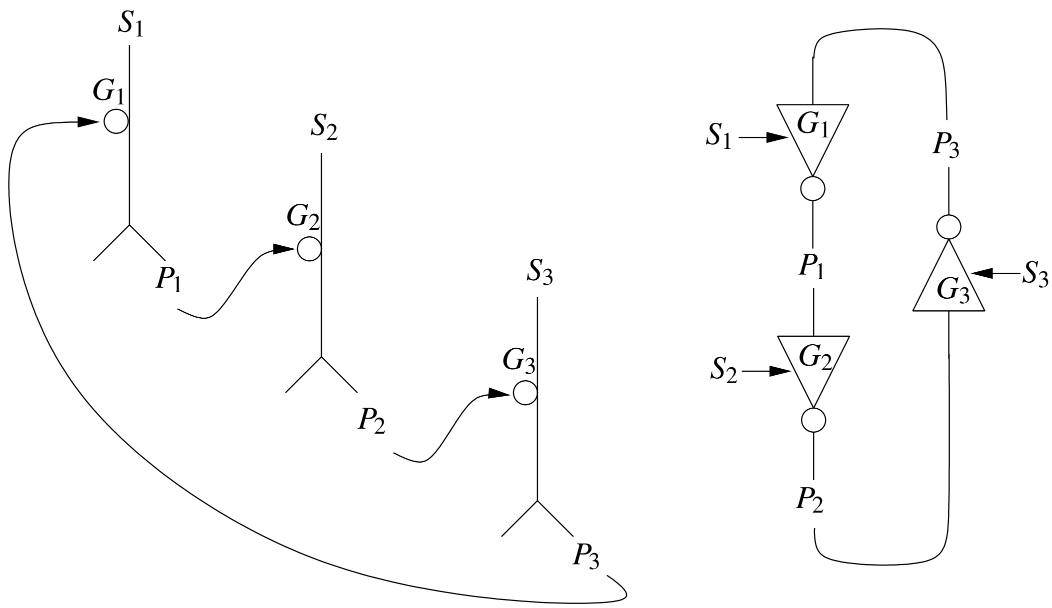

Figure 14: Schematic of the oscillator reaction network. Left: The chemical gates. Gate G_1 cleaves substrate S_1 to product P_1. Gate G_2 cleaves substrate S_2 to produce P_2. Gate G_3 cleaves substrate to produce P_3. The three NOT gates are connected in a cycle of inhibition. Right: Schematic of the corresponding electronic circuit.

The reactor will have six solutions flowing in, and one outflow to maintain constant volume. Solutions containing gates 1, 2, and 3, and substrates 1, 2, and 3, will be continuously supplied to the reactor. As before, we define constants G_1^i, G_2^i, and G_3^i to be the influx of gate solution. The concentration of gate molecules within each solution is given by G_1^c, G_2^c, and G_3^c. The influx and concentration of substrates are given by S_1^i, S_2^i, S_3^i, S_1^c, S_2^c, and S_3^c.

This yields a rate of efflux of:

$$E = G_1^i + G_2^i + G_3^i + S_1^i + S_2^i + S_3^i$$

We define $P_1(T)$, $P_2(T)$, $P_3(T)$, $S_1(T)$, $S_2(T)$, $S_3(T)$, $G_1(T)$, $G_2(T)$ and $G_3(T)$ to be the concentrations within the reactor at time T of the products, substrates and gates. The system dynamics may be described by the following system of nine ordinary differential equations:

$$\frac{dG_1}{dT} = \frac{G_1^i G_1^c - EG1(T)}{V} \tag{10}$$

$$\frac{dG_2}{dT} = \frac{G_2^i G_2^c - EG2(T)}{V} \tag{11}$$

$$\frac{dG_3}{dT} = \frac{G_3^i G_3^c - EG3(T)}{V} \tag{12}$$

$$\frac{dP_1}{dT} = \beta_1 S_1(T)\max(0, G_1(T) - P_3(T)) - \frac{EP1_(T)}{V} \tag{13}$$

$$\frac{dP_2}{dT} = \beta_2 S_2(T)\max(0, G_2(T) - P_1(T)) - \frac{EP_2(T)}{V} \tag{14}$$

$$\frac{dP_3}{dT} = \beta_3 S_3(T)\max(0, G_3(T) - P_2(T)) - \frac{EP_3(T)}{V} \tag{15}$$

$$\frac{dS_1}{dT} = \frac{S_1^i S_1^c}{V} - \beta_1 S_1(T)\max(0, G_1(T) - P_3(T)) - \frac{ES_1(T)}{V} \tag{16}$$

$$\frac{dS_2}{dT} = \frac{S_2^i S_2^c}{V} - \beta_2 S_2(T)\max(0, G_2(T) - P_1(T)) - \frac{ES_2(T)}{V} \tag{17}$$

$$\frac{dS_3}{dT} = \frac{S_3^i S_3^c}{V} - \beta_3 S_3(T)\max(0, G_3(T) - P_2(T)) - \frac{ES_3(T)}{V} \tag{18}$$

where β_1, β_2, and β_3 are the reaction rate constants, and V is the volume of the reactor.

7.1 Analyzing the model

Now we will examine this model to determine the conditions under which it will function as an oscillator. As with the flip-flop, we may treat the gate concentration as a constant. Although substrate concentrations will oscillate, they will always be much higher than the concentrations of the products, and this will allow us to view the substrate concentrations as essentially constant and further simplify the model. Thus, the behavior of this network can be understood using the following dimensionless rate equations:

$$\frac{dp_1}{dt} = r_1 \max(0, g_1 - p_3(t)) - kp_1(t) \tag{19}$$

$$\frac{dp_2}{dt} = r_2 \max(0, g_2 - p_1(t)) - kp_2(t) \tag{20}$$

$$\frac{dp_3}{dt} = r_3 \max(0, g_3 - p_2(t)) - kp_3(t) \tag{21}$$

where p_1, p_2, and p_3 are the product concentrations, g_1, g_2, and g_3 are the gate concentrations, r_1, r_2, and r_3 are lumped substrate and reaction rate terms, and k is the decay constant.

These three nullclines are planes in the positive quadrant which intersect to form a fixed point. We begin our analysis by examining the parameters' effect on the stability of this fixed point. Equations (19)–(21) produce the following Jacobian matrix:

$$\mathrm{J} = \begin{bmatrix} -k & 0 & -r_1 \\ -r_2 & -k & 0 \\ 0 & -r_3 & -k \end{bmatrix}$$

The eigenvalues of J consist of two complex conjugates, and one negative real:

$$\begin{aligned} \lambda_1 &= -k + \frac{1}{2}(1 - i\sqrt{3})\sqrt[3]{r_1 r_2 r_3} \\ \lambda_2 &= -k + \frac{1}{2}(1 + i\sqrt{3})\sqrt[3]{r_1 r_2 r_3} \\ \lambda_3 &= -k - \sqrt[3]{r_1 r_2 r_3} \end{aligned}$$

If the real parts of all eigenvalues are negative, then any perturbation from a fixed point will decay; the fixed point will be a stable node. Thus when $k > \frac{\sqrt[3]{r_1 r_2 r_3}}{2}$ we will have a single stable fixed point. Otherwise the fixed point will be the center of a limit cycle. When the stable node turns into an expanding limit cycle the systems undergoes a so-called *supercritical Hopf* bifurcation [10].

The analysis above has quantitatively determined the circumstances under which oscillations will occur and given a qualitative description of how variation of parameters affects the limit cycle. However, when designing computational devices which use an oscillator we would like more quantitative information about its behavior. Namely, we would like to know the shape, center, and period of the oscillations and how these characteristics are related to the parameters of the system.

Toward this end, we simplify the system so that it may be explicitly solved. Observe that when gate concentration exceeds product concentration the system evolves according to purely linear terms. The result of this approximation is a linear system that may be solved explicitly. This solution is shown in Appendix A.

The result of the solution to the linear system is a sinusoidal limit cycle with a period of $\frac{4\pi}{\sqrt{3}r}$ and a center of $p_1 = p_2 = p_3 = \frac{gr}{k+r}$. These are only good approximations when the system spends a significant amount of time in the range of linear operation. As the non-linearity of the system is increased, more error is introduced into these approximations and the oscillations are no longer sinusoidal in shape. Rather, we get more abrupt, saw-tooth type, oscillations.

7.2 Designing an oscillator

The constraints for oscillations and the linear approximation can be used to systematically design an oscillator. We aim for oscillations of product centered at $1\,\mu$M with a period of $6 \cdot 10^3$ s. We will use the reaction rate and reactor volume from Sections 2 and 3. The linear approximation for the period, and the fixed reaction rate allow the calculation of the necessary substrate concentration: $S = \frac{4\pi}{3\sqrt{3}}\,\mu$M. The condition for oscillations ($k < \frac{r}{2}$) and fixed reactor volume put an upper bound on the efflux rate: $E < \frac{\pi}{\sqrt{3}} \cdot 10^{-5}\,\mathrm{m^3 s^{-1}}$. The approximation for center allows the calculation of the necessary gate concentration: $G = 1.5\,\mu\mathrm{M}$. We will start the reactor off with a $1\,\mu\mathrm{M}$ concentration of product 1, and zero concentration for the rest of the species.

The system from equations (10)-(18) can now be fully specified. Figure 15 shows this system being simulated numerically. The oscillations have a period of about $4.8 \cdot 10^3$ s and a center of about $1.2\,\mu\mathrm{M}$. Thus, the actual oscillator had a period and center that both differed by about 20% from our linear approximation. This error may be too high for some applications, in which case the approximations could be used as a starting point from which the non-linear equations could be further refined to produce the desired behavior.

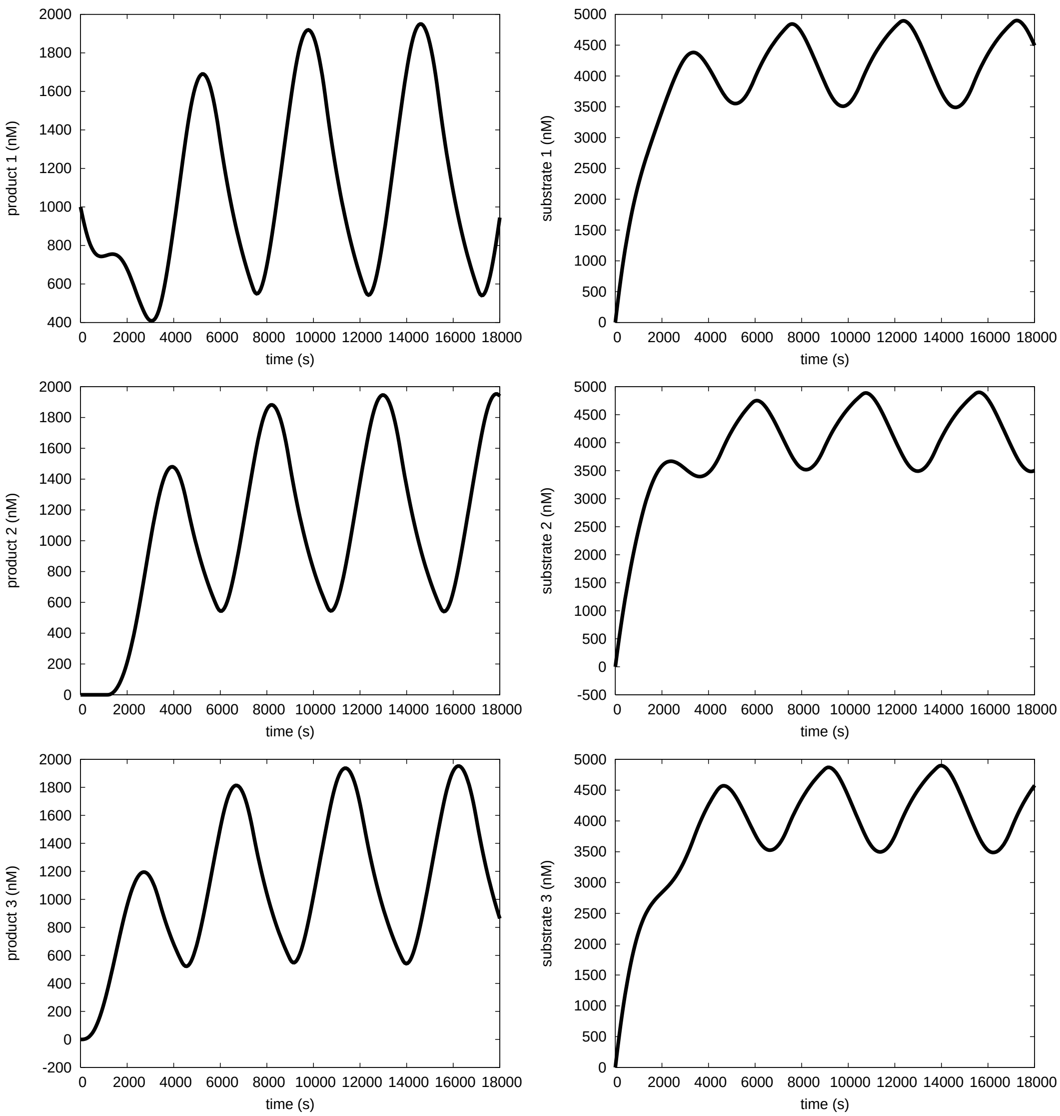

Figure 15: Numerical integration of oscillator system. The traces on the left show, from top to bottom, the concentrations of products 1, 2, and 3. The traces on the right show, from top to bottom, the concentrations of substrates 1, 2, and 3.

8 Conclusions

Deoxyribozyme based logic gates can be used to construct a biomolecular computer. Circuits are constructed from primitive boolean gates in a similar manner as in electronics. Deploying these gates in a closed reactor places a limit on the computations which may be performed. By moving to open reactors, we increase the computational abilities of the underlying logic gates. However, this move introduces complexities into the engineering task.

Techniques from dynamical systems offer qualitative and quantitative insights about the behavior of chemical networks in open reactors. Using these techniques, we have designed three fundamental components of a biomolecular computer: a single-bit inverter, a flip-flop that provides a single-bit of memory , and an oscillator that provides a clocking signal. By using feedback, the flip-flop and oscillator circuits take advantage of the homogeneous nature of our chemistry. The construction and demonstration of systems that make use of feedback between elements is on the frontier of deoxyribozyme-based computation.

Compared to electronic computers, this technology is extremely slow; components operate at rates measured in millihertz rather than in gigahertz as with modern electronic computers. However, the possibility of performing amorphous computations inside living cells offers advantages over the conventional electronic approach to computation.

Future work will involve constructing these systems in the laboratory to test the validity of the mathematical models, and the feasibility of implementation. After the mechanics of these simple systems are well understood and verified we shall construct more complicated systems from these elemental components. We shall also explore un-stirred, heterogeneous reactors where spatiotemporal dynamics perform computational functions in a manner similar to that proposed by Hiratsuka [8].

References

[1] L. M. Adleman, "Molecular computation of solutions to combinatorial problems," *Science*, vol. 266, pp. 1021–1024, Nov. 1994.

[2] J. Hartmanis, "On the weight of computation," *Bulletin of the EATCS*, vol. 55, pp. 136–138, 1995.

[3] 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, Nov. 2001.

[4] M. N. Stojanovic, T. E. Mitchell, and D. Stefanovic, "Deoxyribozyme-based logic gates," *Journal of the American Chemical Society*, vol. 124, pp. 3555–3561, Apr. 2002.

[5] M. N. Stojanovic, D. B. Nikic, and D. Stefanovic, "Implicit-OR tiling of deoxyribozymes: Construction of molecular scale OR, NAND, and four-input logic gates," *Journal of the Serbian Chemical Society*, vol. 68, no. 4–5, pp. 321–326, 2003.

[6] M. N. Stojanovic and D. Stefanovic, "Deoxyribozyme-based half adder," *Journal of the American Chemical Society*, vol. 125, pp. 6673–6676, May 2003.

[7] M. N. Stojanovic and D. Stefanovic, "A deoxyribozyme-based molecular automaton," *Nature Biotechnology*, vol. 21, pp. 1069–1074, September 2003.

[8] M. Hiratsuka, T. Aoki, and T. Higuchi, "Enzyme transistor circuits for reaction-diffusion computing," *IEEE Transactions on Circuits and systems—I: Fundamental Theory and Applications*, vol. 46, pp. 294–303, Feb. 1999.

[9] I. R. Epstein and J. A. Pojman, *An Introduction to Nonlinear Chemical Dynamics*. New York: Oxford University Press, 1998.

[10] S. H. Strogatz, *Nonlinear Dynamics and Chaos (With Applications to Physics, Biology, Chemistry, and Engineering*. Addison-Wesley, 1994.

[11] V. H. Grinich and H. G. Jackson, *Introduction to Integrated Circuits*. McGraw-Hill, 1975.

[12] T. F. Knight, Jr. and G. J. Sussman, "Cellular gate technology," in *Proceedings UMC98, First International Conference on Unconventional Models of Computation*, 1998.

[13] M. Elowitz and S. Leibler, "A synthetic oscillatory network of transcriptional regulators," *Nature*, vol. 403, pp. 335–338, Jan. 2000.

A Solution of the Linear Oscillator Approximation

This appendix describes an approximate model of our chemical oscillator. Whereas it does not exactly reflect the behavior of the system, it may allow us to make some quantitative approximations. The system consists of three NOT gates that are networked in a cycle of inhibition and reacting in an open system. This models the operation of the oscillator when product concentration is less than gate concentration, and this has allowed us to drop the max functions.

$$\dot{p}_1(t) = r_1(g_1 - p_3(t)) - kp_1(t) \tag{22}$$
$$\dot{p}_2(t) = r_2(g_2 - p_1(t)) - kp_2(t) \tag{23}$$
$$\dot{p}_3(t) = r_3(g_3 - p_2(t)) - kp_3(t) \tag{24}$$

where p_1, p_2, and p_3 are the product concentrations, g_1, g_2, and g_3 are the gate concentrations, r_1, r_2, and r_3 are lumped substrate and reaction rate terms, and k is the decay constant. We now have a non-homogeneous three-dimensional system of linear first-order differential equations. Such a system can be solved explicitly.

In order to both preserve symmetry and reduce the number of parameters we will assume that all gate concentrations are equal to g, and that all reaction rate terms are equal to r. Converting the system to matrix notation we get:

$$\dot{\mathrm{P}} = \begin{bmatrix} -k & 0 & -r \\ -r & -k & 0 \\ 0 & -r & k \end{bmatrix} P + \begin{bmatrix} rg \\ rg \\ rg \end{bmatrix} \tag{25}$$

The matrix for the homogeneous portion of equation (25) has one real and two complex eigenvalues:

$$\lambda_1 = -(k+r) \quad \lambda_2 = r/2 - k - i\sqrt{3}r/2 \quad \lambda_3 = r/2 - k + i\sqrt{3}r/2$$

The corresponding eigenvectors are:

$$\mathrm{X}_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \quad \mathrm{X}_2 = \begin{bmatrix} -1/2 - i\sqrt{3}/2 \\ -1/2 + i\sqrt{3}/2 \\ 1 \end{bmatrix} \quad \mathrm{X}_3 = \begin{bmatrix} -1/2 + i\sqrt{3}/2 \\ -1/2 - i\sqrt{3}/2 \\ 1 \end{bmatrix}$$

For λ_1 we have a real solution of the form $\mathrm{U}_1 = \mathrm{X}_1\ e^{\lambda_1 t}$. For λ_2 we have a solution of the form $\mathrm{X}_2\ e^{\lambda_2}$ which can be separated into real and imaginary parts:

$$\mathrm{X}_2 e^{\lambda_2} = \mathrm{X}_2 e^{(r/2-k)t} e^{-i\sqrt{3}rt/2} = \mathrm{X}_2 e^{(r/2-k)t}(\cos(-\sqrt{3}rt/2) + i\sin(-\sqrt{3}rt/2)) = \mathrm{U}_2 + i\mathrm{U}_3$$

where

$$\mathrm{U}_2 = \begin{bmatrix} -1/2\cos(-\sqrt{3}rt/2) + \sqrt{3}/2\sin(-\sqrt{3}rt/2) \\ -1/2\cos(-\sqrt{3}rt/2) - \sqrt{3}/2\sin(-\sqrt{3}rt/2) \\ \cos(-\sqrt{3}rt/2) \end{bmatrix} e^{(r/2-k)t}$$

and

$$\mathrm{U}_3 = \begin{bmatrix} -\sqrt{3}/2\cos(-\sqrt{3}rt/2) - 1/2\sin(-\sqrt{3}rt/2) \\ \sqrt{3}/2\cos(-\sqrt{3}rt/2) - 1/2\sin(-\sqrt{3}rt/2) \\ \sin(-\sqrt{3}rt/2) \end{bmatrix} e^{(r/2-k)t}$$

The real and imaginary parts are two linearly independent solutions to the homogeneous part of (25). When combined with the real eigenvalue solution we get the fundamental matrix:

$$\mathrm{U} = \begin{bmatrix} \mathrm{U}_1, & \mathrm{U}_2, & \mathrm{U}_3 \end{bmatrix}$$

To satisfy the non-homogeneous part of (25) we must find a particular solution of the form:

$$\mathrm{X}_p = \begin{bmatrix} k1 \\ k2 \\ k3 \end{bmatrix}$$

Substituting X_p for X in equation (25) we get:

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -k & 0 & -r \\ -r & -k & 0 \\ 0 & -r & k \end{bmatrix} \begin{bmatrix} k1 \\ k2 \\ k3 \end{bmatrix} + \begin{bmatrix} rg \\ rg \\ rg \end{bmatrix}$$

Which implies that

$$\mathrm{X}_p = \begin{bmatrix} gr/(k+r) \\ gr/(k+r) \\ gr/(k+r) \end{bmatrix}$$

Thus we have a general solution of the form:

$$\mathrm{P} = \mathrm{UC} + \mathrm{X}_p$$

for some constant vector C as determined by the initial conditions.