

Circuit Development Using Biological Components: Principles, Models and Experimental Feasibility

RAJESH KRISHNAN

Electrical & Computer Engineering Department
University of Cincinnati
Cincinnati, Ohio 45221-0030
Email: krishnrh@ececs.uc.edu

CARLA PURDY

Electrical & Computer Engineering Department
University of Cincinnati
Cincinnati, Ohio 45221-0030
Email: carla.purdy@uc.edu

1 Abstract

Today most VLSI circuits are built in silicon using CMOS transistors. Developments in design automation and process fabrication have resulted in the progressive increase of the number of transistors per chip and decrease in the size of the transistors. But transistor designers are fast approaching fundamental physical barriers to further size reduction. Thus engineers are looking at alternate technologies such as nanodevices and biocircuits for next-generation circuits. In our research, we concentrate on the development of biocircuits and their applications. Our eventual goal is the design and simulation of complete systems integrating biocircuits and VLSI technology appropriately. Biocircuits are circuits developed *in vivo* or *in vitro*, using DNA and proteins. A biological process such as glycolysis or bioluminescence can be viewed as a genetic regulatory circuit, a complex set of biochemical reactions regulating the behavior of genes, operons, DNA, RNA and proteins. Similar to voltage in an electrical circuit, a genetic regulatory circuit produces an output protein in response to an input stimulus. We can engineer biocircuits to meet design specifications, using genetic engineering. Our aim is to build a library of *in vitro* biocircuits representing the Boolean functions. The biocircuits from this library can be further cascaded to form larger circuits. In this paper, we review the feasibility of building biocircuits. We discuss the construction of Boolean logic gates such as NOT, AND and OR and their verification by simulation. We also address important aspects such as cascading of the biocircuits and practical implementation. In addition, we describe an algorithm Box that can help to control biocircuit characteristics such as gain and switching behavior. This approach is similar to design space exploration in traditional VLSI, but takes into account biological knowledge obtained through experiments. We also provide insight into the robustness of biocircuits in the presence of noise. This paper is intended to pave the way for electrical engineers to start exploring the field of biocircuits.

Keywords – biocircuits, bio-inverter, noise model, sensitivity analysis, optimization.

2 Introduction

Conventionally, VLSI circuits are built using CMOS transistors and the circuits are typically fabricated on silicon. Developments in the areas of design automation

and process fabrication have resulted in the progressive increase of the number of transistors per chip and the progressive decrease in the size of the transistors. Transistors are fast approaching the size of atoms and this presents a fundamental barrier to further reduction in transistor sizes [1]. Scientists are already looking at alternate technologies that can be used to develop the next-generation circuits. The three main research areas in this direction are nanotechnology, quantum computing and biocomputing.

Nanotechnology deals with the design of electronic circuits at the atomic range of 0.1-100nm [2]. At the time this paper is being written, Intel has already developed the 45nm process technology [3]. Several issues must be considered while developing nanocircuits, including drain-induced barrier lowering (DIBL) and subthreshold leakage. Nanotechnology paves the way to develop lab-on-a-chip systems by merging circuits and MEMS technology. These lab-on-a-chip systems are widely used in the field of nanomedicine [4].

Quantum computing processes information based on the concepts of quantum physics such as quantization, interference and entanglement. The data is represented via qubits, $|0\rangle$ and $|1\rangle$. The qubit not only has a continuous state such as an analog quantity but also can exist in a superposition of states. This implies that the quantum bit can exist in the two states ($|0\rangle$ and $|1\rangle$) at the same time [5]. A few hundred qubits can perform large computations that would require a large set of bits to solve in a classical computer. Two important algorithms which have been developed for quantum computation are Grover's algorithm for database searching [6] and Shor's algorithm for factoring large numbers [7]. The biggest problem in the field of quantum computation is decoherence or the loss of quantum properties [8]. The topological quantum computer overcomes this problem by using anyons (quasi particles) that possess higher stability [9]. Recent efforts have shown the possible practical development of the topological quantum computer [9].

Biocomputing, as the name suggests, is computation performed using biomolecules such as DNA and proteins. A biological process such as glycolysis [10] or bioluminescence [11] can be viewed as a genetic regulatory circuit. A genetic regulatory circuit is a

complex set of bio-chemical reactions that regulates the behavior and function of genes, operons, DNA, RNA and proteins. The biological process exhibits regulatory functions such as the production of luminescence in *Vibrio Fischeri* [12] cells or the production of Cro protein in the phage lambda circuit [13]. The analogy to an electrical circuit can be easily shown. Any electrical circuit exhibits clearly defined behavior or function. When the input voltage is applied to the circuit, the circuit exhibits an output response. Similarly the gene regulatory circuit produces an output protein in response to the concentration of one or more input proteins. The operation of the gene regulatory circuit can be verified by observing the biological process *in vivo* or in the lab, *in vitro*.

3 Advantages of biocircuits

Biological circuits are cheap, consume very little energy and are self reproducing. Circuit operations in living cells also pave the way for a number of applications such as sensor arrays and drug discovery [14]. The reduction in cost and self reproduction of biocircuits is essential for creating multiple copies of biocircuits performing similar functions. Creation of multiple copies of cheaper biocircuits paves the way for the development of a parallel processor. Cells with well defined intercellular communication channels can be modified to serve as circuits with predefined complex interconnections.

4 Disadvantages of biocircuits

The main disadvantage of biocircuits is the switching time or the response time. The typical switching time for an electrical inverter is a few nanoseconds (10^{-9}) whereas the typical switching time for the bio-inverter is in minutes [15]. It is too early to consider the long switching time to be a deterrent from exploring biocircuits as an alternate technology. Future research may find ways to either reduce this time through genetic control or alleviate its effect through parallelism. The other main disadvantage is the need for a vast library of proteins and matching gates. No two proteins can be repeated in one logic circuit. These conditions severely restrict the number of gates that can be cascaded together to form a circuit. Another important concern is that the output concentration decays constantly. Hence the signal strength or concentration is another important factor to be considered when biocircuits are cascaded. These issues have to be addressed when trying to build large circuits from the basic library of biocircuits. Finally there is no standard fabrication procedure or experimental standard defined for mass production of biocircuits. These issues have to be addressed in order to make biocircuits a viable alternate technology to replace silicon technology.

5 Issues to be considered in designing biocircuits

There are several issues and questions that arise in the mind of a conventional circuit designer using DNA and proteins to build biocircuits. Some of these are

- 1) Is it possible to develop Boolean logic gates such as NOT, AND, OR, etc. from proteins and DNA?
- 2) Is it possible to accurately simulate the behavior of these biocircuits before they are constructed?
- 3) Is it feasible to build biocircuits in the laboratory?
- 4) What is the robustness of the biocircuits in the presence of noise?
- 5) How can we modify and control circuit parameters such as gain and noise margin?

This paper has the goal of providing answers to all these questions in a particular case. We discuss the feasibility of developing biocircuits, show the simulation of a bio-inverter and also survey experimentally developed biocircuits such as the BioBricks modules [16] that can implement Boolean operations.

6 Development of the most fundamental biocircuit: the bio-inverter

The most fundamental biocircuit is the bio-inverter. This inverter can be implemented using the switching mechanism observed in the phage lambda virus infecting an *E. coli* bacterium. Such an inverter has been implemented experimentally by Weiss [17]. We have developed a model for such an inverter in both Matlab and VHDL-AMS. Our simulations show that the bio-inverter possesses good noise margin and gain, even though the gate switching time is approximately 10^3 seconds.

The switching mechanism of the phage lambda system representing the “OFF” and “ON” state of a switch matches the typical inverter behavior as explained below. The virus infecting the bacterium exists in two states, lytic and lysogenic. The first state, the lysogenic state or the “OFF” state, is the one where the virus is dormant in the absence of UV light. In this state, the input protein, Rp, is present in high concentration and the output protein, Cro, is suppressed. In the second state, the lytic state or the “ON” state, the virus divides aggressively in the presence of UV light [18]. In this state, the input protein (Rp) is suppressed and the output protein (Cro) is synthesized [Figure 1].

7 Languages for modeling bio-circuits

The bio-inverter was modeled in Matlab and VHDL-AMS. While the Matlab model was developed for simulation purposes, the VHDL-AMS model was developed for compatibility purposes. The VHDL-AMS language supports multiple energy domains and can be easily extended to model biosystems also. Hence the development of VHDL-AMS models is very useful, since

both biocircuits and analog and digital hardware can be modeled using the same language and in one program. This is an important step towards routinely interfacing biocircuits with traditional hardware.

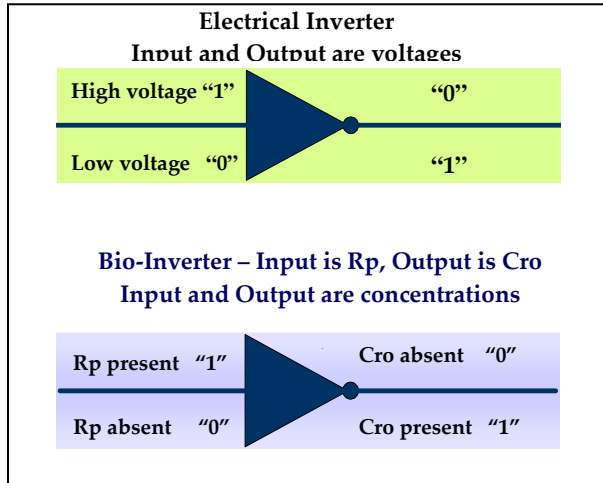


Figure 1: Analogy between electrical inverter and the phage lambda switching states.

8 Modeling the bio-inverter

We first model the bio-inverter using a deterministic, differential equation model. Our model is based on the work done by Weiss [17]. Our inverter model consists of ten different chemical reactions that occur in *E. coli* infected with lambda virus [Figure 2]. These chemical reactions can be rewritten as a set of first order differential equations [Figure 3].

The bio-inverter model includes eight species, Rp, Rp2, mRNA, RNA_{poly}, rRNA, Cro, Cg and CgRp2. We assume that the concentrations of RNA_{poly}, rRNA and Cg are constant. The model includes ten reactions with ten rate constants. The rate constants k1 to k10 are obtained from Ptashne and Weiss [12, 17]. The rate constants used in modeling the rate equations are given in Table 1.

Rate	k1	k2	k3	k4	k5
Value	8.333	0.1667	0.5775	0.5775	66.67
Rate	k6	k7	k8	k9	k10
Value	0.2	0.001	0.03	2.0	0.5575

Table 1: Rate constants for the bio-inverter.

To maintain a constant supply of input protein (Rp), we also add a “drive” term to the rate equation involving Rp, as in the Weiss model [17]. The values of RNA_{poly} and rRNA are taken as 1 Molar and the concentration of Cg is 0.07μM. The characteristics of our simplified inverter model are in agreement with those of [17]. The model was simulated in Matlab and VHDL-AMS. The VHDL-AMS model was created in order to interface the inverter model to the digital circuit described in [19]. VHDL-AMS allows us to explore the system level behavior of circuits, including circuits with biocomponents. Our

deterministic model will allow for efficient simulation. The transfer characteristics between output Cro and input Rp were calculated by solving the differential equations under equilibrium. At equilibrium the rate of change (d/dt) is zero and hence the RHS of the equation is equated to zero. Then the relation between Cro and Rp is derived by solving all the equations from Figure 3, to give equation (1).

$$Cro = \frac{k_7 \cdot k_8 \cdot [RNA_{poly}] \cdot [rRNA] \cdot C_g}{k_9 \cdot k_{10} \cdot \left(1 + \frac{k_5}{k_6} \cdot R_p\right)} \quad (1)$$

Figure 4 shows the relation between Cro and Rp. The repressor affinity, i.e., the binding affinity of the repressor (Rp) to the gene Cg is very high. Hence the switching threshold is very low. We also model the transient characteristics using a deterministic approach. The transient characteristics were also modeled using both Matlab and VHDL-AMS. The switching mechanism of the inverter can be observed clearly in Figure 5. The slight overshoot of Rp concentration is due to the external drive added to the system. The drive helps in maintaining a constant supply of input protein for a given time period. The output switches from logic 0 to logic 1 in about 100 seconds but output switches from 1 to 0 within 30 seconds as seen in Figure 5. The inverter has a lower transition time (high to low) due to the high repressor affinity. The dynamic behavior of the bio-inverter is comparable to that of an electrical inverter. The high output steady state value is 0.1883μM.

9 Comparing the bio-inverter model to an electrical inverter circuit

The input and output to the bio-inverter model are concentrations of the molecular species, whereas the input and output to the electrical inverter circuit are voltages. The transient characteristics of the bio-inverter are similar to the electrical inverter. The main difference between the bio-inverter and the electrical inverter is the total time for switching. The typical switching time for an electrical inverter is a few nanoseconds (10^{-9}), whereas the typical switching time for the bio-inverter is in seconds (approx 10^3 seconds).

10 Robustness of the bio-inverter in the presence of noise

The robustness of a circuit is defined in terms of the operation of the circuit in the presence of noise. A robust circuit’s output should be stable and should not change in the presence of noise. In this paper, we have studied the robustness of the bio-inverter circuit by developing a stochastic model that incorporates noise in terms of randomness in reaction time and in the type of reaction chosen. The stochastic model of the phage lambda switching system is based on the Gillespie algorithm [20]. The repressor dimer (Rp2) was fixed as the input to the

stochastic model for simulation purposes. The stochastic model was simulated using the initial values given in Table 2 and Table 3. Figure 6 shows the lysogenic state or the “OFF” state of the bio-inverter. Figure 7 shows the lytic state or the “ON” state of the bio-inverter.

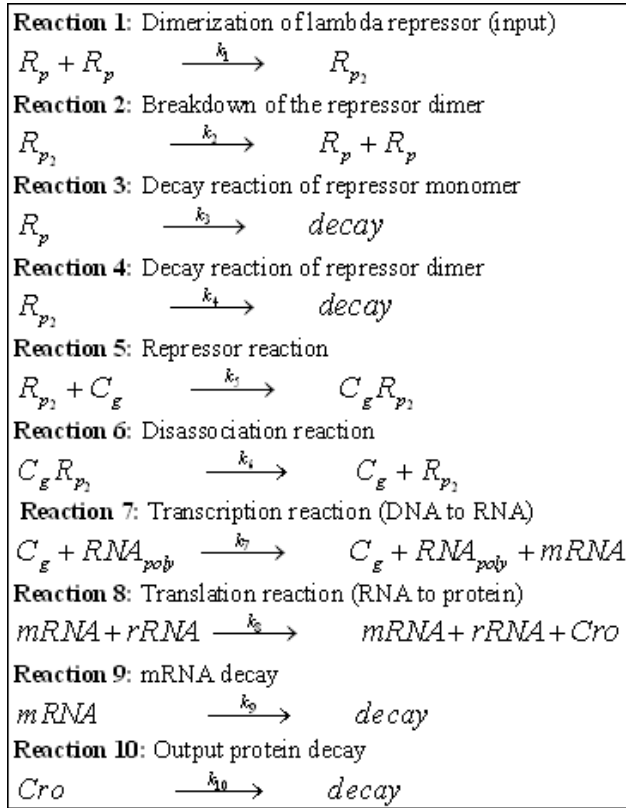


Figure 2: Reactions that model the lytic and lysogenic states of the lambda virus.

$\frac{d[R_p]}{dt} = -2k_1 \cdot [R_p]^2 + 2k_2 \cdot [R_{p_2}] - k_3 \cdot [R_p]$ $\frac{d[R_{p_2}]}{dt} = k_1 \cdot [R_p]^2 - k_2 \cdot [R_{p_2}] - k_4 \cdot [R_{p_2}] - k_5 \cdot [R_{p_2}] [C_g] + k_6 \cdot [C_g R_{p_2}]$ $\frac{d[C_g]}{dt} = -k_5 \cdot [C_g] [R_{p_2}] + k_6 \cdot [C_g R_{p_2}]$ $\frac{d[C_g R_{p_2}]}{dt} = k_5 \cdot [C_g] [R_{p_2}] - k_6 \cdot [C_g R_{p_2}]$ $\frac{d[mRNA]}{dt} = k_7 \cdot [C_g] [RNA_{poly}] - k_9 \cdot [mRNA]$ $\frac{d[Cro]}{dt} = k_8 \cdot [mRNA] [rRNA] - k_{10} \cdot [Cro]$

Figure 3: Differential equations that model the inverter.

From Figure 6, we can see the phage lambda system switches to the lysogenic state or the “OFF” state. The input (Rp2) increases to 50 molecules and the output (Cro) decreases to 0 (approx) in the “OFF” state. From Figure 7, we can see that the phage lambda system switches to the lytic state or the “ON” state. The input

(Rp2) decreases to 0 and the output (Cro) increases to 90 molecules (approx). The simulations show that the phage lambda switching operation is very robust even in the presence of randomness and noise. So irrespective of the model we choose (deterministic or stochastic), the inverter operation can be simulated accurately.

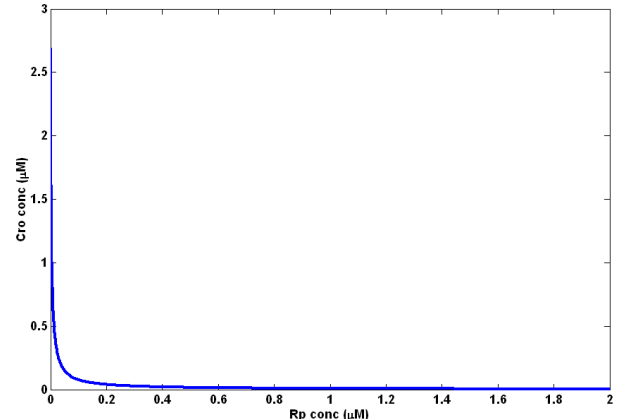


Figure 4: Transfer characteristics between Rp and Cro concentrations (Matlab).

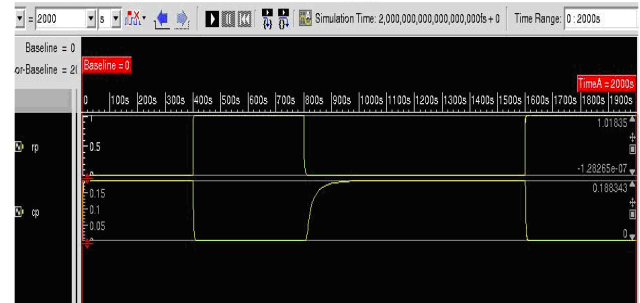
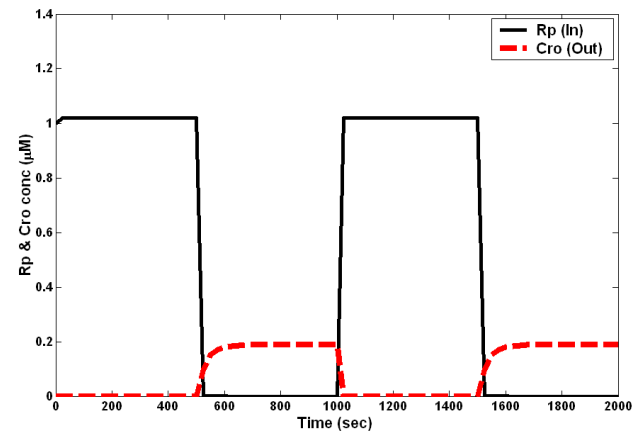


Figure 5: Transient characteristics of the lambda inverter (top value Rp, lower value Cro).

11 Experimental development of the bio-inverter

Example bio-inverters have been experimentally developed in the laboratory by Weiss [21] and Hasty et al [22]. The response time of a wild type naturally existing

phage lambda inverter was measured in hours. The wild type bio-inverter switches from one state to another in about 2 to 3 hours [23]. There are two ways to make the bio-circuits into a feasible method of alternate computation. One method is to reduce the response time of the bio-circuits by genetic modification of the wild type components. Another method is to develop a large number of circuits that will simultaneously work on the inputs and thus develop a parallel processor.

Species	Rp	Rp2	Cg	CgRp2
Molecules	250	1	30	1
Species	RNAPoly	rRNA	mRNA	Cro
Molecules	1	1	1	1

Table 2: Initial values (number of molecules) of the bio-chemical species for the “OFF” (lysogenic) state.

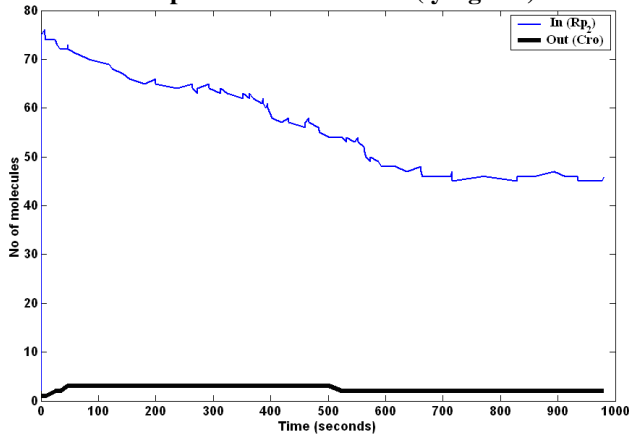


Figure 6: Stochastic simulation of the ‘OFF’ state or lysogenic state of the bio-inverter.

Species	Rp	Rp2	Cg	CgRp2
Molecules	50	1	50	1
Species	RNAPoly	rRNA	mRNA	Cro
Molecules	1	1	20	20

Table 3: Initial values of the bio-chemical species for the “ON” (lytic) state.

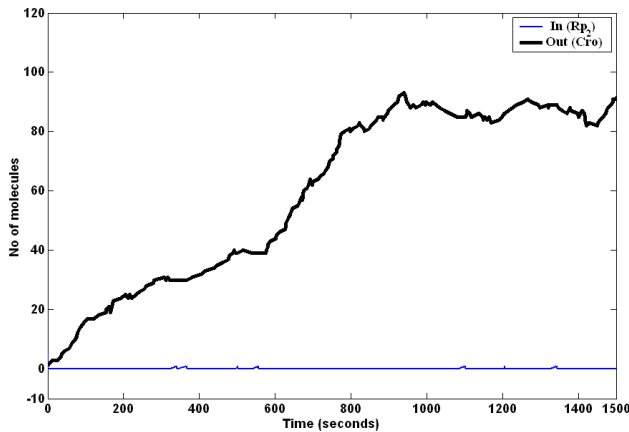


Figure 7: Stochastic simulation of the ‘ON’ state or the lytic state of the bio-inverter.

We have discussed the theoretical simulation and the practical development of the bio-inverter. The bio-inverter is actually a promoter/ repressor pair that exhibits inverter characteristics. The next essential step is develop other Boolean gates such as AND, OR, NAND, NOR and XOR gates. In the VLSI circuit technology, the CMOS transistor forms the fundamental entity and the Boolean gates such as inverter, NAND and NOR are built by assembling the CMOS transistors in different configurations. In the case of bio-circuits, the promoter/repressor pair forms the most fundamental entity and all other logic circuits have to be built by putting together different promoter/repressor pairs. There are several factors to be considered when cascading different promoter/repressor pairs or bio-inverters. We cannot connect two bio-inverters with the same input (Rp) and output protein (Cro). The reason is obvious. The first inverter will only produce Cro as output. Cro is not an input to the second inverter. Hence irrespective of the output of the first inverter, the output of the second inverter will depend only on the presence/absence of Rp. Hence direct cascading of identical gates is not possible. The only way to cascade bio-inverters is to make the output of the first inverter act as the repressor/promoter of the second inverter. Thus we need different types of proteins and matching gates to build a logic circuit. In a promoter/repressor pair, the repressor protein has to be suppressed for the output protein to be produced. Hence an inducer such as UV light or other lytic agents such as hydrogen peroxide, mitomycin, benzpyrene is introduced to the promoter/repressor pair as an input. We have shown the development of Boolean gates such as NOT, AND, OR, NOR and NAND gates using different promoter/repressor pairs and inducer inputs in Figure 8.

The NOT gate is built using two promoter/repressor pairs. The inducer input is applied to the first promoter/repressor pair (P1/R1). The output protein produced by the first repressor/promoter pair acts as the repressor (R2) to the second promoter (P2). Hence whenever the inducer input is introduced then second promoter is repressed and no output is produced. When no inducer input is present, then the second promoter will produced the output protein.

The AND gate can be built using a single repressor/promoter pair which is activated using two inducers. Both the inducers have to be present to activate the output protein production. The NAND gate can be built by connecting another promoter (P) to the AND gate such that the output protein from the AND gate acts as the repressor to the second promoter (P). The cascaded promoter acts as an “always ON” inverter. Whenever the first AND gate produces the output protein then the second promoter (P) is repressed.

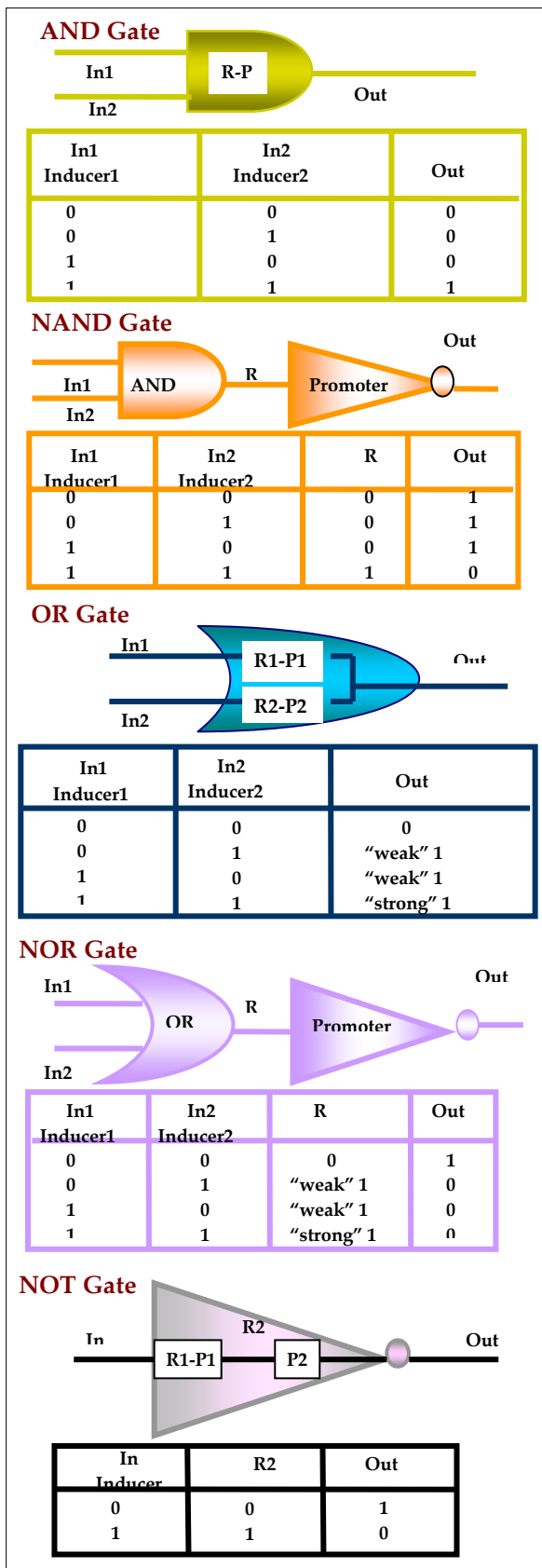


Figure 8: Boolean gates developed using different promoter/repressor pairs. Promoters are denoted as P, P1, P2 and repressors are denoted as R1, R2.

The OR gate is built using two different promoter/repressor pairs that transcribe the same output protein. Output is produced when either of the inducers or both the inducers are present. There will be a difference in output concentration when only one inducer is present and when both the inducers are present. The difference in concentrations is characterized as "weak 1" and "strong 1" as given in the truth table for the OR gate (Figure 8). The important point for the designer is to make sure that the switching threshold of the cascading gate to the OR gate should be less than or equal to the "weak 1" concentration. The NOR gate is built by cascading a promoter to the OR gate such that the output from the OR gate acts as the repressor to the cascaded promoter.

Thus we need a library of different bio-inverters to create a library of bio-circuits. BioBricks is an open source project to develop different types of promoters/ repressor pairs, i.e., bio-inverters [16]. Boolean gates can be put together by combining the different promoter/repressor pairs that are catalogued in the BioBricks website. Circuits such as AND gate [22], NOR logic gate [24] and oscillator [25] have also been built by cascading different promoter/repressor pairs.

13 Inter-cellular communication and circuit interconnects

One way to overcome the cascading problem of individual circuits is to look at well studied and documented biological pathways with predefined blocks that communicate with one another. Intercellular communication among the cells thus forms an excellent form of automatic interconnection among the circuits. Each cell can be programmed to perform a defined function or all the cells can be programmed to do just one function. The intercellular communication among the cells forms automatic complex interconnects. *Vibrio Fischeri* cells communicate with each other via chemical (auto-inducer) release. This type of communication can be regulated to act as interconnect. The intercellular interconnect helps in forming large circuits by cascading smaller circuits. If all the cells perform the same function then the intercellular communication also helps in increasing the parallel processing capability of the circuits [26].

14 Control of the circuit parameters

The gain of the bio-inverter is very low in Figures 4 and 5 above. Hence a wild type or naturally occurring phage lambda inverter cannot be used as a circuit component without improving the characteristics such as gain and switching threshold. The protein concentration and gain can be controlled by modifying one or more of the rate constants for the equations given in Figure 2. Biological systems represent a high dimensional input-output model. Mutations or optimizations of all these reactions, to get the desired outcome, are impossible. Random control of

one or more reaction parameters does not guarantee the expected results and is also time consuming and expensive. Mathematical modeling and simulation can reduce the actual physical experimentation involved [27]. Strict mathematical modeling alone is not enough because some of the mathematically sensitive parameters are not experimentally controllable *in vitro*. Hence there is a need to incorporate experimental constraints in the algorithm.

We have developed a new algorithm, "Box", for controlling the circuit characteristics of the bio-inverter (Figure 9). The algorithm integrates laboratory constraints and process behavior with the mathematical model. The Box algorithm combines a mathematically based procedure with the biological rules that govern the experimentation, to provide the desired results. The control parameters are calculated by heuristic algorithms such as simulated annealing. The algorithm was programmed using Matlab.

The "Box" algorithm (Figure 9) consists of six blocks:

- **Bio-Model:** The first block is the mathematical model which can be either an ordinary differential equation (ODE) model or a stochastic model (model that incorporates noise).
- **Analytical Tools:** The second block is the analytical block (RS-HDMR algorithm) that estimates the effect of each rate constant on the output. The RS-HDMR algorithm also estimates the effect of the pair correlation of the rate constants on the output [28-30]. After applying the analytical algorithm, we sort the rate constants in terms of the sensitivity of the output to each rate constant. Then we assign a priority number to each rate constant, depending on the sensitivity. The output is highly sensitive to the rate constant that has the highest priority number.
- **Bio-Control Database:** The Bio-Control Database provides the control parameters for each reaction based on the reaction type. The algorithm also accepts user-defined controls and adds these controls to the database. The bio-control database is being developed by extensive data mining of publications and articles in the PubMed database [31]. The experimental controls derived from the literature are filled into the corresponding slots. The bio-control database should include slots for all the different types of experimental controls such as pressure, temperature, radiation (e.g., UV light), chemicals, proteins, and ribosomal binding sites. The database also includes the ease of controllability of the reaction. Classification of the reactions is essential for deriving the priorities for the reactions. The bio-rules procedure derives a priority for each reaction

based on the degree of controllability of the reaction.

- **Bio-Rules:** The next block consists of a rule-based procedure. In this procedure a priority number is given to each reaction based on the experimental constraints and controllability of the reaction. Highest priority is given to reaction which has a highest degree of controllability than other reactions. This block is very crucial for converting the simulation results into practical results in the laboratory. The bio-rules procedure is derived from the laboratory constraints such as the ease of experimentation and the reactions that can be easily modified or controlled. The bio-rules procedure also combines the priorities given by the RS-HDMR algorithm with the priorities given by the bio-rules procedure. Thus we obtain a priority order, which is a combination of both the mathematical and the laboratory constraints.
- **Optimization:** This block applies heuristic algorithms such as simulated annealing to optimize the output concentration. The optimization block chooses a subset of rate constants that have the highest priority level. The subset includes two or more rate constants, depending upon user preference. For example, we chose two rate constants for controlling the phage lambda system, whereas the subset for the TNF α -Mediated NF- α B signaling pathway, which we have also worked on, consisted of four rate constants [32]. The rate constants are modified by applying the Metropolis procedure [33]. The limits should be carefully chosen. The limits for each rate constant should be decided after making sure that the limits can be achieved by proper mutation and experimentation in the laboratory. Heuristic algorithms such as genetic algorithm can also be applied for optimization and the results can be compared with simulated annealing.
- **Compare Block:** The output from the optimization block is compared with the desired output. Optimization is iteratively applied until the desired output is obtained. The Compare Block then displays the modified rate parameters and the experimental steps through which the reactions can be controlled.

The Box algorithm was applied to the bio-inverter to increase the gain factor by 10. The output of the normal bio-inverter is 0.2 μ M. An increase in the gain factor by 10 will increase the output to 2 μ M. Figure 10 confirms the improved transient characteristics achieved by applying the Box algorithm.

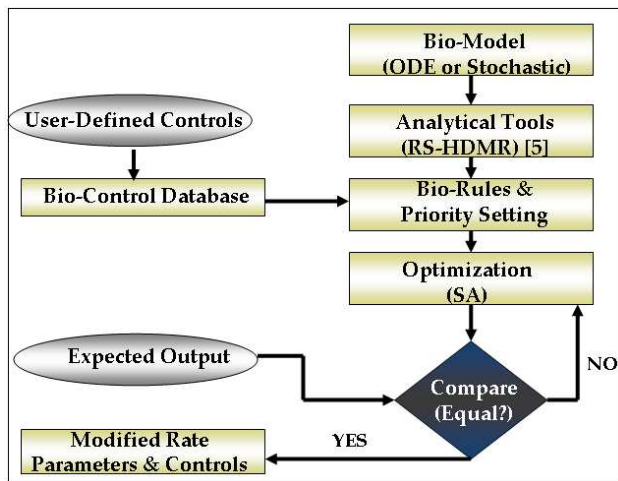


Figure 9: Block diagram of the Box algorithm.

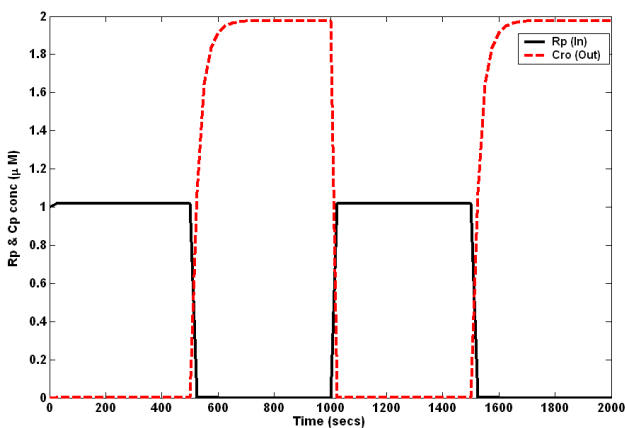


Figure 10: Inverter transient characteristics (ODE model) with expected gain of 10.0 and output value of 2.0.

15 Conclusions and future work

This paper was created to answer the questions that arise when exploring the development of circuits using DNA and proteins. We adopted the conventional circuit designer's perspective so as to help break the major barrier that is present today in the form of lack of knowledge of biological processes and terms. We have explained the switching mechanism of phage lambda in *E. coli* and derived the bio-inverter circuit. Simulation results (Matlab and VHDL-AMS) show that the inverter can be programmed to express good transfer characteristics and dynamic behavior. We were able to control the gain and output values of the bio-inverter using the Box algorithm which we have developed. The bio-inverter is a robust circuit even in the presence of randomness and noise.

We discussed the importance of BioBricks in constructing different Boolean gates and making a library of gates. We discussed the difficulties in cascading bio-circuits and expressing a circuit function. We also mentioned the possibility of developing complex interconnects by

regulating the intercellular communication among the cells. Future research can be done to alleviate the problems of long switching times and vast library construction. We hope this paper provides a strong foundation for further probing in the area of bio-circuits to fully tap their potential.

References

- [1] P. Rutten, M. Tauman, H. Bar-Lev, A. Sonnino, "Is Moore's Law Infinite? the economics of Moore's law", *Kellogg TechVenture Anthology*, 2001.
- [2] M. Ratner, D. Ratner, *Nanotechnology: A Gentle Introduction to the Next Big Idea*, Upper Saddle River, NJ : Prentice Hall, 2003.
- [3] 45-Nanometer Technology, developer.intel.com/technology/silicon/45nm_technology.htm, date accessed 04/02/07.
- [4] R. A. Freitas Jr, *Nanomedicine*, Austin, TX : Landes Bioscience, 1999.
- [5] D. Bouwmeester, A. Ekert, A. Zeilinger, *The Physics of Quantum Information: Quantum Cryptography, Quantum Teleportation, Quantum Computation*, NY: Springer, 2000.
- [6] L. K. Grover, "A fast quantum mechanical algorithm for database search", *28th Annual ACM Symposium on the Theory of Computing*, pp. 212-219, May 1996.
- [7] P.W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer", *SIAM J. Comp* 26 (5), pp. 1484-1509, 1997.
- [8] W. H. Zurek, "Decoherence and the transition from quantum to classical", *Physics Today* 44 (10), pp. 36-44, 1991.
- [9] G. P. Collins, "Computing with quantum knots", *Scientific American* 294 (4), pp. 56-63, 2006.
- [10] M. L. Campbell, *Cell Modeling*, M.S. Thesis, Air Force Institute of Technology, WPAFB, March 2002.
- [11] C. Belta, J. Schug, D. Thao, V. Kumar, G. J. Pappas, H. Rubin, P. Dunlap, "Stability and reachability analysis of a hybrid model of luminescence in the marine bacterium *Vibrio fischeri*", *40th IEEE Conference on Decision and Control* 1 (4), pp. 869-874, 2001.
- [12] R. Weiss, *Cellular Computation and Communications Using Engineered Genetic Regulatory Networks*, PhD Thesis, MIT, September 2001.
- [13] M. Ptashne, "A Genetic Switch: Phage Lambda Revisited", Cold Spring Harbor, NY: Cold Spring Harbor Laboratory Press, 2004.
- [14] NF-kB Family - Interactive Pathway™ and Product Listing, EMD Biosciences, http://www.emdbiosciences.com/html/CBC/NFKB_NFkappaB_IKB_IKK_Pathway_Products.htm, date accessed 04/02/07.
- [15] A. Arkin, J. Ross, H. H. McAdams, "Stochastic Kinetic Analysis of Developmental Pathway Bifurcation in phage {lambda}-Infected *Escherichia coli* Cells", *Genetics* 149 (4), pp. 1633-1648, 1998.
- [16] Registry of Standard Biological Parts, http://parts.mit.edu/registry/index.php/Main_Page, date accessed 04/02/07.
- [17] R. Weiss, E. Homsy, F. Knight, "Toward in vivo digital circuits", *DIMACS Workshop on Evolution as Computation*, January 1999.
- [18] M. Ptashne, *A genetic switch: gene control and phage [lambda]*, Cambridge, MA, Cell Press, Blackwell Scientific Publications, 1986.

- [19] R. Krishnan, and C. Purdy, "Bio-inverter model and interface to digital hardware", *48th International IEEE Midwest Symposium on Circuits & Systems* 1, pp. 766-769, August 2005.
- [20] T. Gillespie, "Exact stochastic simulation of coupled chemical reactions", *J.Phys.Chem* 81 (25), pp. 2340-2361, 1977.
- [21] R. McDaniel, R. Weiss, "Advances in synthetic biology: on the path from prototypes to applications", *Current Opinion in Biotechnology* 16 (4), pp. 476-483, 2005.
- [22] J. Hasty, D. McMillen, J. J. Collins, "Engineered gene circuits", *Nature* 420 (6912), pp. 224-230, November 2002.
- [23] S. Atsumi, J. W. Little, "A synthetic phage {lambda} regulatory circuit", *PNAS* 103 (50), pp. 19045-50, December 2006.
- [24] C. Guet, M.B. Elowitz, W. Hsing, S. Leibler, "Combinatorial synthesis of genetic circuits", *Science* 296 (5572), pp. 1476-1470, May 2002.
- [25] M.B. Elowitz, and S. Leibler, "A synthetic oscillatory network of transcriptional regulators", *Nature* 403 (6767), pp. 335-338, January 2000.
- [26] C. D. Cox, G. D. Peterson, M. S. Allen, J. M. Lancaster, J. M. Mccollum, D. Austin, L. Yan, S. Sayler, M. L. Simpson, "Analysis of noise in quorum sensing", *OMICS: A Journal of Integrative Biology* 7 (3), pp. 317-334, 2003.
- [27] U.S. Bhalla, and R. Iyengar, "Emergent properties of networks of biological signaling pathways", *Science* 283 (5400), pp 339-340, January 1999.
- [28] G. Li, C. Rosenthal, H. Rabitz, "High dimensional model representations", *J.Phys.Chem. A* 105 (33), pp. 7765-7777, 2001.
- [29] G. Li, S. W. Wang, H. Rabitz, S.K. Wang, P.Jaffe, "Global uncertainty assessments by high dimensional model representations (HDMR)", *Chemical Engineering Science* 57 (21), pp. 4445-4460, 2002.
- [30] G. Li, S.W. Wang, and H. Rabitz, "Practical approaches to construct RS-HDMR component functions", *J.Phys.Chem.A* 106 (37), pp. 8721-8733, 2002.
- [31] PubMed Database, <http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?DB=pubmed>, date accessed: 04/02/07.
- [32] K. H. Cho, S. S. Young, W. Kolch, O. Wolkenhauer, "Experimental design in systems biology, based on parameter sensitivity analysis using a Monte Carlo method: A case study for the TNF α -mediated NF- κ B signal transduction pathway", *Simulation* 79 (12), pp. 726-739, 2003.
- [33] S. Kirkpatrick, C. D. Gelatt, M. P. Vecchi, "Optimization by simulated annealing", *Science* 220 (4598), pp. 671-680, May 1983.