# FOLD CHANGE DETECTION IN 3-NODE ENZYMATIC NETWORKS

**by**

**MAJA ŠKATARIĆ**

**A thesis submitted to the**

**Graduate School-New Brunswick**

**Rutgers, The State University of New Jersey**

**In partial fulfillment of the requirements**

**For the degree of**

**Master of Science**

**Graduate Program in Electrical and Computer Engineering**

**Written under the direction of**

**Professor Eduardo Sontag**

**And approved by**

_____

_____

_____

**New Brunswick, New Jersey**

**May, 2011**

## ABSTRACT OF THE THESIS

### Fold Change Detection in 3-node Enzymatic Networks

**by Maja Škatarić**

**Thesis Director:**
**Professor Eduardo Sontag**

Complex networks are studied across many fields of science. To discover design principles that underlie these networks, network motifs are introduced, as sub-graphs of interconnections occurring in complex networks much more often then expected at random. A distinct set of network motifs were identified in many types of biological networks, such as gene transcriptional networks, neuronal networks, and enzymatic networks, but only a small fraction of them have been well described. By connecting recurrent motifs with a particular cellular function, it is hoped that one can understand the dynamics of the entire network based on the dynamics of its core motifs.

Two biologically important functions were introduced and motivated through examples from biology, namely, exact adaptation, which represents a system's ability to respond to a change in the input signal and return to its pre-stimulated state even when the change in input persists, and Fold Change Detection, which is a special property of adapting systems, where the output is invariant under the scaling of inputs.

In this thesis, the study of network motifs was used as a motivation to further explore the dynamics of all 3-node enzymatic networks capable of achieving Fold Change Detection property. A search through 16,038 topologies sampled with 10,000 parameters each, led to the conclusion that despite the diversity of enzymatic circuits, only small number of them is capable of achieving the FCD property, and the mechanism for achieving it can be understood through a theoretical and computational analysis.

# Acknowledgements

First and foremost I express my deepest and sincerest gratitude to my thesis advisor, Dr. Eduardo Sontag, who has supported me throughout my thesis with his guidance and patience. His knowledge, constructive comments, logical way of thinking, and persistent help have been of great importance for me.

I would like to thank my committee members, Prof. Sophocles Orfanidis and Prof. Predrag Spasojević. I would also like to thank Electrical and Computer Engineering Department faculty for teaching me valuable engineering concepts and providing me with a strong background for my research. My sincere gratitude goes to Prof. Zoran Gajić for his advice and help.

It is a pleasure to thank colleagues and great friends, Prashanth Kumar Kurup, Gun-Hyung Park and Thang Nguyen for good discussions and support.

Finally, my deepest gratitude goes to my family in Belgrade, for their unconditional love and support.

# Table of Contents

# List of Figures

# Chapter 1

# Introduction

## 1.1 Network motifs

Complex networks are studied across many fields of science. To discover design principles that underlie these networks Uri Alon and collaborators introduced the concept of a network motif.

Network motifs are patterns (sub-graphs) of interconnections occurring in complex networks much more often than expected at random (Alon, 2007a; Milo *et al.*, 2004). A distinct set of network motifs were identified in many types of biological networks such as gene transcriptional networks, neuronal networks, and enzymatic networks, but only a small fraction of networks have been well characterized (Ma *et al.*, 2009). In papers by Tyson and Novak (2010); Tyson *et al.* (2003), network motifs were indentified as basic information-processing modules in protein regulation networks. Cells employ receptors for temperature, damaged proteins or DNA, energy, availability, etc., and continually process these steams of information to make decisions about appropriate responses in terms of gene expression, metabolic activity, movement, growth and division and apoptosis (programmed cell death) (Tyson and Novak, 2010). By understanding the dynamics behind the behavior of individual motifs one expects to gain understanding how complex networks emerge, and establish rules that help verify if network can achieve particular function.

Moreover, despite the complexity of cellular networks, there might be only a limited number of network topologies that are capable of robustly executing any particular biological function. Some may be more favorable because of fewer parameter constraints. The same core set of network topologies might underlie functionally related cellular behaviors (Ma *et al.*, 2009). To understand how motifs are built and why they exhibit

particular functions that match cellular functions, in the paper by Milo *et al.* (2004) the authors start with simple networks where the interactions between nodes are represented by directed edges. Each network was scanned for all possible $n$-node subgraphs (here $n = 3$ and $n = 4$), and the number of occurrences of each subgraph was recorded. Each network contains numerous types of $n$-node subgraphs, as shown in Figure 1.1. To focus on those that are likely to be important, the real network was compared to a suitably randomized ensamble of networks, and only selected patterns that appear in the real network at numbers significantly higher than those in the randomized networks, as shown in Figure 1.2. Randomized networks are networks with same characteristics as the real network (same number of nodes and edges as the real network), but where connections between nodes are made at random. Patterns that occur in real networks significantly more often than in randomized networks are called "network motifs".

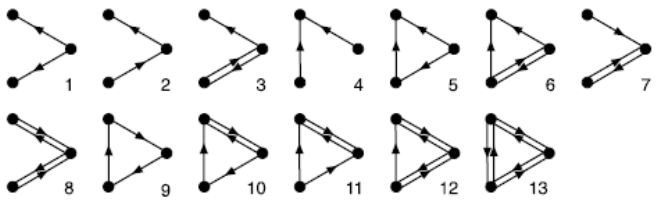



Figure 1.1: 13 types of three-node connected subgraphs, adapted from (Milo *et al.*, 2004)

The basic idea that patterns that occur in a real network much more often than in randomized networks must have been preserved over evolutionary timescales against mutations that randomly change edges. To appreciate this, it is important to point out that edges are easily lost in transcription networks- a mutation that changes a single DNA letter in a promoter might abolish binding of a transcription factor and cause the loss of an edge in the network. Similarly, new edges can be added to the network by mutations that generate a binding site for a transcription factor $X$ in a promoter region $Y$. These two concepts suggest that if a network motif appears in a network much more often than in randomized networks, it must have been selected based on some advantage it gives to the organism. For a stringent comparison in the Figure

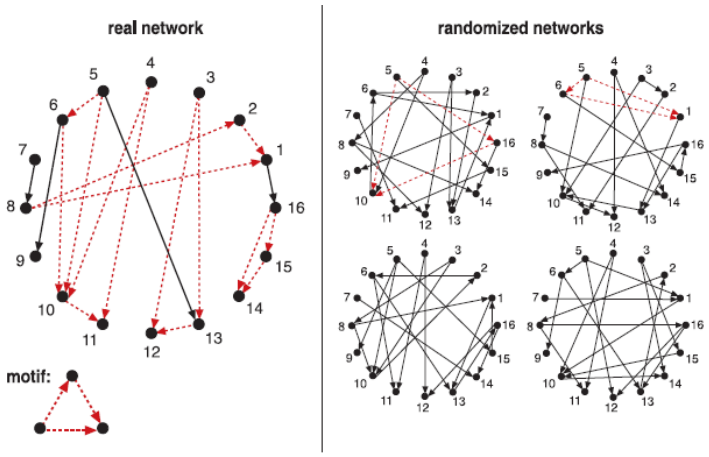



Figure 1.2: Network motif detection: Networks motifs are patterns that recur much more frequently in the studied network (left) then in an assemble of randomized networks(right), Figure is adapted from adapted from (Milo *et al.*, 2004)

1.2, randomized networks that have the same single-node characteristics, meaning the same number of incoming and outgoing edges as does the corresponding node in real network were used. Therefore, motifs are those patterns for which their probability $P$ of appearing in a randomized network an equal or greater number of times than in the real network is lower than a cutoff value (here $P = 0.01$) (Alon, 2007a).

## 1.2 Detecting network motifs by comparison to randomized networks

To detect network motifs, real networks must be compared to an assemble of randomized networks. The simplest way of generating random networks is due to Erdos and Renyi, and is as follows. The real network has $E$ edges and $N$ nodes. To compare using the ER model, one builds a random network with the same number of edges and nodes. In the random network defined by the ER model, directed edges are assigned at random between pairs of nodes. Since there are $N$ nodes, there are $N(N-1)/2$ possible pairs of nodes that can be connected by an edge. Each edge can point in one of two directions, for a total of $N(N-1)$ possible places to put a directed edge between two nodes. In addition, an edge can begin and end at the same node, forming a self-edge for total of $N$ possible self-edges. The total number of edges therefore is:

$$N(N-1) + N = N^2. \tag{1.1}$$

In the ER model, the $E$ edges are placed at random at $N^2$ possible positions, and therefore each possible edge position is occupied with probability

$$p = \frac{E}{N^2}. \tag{1.2}$$

## 1.3 Transcriptional networks

The living cell can sense many different signals (temperature, osmotic pressure, biological signaling molecules from other cells, beneficial nutrients, harmful chemicals, etc.) having corresponding receptors for each of these stimuli, and also respond to these signals by producing appropriate proteins that act upon internal and external environments. To represent environmental states, the cell uses special proteins called transcription factors, that rapidly transit between active and inactive molecular states at a rate that is indirectly modulated by a specific environmental signal (input). Each active transcription factor can bind the DNA to regulate the rate at which specific target genes are transcribed. A gene is a stretch of DNA whose sequence encodes the information needed for production of a given protein. Genes are transcribed into mRNA, which is then translated into protein (Central Dogma of Biology), which in turn can act on the environment (Alon, 2007a). The interaction between transcription factors and the genes that these factors regulate, is described by transcriptional network. Hence, transcription factors regulate the rate at which the proteins encoded by genes are produced. These regulations can be positive-activation, and occur when a transcription factor helps increase the rate of transcription when it binds to a promoter, or negative-repression, which happens when a transcription factor helps reduce the rate of transcription when it binds to a promoter. In terms of node-edge representations of the network, each edge in a network is given a sign, + for activation and - for repression.

Some proteins are themselves transcription factors that can activate or repress other genes. The rate at which a gene is transcribed, the number of mRNAs produced per unit time, is controlled by the promoter, a regulatory region of DNA that precedes the gene. RNA polymerase (RNAp) binds a definite site (specific DNA sequence) at the

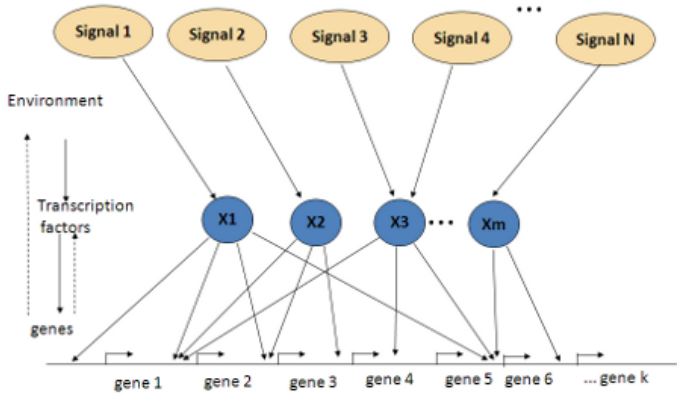



Figure 1.3: Transcription network. Figure adapted from (Alon, 2007a).

promoter. The quality of this site specifies the transcription rate of the gene (Alon, 2007b).

## 1.4 Network input functions

The strength of the effect of a transcription factor on the transcription rate of its target gene is described by an input function (Alon, 2007a). In terms of node-edge representations, the input function assigns the strength of the interaction to each individual edge in the graph. The simplest example is that in which the production rate of protein $Y$ is controlled by a single transcription factor $X$, symbolically denoted by $X \to Y$, and is described mathematically by:

$$\text{Rate of production of } Y = f(X). \tag{1.3}$$

where $f(X)$ is a monotonic function: in case when $X$ is activator, $f$ is an increasing function, and it is a decreasing function when $X$ is a repressor. A good approximation for many promoters is a Hill function:

$$f(X) = \frac{\beta X^n}{K^n + X^n}. \tag{1.4}$$

The Hill function has three parameters, $K$, $\beta$ and $n$. The first parameter, $K$, is an activation coefficient, and has units of concentration. It defines the concentration of active $X$ needed to significantly activate expression. Half-maximal repression is reached

when $X$ is equal to $K$. The value of $K$ is related to the chemical affinity between $X$ and its site on the promoter, as well as additional factors. The second parameter is the maximal expression level of the promoter, $\beta$. Maximal expression level is reached at high activator concentrations, $X \gg K$, because at high concentrations, $X$ binds the promoter with high probability and stimulates RNAp to produce many mRNAs per unit time. The Hill coefficient $n$ governs the steepness of the input function. The larger the $n$, the more step-like the input function. Typically, $n = 1 - 4$. The Hill function approaches a limiting value at high levels of $X$.

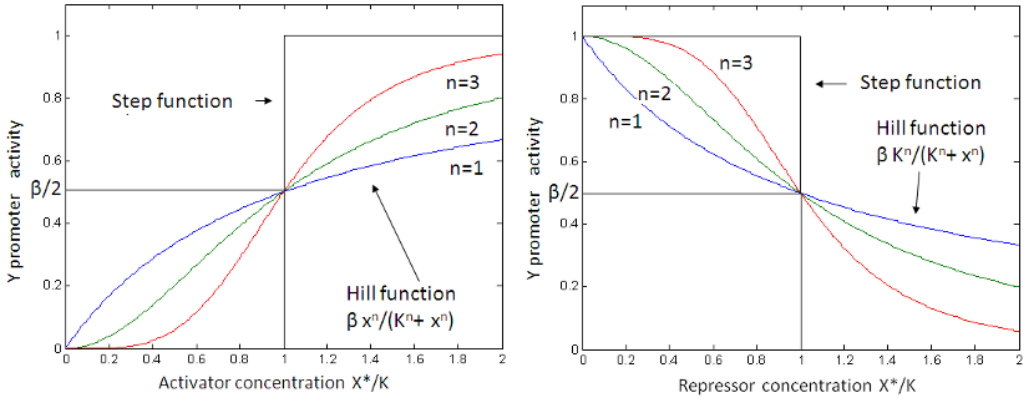



Figure 1.4: Input functions for (a) activator (b) repressor X described by Hill functions

For a repressor, the Hill input function is decreasing, and its shape depends on the same 3 parameters:

$$f(X) = \frac{\beta}{1 + (X/K)^n}. \tag{1.5}$$

Since a repressor allows strong transcription of a gene when it is not bound to the promoter, this function can be derived by considering the probability that the promoter is not bound by $X$. The maximal production rate $\beta$ is obtained when the repressor does not bind the promoter at all, that is when $X = 0$.

The input functions described range from a transcription rate of zero to a maximal transcription rate $\beta$. Many genes have non zero minimal expression level. This is called the genes' basal expression level, and can be described by adding a constant term $\beta_0$ to the input function.

## 1.5 Simple motifs in transcription networks

This section gives insight into the simplest motifs that Alon (2007a) introduces, and serves as a starting point for further chapters.

### 1.5.1 Network motif: Simple regulation

Simple regulation occurs when a transcription factor $Y$ regulates gene $X$ with no additional interactions, and is denoted by $X \rightarrow Y$. In the absence of its input signal, $X$ is inactive and $Y$ is not yet produced. When the signal $S_x$ appears, $X$ rapidly transits to its active form $X^*$ and binds the promoter of gene $Y$. Gene $Y$ begins to be transcribed, and mRNA is translated, resulting in the accumulation of protein $Y$. The cell produces protein $Y$ at a constant rate, $\beta$.

Production of $Y$ is balanced by two processes, protein degradation (its specific degradation by specialized proteins in the cell) and dilution (the reduction in its concentration due to the increase of cell volume during growth). The dynamics of $Y$ is described by its production rate $f(X) = \beta$, and degradation/dilution rate $\alpha$:

$$\frac{dY(t)}{dt} = \beta - \alpha Y. \tag{1.6}$$

At steady state, $Y$ reaches a constant concentration $Y_{st}$, solving for $\frac{dY(t)}{dt} = 0$, it is easy to see that the steady state concentration is the ratio of production and degradation/dilution rates. The higher the production rate, the higher the protein concentration reached, $Y_{st}$. If the input signal is removed, so the production of $Y$ stops, the concentration of $Y$ will exponentially decay:

$$Y(t) = Y_{st}e^{-\alpha t}. \tag{1.7}$$

The measure for the speed at which $Y$ levels change is the response time, which is defined as the time to reach half-way between the initial and final levels in a dynamic process. In this case response time is: $T_{0.5} = \log(2)/\alpha$. Degradation/dilution rate, $\alpha$, directly determines the response time: fast degradation/dilution allows rapid changes in

concentration. The production rate affects the steady state level, but not the response time.

Consider the opposite case in which an unstimulated cell with $Y = 0$ is considered, so that protein $X$ begins to accumulate. If an unstimulated gene becomes suddenly stimulated by a strong signal $S_x$, the dynamics equation results in an approach to steady state: $Y(t) = Y_{st}(1 - e^{-\alpha t})$. The concentration of $Y$ rises from zero and gradually converges on the steady state $Y_{st} = \beta/\alpha$. At early times when $\alpha t \ll 1$, accumulation of $Y$ can be found using Taylor expansion: $Y \sim \beta t,\ \alpha t \ll 1$ .

### 1.5.2 Network motif: Autoregulation

Autoregulation is regulation of a gene by its own gene product. To form a self-edge, an edge needs to choose its node of origin, and its destination out of $N$ possible target nodes. This probability is:

$$p_{self} = 1/N. \tag{1.8}$$

assuming $N$-node, $E$-edge network. Since $E$ edges are placed at random to form a random network, the probability of having $k$-self edges is approximately binomial:

$$P(k) = \binom{E}{k} p_{self}^{k} (1 - p_{self})^{E-k}. \tag{1.9}$$

The average number of self-edges is equal the number of edges $E$ times the probability than an edge is a self-edge:

$$< N_{self} >_{rand} \sim E p_{self} \sim E/N. \tag{1.10}$$

with a standard deviation that is approximately the square root of the mean:

$$\sigma_{rand} \sim \sqrt{E/N}. \tag{1.11}$$

The significant difference in the number of self-edges in real and randomized network can be described by the number of standard deviations by which the real network exceeds the random ensemble:

$$Z = \frac{< N_{self} >_{real} - < N_{self} >_{rand}}{\sigma_{rand}}. \tag{1.12}$$

The E. Coli transcription network has $N = 424$ and $E = 519$, and the corresponding ER network with the same $N$ and $E$ would be expected to have only about one self edge, plus minus one: $< N_{self} >_{rand} \sim E/N \sim 1.2$, $\sigma_{rand} \sim \sqrt{1.2} \sim 1.1$.

In contrast, the real network has 40 self-edges, which exceeds the random network by many standard deviations. In this example self-edges show $Z \sim 32$ which means they occur far more often than at random. Based on this example it can be concluded that self-edges are network motif.

### 1.5.3 Network motif: Negative autoregulation

Negative autoregulation (NAR) occurs when a transcription factor represses the transcription of its own gene. The self-repression occurs when $X$ binds its own promoter to inhibit production of mRNA. As a result the higher the concentration of $X$, the lower its production rate.

Dynamics of $X$ is described by its production and degradation rate:

$$\frac{dX(t)}{dt} = f(X) - \alpha X, \tag{1.13}$$

where $f(X)$ is a decreasing function of $X$.

Good approximation is decreasing Hill function:

$$f(X) = \frac{\beta}{1 + (X/K)^n}. \tag{1.14}$$

When $X$ is much smaller than the repression coefficient $K$, the promoter is free and the production rate reaches its maximal value $\beta$. On the other hand, when repressor $X$ is at high concentration, no transcription occurs, $f(X) \sim 0$. Coefficient $K$ is the repression coefficient, and defines the concentration at which $X$ represses the promoter activity by 50%.

The simplest case where $f(X) = \beta\theta(X < K)$ is used to analyze what happens to the dynamics of the network. To study response time, $X$ is initially absent and its production starts at $t = 0$. At early times when concentration of $X$ is low, the promoter is unrepressed, and production is full steam at a rate $\beta$ described by the production/degradation equation:

$$\frac{dX(t)}{dt} = \beta - \alpha X \quad \text{while} \quad X < K. \tag{1.15}$$

This result is the approach to a high steady state value. At early times degradation can be neglected ($\alpha X \ll \beta$) to find linear accumulation of $X$ with time:

$$X(t) \sim \beta t \text{ while } X < K \text{ and } X \ll \beta/\alpha. \tag{1.16}$$

However, production stops when $X$ levels reach the self-repression treshold, $X = K$. Small oscillations will occur if there are any delays in the system. Delays might cause $X$ to overshoot beyond $K$ slightly, but then production stops and $X$ levels decline until they decrease beyond $K$. These oscillations are damped if $f(X)$ is a Hill function.

Thus, $X$ effectively locks into a steady-state level equal to the repression coefficient of its own promoter:

$$X_{st} = K. \tag{1.17}$$

The resulting dynamics shows a rapid rise and a sudden saturation. The response time can be found by requiring that $X$ reaches half of its steady state. $X_{T_{0.5}} = X_{st}/2$. If linear accumulation of $X$ is used, where $X = \beta t$, the response time for negative autoregulation is given by:

$$T_{0.5}^{NAR} = \frac{K}{2\beta}. \tag{1.18}$$

So, NAR has been shown to display two important functions:

1. NAR speeds up the response time of gene circuits. This occurs when NAR uses a strong promoter to obtain a rapid initial rise in the concentration of protein $X$. When $X$ concentration reaches the repression threshold for its own promoter, the production rate of new $X$ decreases. Thus, the concentration of $X$ locks into a steady-state level that is close to its repression threshold. This steady-state level can be adjusted over evolutionary time by mutations that cause variation in the repression threshold of $X$ to its own promoter. However NAR can give overshoot during the initial rise. By contrast, a simply regulated gene that is designed to reach the same steady-state level must use a weaker promoter. As a result, an NAR system reaches 50% of its steady state faster than a simply regulated gene. The dynamics of NAR show a rapid initial rise followed

by a sudden locking into the steady state, possibly accompanied by an overshoot or damped oscillations.

2. NAR can reduce cell-cell variation in protein levels due to an inherent source of noise: the production rates of proteins fluctuate by tens of percents. This noise results in cell-cell variation in protein level. NAR can, in many cases, reduce these variations: high concentrations of $X$ reduce its own rate of production, whereas low concentrations cause an increased production rate. The result is a narrower distribution of protein levels than would be expected in simply regulated genes. However, if the NAR feedback contains a long delay, noise can also be amplified.

### 1.5.4 Network motif: Positive autoregulation

Positive autoregulation (PAR) occurs when a transcription factor enhances its own rate of production. The effects are opposite to those of NAR: response times are slowed and variation is usually enhanced. PAR slows the response time because at early stages, when levels of $X$ are low, production is slow. Production picks up only when $X$ concentration approaches the activation threshold for its own promoter. Thus, the desired steady state is reached in an S-shaped curve. The response time is longer than in a corresponding simple-regulation system. PAR tends to increase cell-cell variability. If PAR is weak (that is, $X$ moderately enhances its own production rate), the cell-cell distribution of $X$ concentration is expected to be broader than in the case of a simply regulated gene. Comparison of the dynamics of a negatively autoregulated gene, a positively regulated gene, and a simply regulated gene is given in Figure 1.5.

### 1.5.5 Network Motif: The Feed-forward loop

The feed-forward loop is composed of transcription factor $X$ that regulates a second transcription factor $Y$, and both $X$ and $Y$ regulate gene $Z$. Thus, the feed-forward loop has two parallel regulation paths, a direct path from $X$ to $Z$, and an indirect path that goes through $Y$. Each of the three edges can correspond to activation (plus sign) or repression (minus sign).

As in the previous sections, we are interested in knowing if feed-forward loop is a

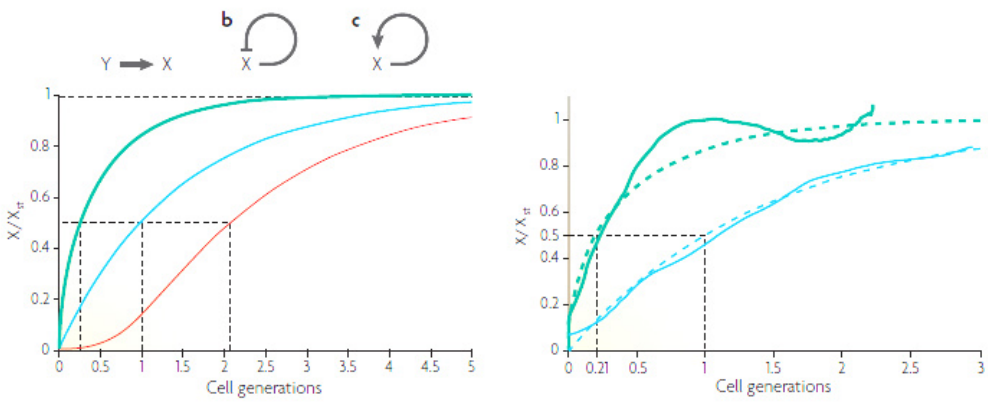



Figure 1.5: Dynamics of a negatively autoregulated gene, a simply regulated gene and a positively autoregulated gene, theoretical data (left), experimental data (right). Adapted from Alon (2007b)

network motif. The average number of times a given subgraph $G$ (with $n$ nodes and $g$ edges) appears in a random network is given by:

$$< N_G >= \frac{N^n p^g}{a}. \tag{1.19}$$

where $p$ is defined by Equation (1.2), and $a$ is a combinatorial factor related to the structure and symmetry of each subgraph (i.e, $a = 1$ FFL, $a = 3$ feedback loop). The formula denotes the number of ways of choosing a set of $n$ nodes out of $N$ for large networks is proportional to $N^n$ and multiplied by the probability to get $g$ edges in the proper places, each with probability $p$.

For the transcription network of E. coli in the Figure 1.6, it can be inferred that FFL is a network motif.

There are $2^3 = 8$ possible types of FFLs. These eight types can be classified into two groups: coherent and incoherent, based on comparing the sign of direct path from $X$ to $Z$, to the sign of indirect path that goes through $Y$. In the coherent FFL, the indirect path has the same overall sign as the direct loop. The overall sign of a path is given by multiplication of the signs of each arrow on the path. In the Incoherent FFL, the sign of a direct path is opposite to that of an indirect path. Not all the FFLs appear with equal frequency in transcription networks. The most abundant type is Type I Coherent FFL in which all three regulations are positive. To understand the

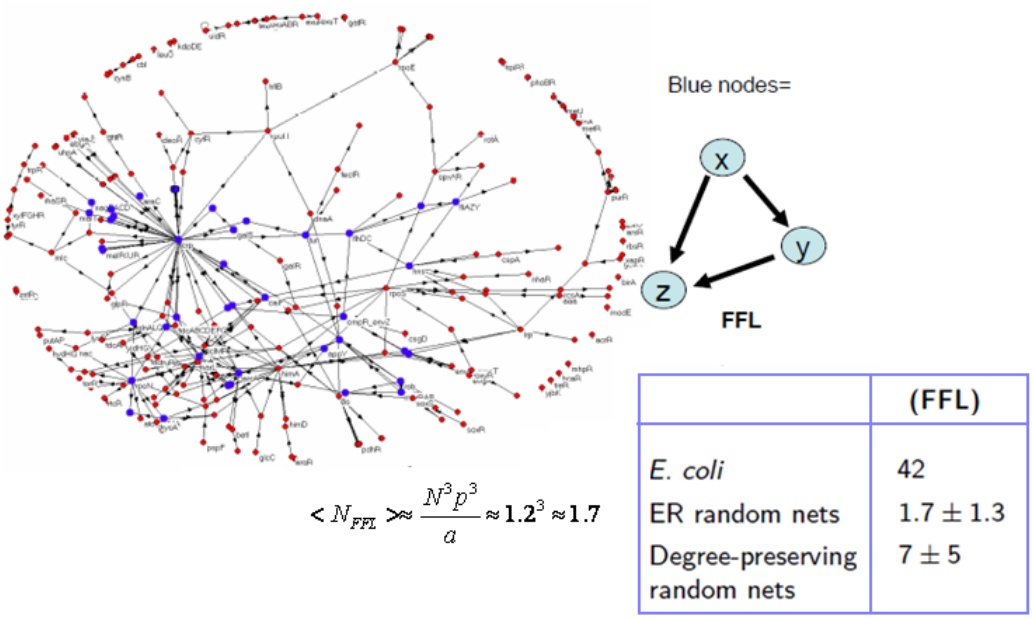



Figure 1.6: Transcription network of E coli containing FFL motif

dynamics of the FFLs it is necessary to know how the inputs from two regulators $X$ and $Y$ are integrated at a promoter of gene $Y$. Thus, we need to know the input function of gene $Z$. Two biologically reasonable functions are AND logic (where both $X$ and $Y$ activities need to be high to turn on $Z$ expression) and OR logic (where either $X$ or $Y$ is sufficient), as shown in Figure 1.9. The transcription factors $X$ and $Y$ usually respond to external stimuli (represented by the input signals $S_x$ and $S_y$).

The effect of the signals which carry information from the external world, usually operates at much faster timescales than the transcriptional interactions in the FFL. When $S_x$ appears transcription factor $X$ rapidly becomes active, $X^*$, binds to specific DNA sites in the promoters of genes $Y$ and $Z$ in a manner of seconds, and changes the transcription rate so that the concentration of protein $Z$ changes on the timescale of minutes to hours.

### Dynamics of the Coherent Type I FFL with AND logic

The input to transcription factor $X$ is the signal $S_x$, shown in Figure 1.8. Without the signal, $X$ is in the inactive form. At time $t = 0$, a strong step-like signal $S_x$ triggers the activation of $X$. As a result the transcription factor X rapidly transits to its active form

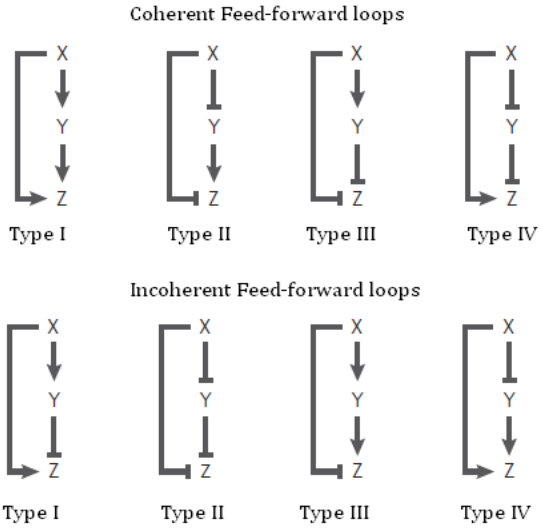



Figure 1.7: Types of Feed-forward loops: (a) -Coherent (b) -Incoherent

$X^*$. The active protein $X^*$ binds the promoter of gene $Y$, initiating the production of gene $Y$, the second transcription factor in the FFL. In parallel, $X^*$ also binds the promoter of gene $Z$. However, since the input function at the $Z$ promoter is AND logic, $X^*$ alone cannot activate $Z$ production. The production of $Z$ requires binding of both $X^*$ and $Y^*$. This means that the concentration of $Y$ must build up to sufficient levels to cross the activation treshold, $K_{yz}$ for gene $Z$. In addition, $Z$ activation requires that the second input signal, $S_y$, is present, so that $Y$ is in its active form, $Y^*$. Thus, once signal $S_x$ appears, $Y$ needs to accumulate in order to activate $Z$. This results in a delay in $Z$ production. To gain insight into simplest mathematical model behind this, logic input functions were used: the production of $Y$ occurs at a rate $\beta_y$ when $X^*$ exceeds the activation treshold, $K_{xy}$, as described by the step function:

$$\text{Production rate of } Y = \beta_y \theta(X^* > K_{xy}). \tag{1.20}$$

When the signal $S_x$ appears, $X$ rapidly shifts to active $X^*$. If the signal is strong enough, $X^*$ exceeds the activation treshold $K_{xy}$ and rapidly binds the $Y$ promoter to activate transcription. Thus, $Y$ production begins shortly after $S_x$. The accumulation of $Y$ is described by:

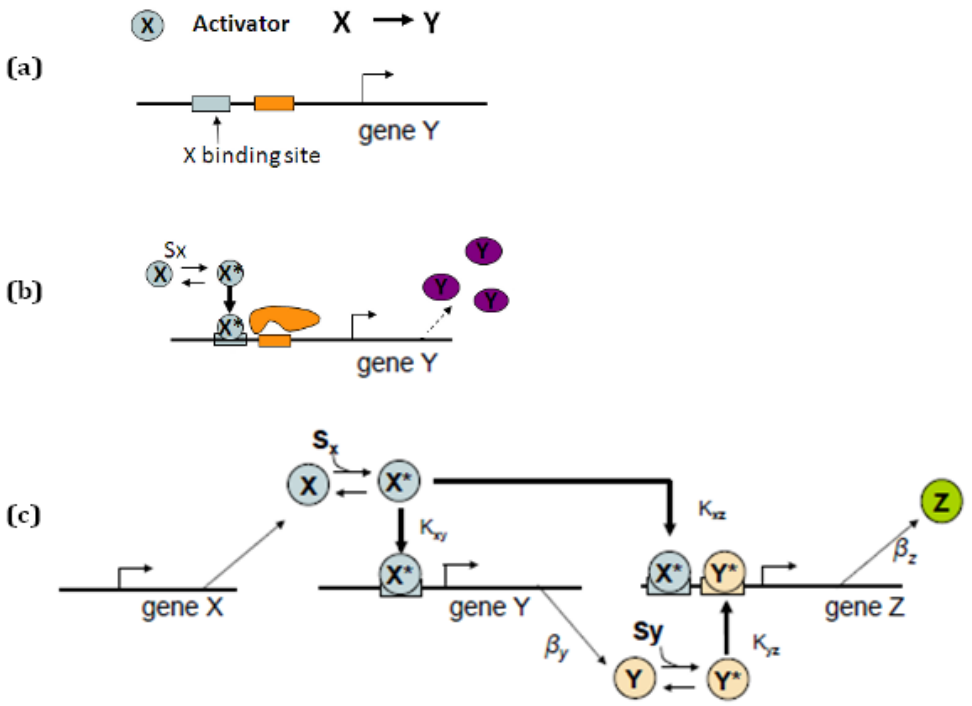



Figure 1.8: Molecular interactions in C1-FFL

$$\frac{dY(t)}{dt} = \beta_y \theta(X^* > K_{xy}) - \alpha_y Y. \tag{1.21}$$

Since the promoter of $Z$ is governed by an AND gate function, the production of $Z$ can be described by a product of two step functions, each indicating if the appropriate regulator crossed the activation treshold.

$$\text{production of } Z = \beta_z \theta(X^* > K_{xz}) \theta(Y^* > K_{yz}). \tag{1.22}$$

The delay in the production of $Z$ is due to time it takes $Y^*$ to accumulate and cross its treshold $K_{yz}$. Only after $Y^*$ crosses the treshold, can $Z$ production proceed at rate $\beta_z$. The dynamics of $Z$ are governed by a degradation/dilution term and production term with an AND input function:

$$\frac{dZ(t)}{dt} = \beta_z \theta(X^* > K_{xz}) \theta(Y^* > K_{yz}) - \alpha_z Z. \tag{1.23}$$

Figure 1.10 considers response to steps of $S_x$, in which signal $S_x$ is first absent and then

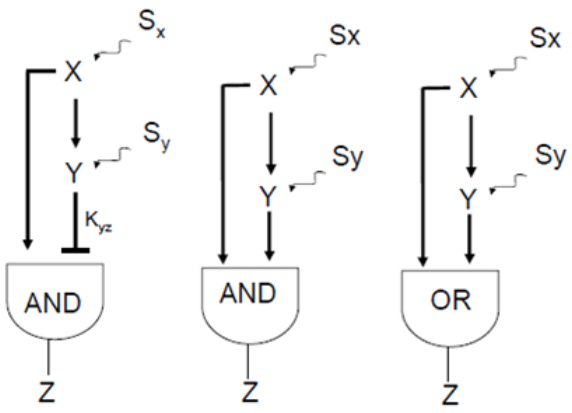



Figure 1.9: Input functions to I1-FFL and C1-FFL governed by AND and OR logic

saturating $S_x$ suddenly appears (ON steps). Following an ON step of $S_x$, $Y^*$ begins to be produced at rate $\beta_y$. Concentration $Y$ begins to exponentially converge to its steady state level.

$$Y^*(t) = Y_{st}(1 - e^{-\alpha_y t}). \tag{1.24}$$

The delay in expression of $Z$ can be found as:

$$\begin{aligned} Y^*(t) &= Y_{st}(1 - e^{-\alpha_y T_{ON}}) = K_{yz} \\ T_{ON} &= 1/\alpha_y \log[1/(1 - K_{yz}/Y_{st})]. \end{aligned} \tag{1.25}$$

If $S_x$ is suddenly removed, following an OFF-step, $X$ rapidly becomes inactive and unbinds from the promoters of genes $Y$ and $Z$. Since $Z$ is governed by an AND gate, it takes only one input to go off for the AND gate to stop $Z$ expression, as shown in Figure 1.10. There is no delay in $Z$ dynamics after an OFF step. To summarize, $Z$ shows delay following ON-step of $S_x$ and has no delay in case of OFF-step of $S_x$, and this kind of behavior is called sign-sensitive delay, because the delay depends only on the sign of the step. This delay property is a good mean of protection against brief input fluctuations: a brief pulse of $S_x$ does not give $Y$ enough time to accumulate and cross its activation threshold for $Z$. Hence, $Z$ is not expressed. A persistent pulse yields $Z$ expression at a delay, as shown in Figure 1.11. Therefore, C1-FFL serves also as a persistence detector. In the paper by Mangan and Alon (2003), is has been shown that all coherent FFL circuits serve as sign sensitive delay element to ON steps of $S_x$, but not for OFF steps of $S_x$. Moreover, the authors have concluded that only types I and

II are responsive to $S_y$ inputs.

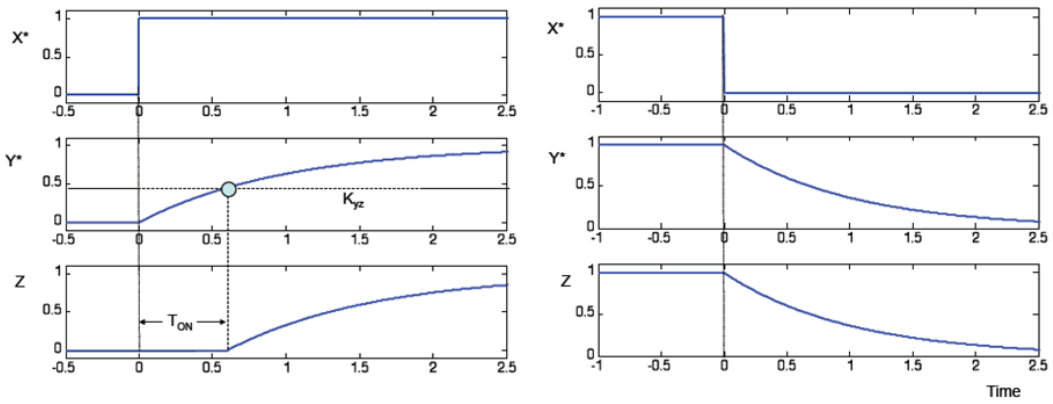



Figure 1.10: Dynamics of Coherent Type I FFL with AND logic

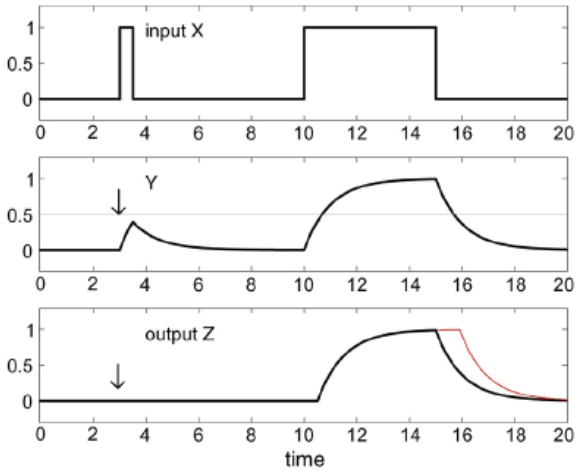



Figure 1.11: C1-FFL is a persistence detector

### Dynamics of the Incoherent Type I FFL

The I1-FFL is made up of two parallel antagonistic regulatory paths. In this circuit, activator $X$ activates $Z$, but also activates $Y$- a repressor of $Z$. The gene $Z$ shows high expression when the activator $X^*$ is bound and much weaker expression when the repressor $Y^*$ binds. Upon a step of $S_x$, protein $X$ becomes activated, binds the promoter of gene $Z$, causing protein $Z$ to be transcribed. In parallel, $X$ activates the production of $Y$. Therefore, after a delay enough protein $Y$ accumulates to repress $Z$

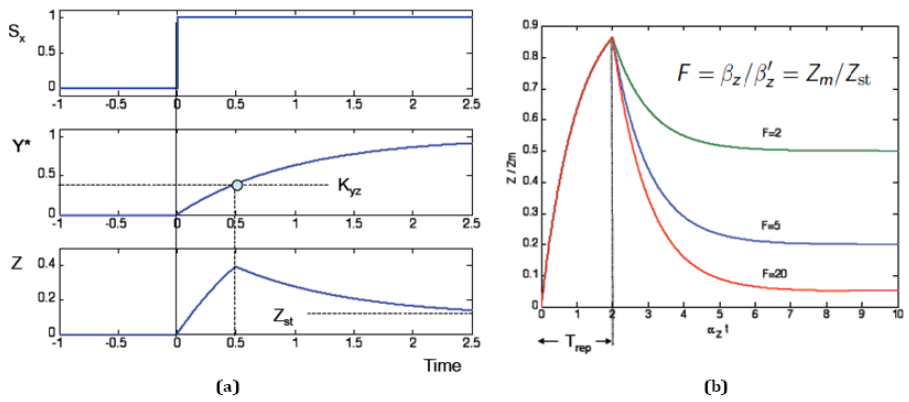



Figure 1.12: (a) Dynamics of I1-FFL with AND logic (b) I1-FFL can serve as pulse generator

production so that $Z$ levels decrease. Therefore, the I1-FFL can generate a pulse of $Z$ production as shown on (b) panel of Figure 1.12.

Consider the response to a step addition of the signal $S_x$ in the presence of another signal $S_y$. When the signal $S_x$ appears, protein X rapidly transforms to its active form, $X^*$. The active transcription factor $X^*$ binds its DNA site in $Y$ promoter within seconds, and $Y$ begins to be produced. Since $S_y$ is present, the protein $Y$ is in its active form, $Y^*$, and accumulates over time according to the production and degradation equation:

$$\frac{dY^*(t)}{dt} = \beta_y - \alpha_y Y^*. \tag{1.26}$$

Hence Y shows exponential convergence to its steady state $Y_{st} = \beta_y/\alpha_y$

$$Y^*(t) = Y_{st}(1 - e^{-\alpha_y t}). \tag{1.27}$$

Molecules of $X^*$ also bind the $Z$ promoter and protein $Z^*$ begins to be produced at a rapid rate $\beta_z$, since its promoter is occupied by the activator $X^*$, but there is not yet enough repressor $Y^*$ in the cell to inhibit production. In this phase:

$$\frac{dZ^*(t)}{dt} = \beta_z - \alpha_z Z, \tag{1.28}$$

and $Z$ accumulates, beginning an exponential convergence to a high level

$$\begin{aligned} Z_m &= \beta_z/\alpha_z \\ Z(t) &= Z_m(1 - e^{-\alpha_z t}) \quad \text{while } Y^* < K_{yz}. \end{aligned} \tag{1.29}$$

The fast production of $Z$ lasts until the repressor $Y^*$ crosses its repression treshold for $Z$, $K_{yz}$. Then, the production rate of $Z$ suddenly drops to a low value $\beta'_z$. The onset of repression occurs at the moment that $Y^*$ reaches $K_{yz}$. Repression time, $T_{rep}$ can be found by setting $Y^*(t) = K_{yz}$ obtaining:

$$T_{rep} = 1/\alpha_y \log[1/(1 - K_{yz}/Y_{st})]. \tag{1.30}$$

At times after $T_{rep}$ the $Z$ promoter is bound by the repressor $Y^*$ and the production rate is reduced. $Z$ concentration decays exponentially to a new lower steady state:

$$\begin{aligned} Z_{st} &= \beta'_z/\alpha_z \\ Z(t) &= Z_{st} + (Z_0 - Z_{st})e^{-\alpha_z(t-T_{rep})}, \end{aligned} \tag{1.31}$$

where $Z_0$ is the level reached at time $T_{rep}$ given by equation (1.33) at t=$T_{rep}$

$$Z_0 = Z_m + (1 - e^{-\alpha_z T_{rep}}), \tag{1.32}$$

and $Z_{st}$ is the final steady state $Z$ level due to low repression level when both $X^*$ and $Y^*$ bind the $Z$ promoter. The shape of dynamics generated by I1-FFL depends on $\beta'_z$, the basal production rate of $Z$. The basal rate corresponds to the low rate of transcription from the repressed promoter. Introducing the repression factor $F$, as ratio of the maximal and basal activity of the $Z$ promoter, also equal to the ratio of unrepressed and repressed steady state levels of $Z$:

$$F = \beta_z/\beta'_z = Z_m/Z_{st}. \tag{1.33}$$

When the repressor has a strong inhibitory effect on $Z$ production, that is when $F \gg 1$, $Z$ dynamics shows a pulse-like shape. In the pulse, $Z$ levels first increase than decrease to a low level. The I1-FFL acts, therefore, like a pulse generator, shown on (b) panel of Figure 1.12. In the paper by Mangan and Alon (2003), the authors have shown that only types III and IV of the Incoherent FFL with AND logic are good pulsers.

In the paper by Kim *et al.* (2008), the authors argue the biphasic behavior of IFFL in the sense that dose-dependent and time-dependent behavior have mutually exclusive dynamics. Time-dependent behavior is necessary for achieving transients important to control cell fate and decisions, therefore it can be thought as of a pulse generator. As

time evolves, the output response initially increases but subsequently decreases, even if the input is sustained. Dose-dependent biphasic behavior can be thought of as a "band-pass filter", since the steady state output response initially increases and subsequently decreases when increasing input dose. The dose- dependent biphasic response is required when an output responses only to a certain range of input dose strengths. Using Genetic Algorithm simulations, the authors showed that the dynamics of the two types are mutually exclusive.

### 1.5.6 Network motifs in other biological networks

So far, sensory transcription networks that respond rapidly and make reversible decisions were analyzed. Developmental transcription networks transduce signals into cell-fate decisions, and have different constraints: they usually function on the timescale of one or several cell generations, and often need to make irreversible decisions that last even after the input signal has vanished. Developmental transcription networks use all the network motifs described above. In addition, as a result of their specific requirements, developmental networks use several other network motifs that are not commonly found in sensory networks; they often use positive feedback loops that are made up of two transcription factors that regulate each other. In addition to transcription networks, one can seek composite network motifs that include different types of interactions. One of the most common composite motifs is a negative feedback loop between two proteins, in which one arm is a transcriptional interaction and the other arm is a protein-protein interaction. Composite negative feedback loops seem to be much more common than purely transcriptional negative feedback loops. The separation of timescales between the slow transcription arm and the faster protein interaction arm might help to stabilize the dynamics of composite loops, avoiding feedback at a delay that promotes instability. Experiments on individual living cells have shown that negative feedback loops, embedded within additional interactions, can sometimes generate oscillations, whereby the levels of X and Y rise and fall.

## 1.6 Other perspective on modeling cellular response in complex regulatory networks

In the paper by Tyson *et al.* (2003), mechanisms of protein synthesis and degradation, and phosphorylation and dephosphorylation were put in mass-action rate equations and linear and hyperbolic response of protein and phosphorylated form were obtained. If phosphorylation and deposhorylation were governed by Michelis-Menten kinetics, switch like signal response curve was generated, called zero-order ultrasensitivity, using Goldbeter-Koshland function (Goldbeter and Jr, 1984). Another type of response called sniffer, as an analogy to the sense of smell, can be implemented by casting the simple linear element with a second signaling pathway creating a mechanism that achieves perfect adaptation to the signal. "Sniffer" model was further expored in (Sontag, 2008), where it is shown that this circuit is an Incoherent feed-forward loop. Similarly, simple pathways can be embedded in a network using positive and negative feedback to create homeostasis and oscillations or irreversible switches (using positive feedback).

## 1.7 Conclusion

Based on the knowledge of the basic properties of the simplest network motifs that are present in a complex biological system, further chapters will deal with connecting recurrent motifs with a particular cellular function. Moreover, concepts of network adaptation and fold change detection, as biologically meaningful functions, will be analyzed on an example of 3-node enzymatic networks.

# Chapter 2

# Robustness and adaptation of biological circuits. Fold change detection.

## 2.1 Robustness principle in cellular networks

Properties of biological circuits depend on the biochemical parameters of its components, for example concentration of proteins that are building elements of the network. Cellular biochemical networks are highly interconnected: a perturbation in the reaction rates or molecular concentrations may affect numerous cellular processes (Barkai and Leibler, 1997). Due to stochastic fluctuations in the cell, parameters vary significantly from cell to cell, even if the cells are identical (Alon, 2007a). Having in mind the complexity of cellular networks, the perturbation in parameters of the network raises the questions of stability of the network. Yet, biological systems function reliably despite all these perturbations. One possibility to achieve this function is by "fine-tuning" of the reaction rate constants and enzyme concentrations, but any deviation from these values will ruin network's behavior (Alon *et al.*, 1999). Alternative is, that cell's key parameters are itself, "robust", meaning that they are insensitive to precise values of biochemical parameters, therefore fluctuations will not significantly influence cell's behavior. In the paper by Savageau (1971), robustness was introduced as an important design principle in the analysis of gene circuits. Cellular systems have robust designs such that their essential function is nearly independent of biological parameters that vary from cell to cell (Alon, 2007a). So in this context, robustness is used with respect to parameters. Properties that are not robust are called fine-tuned, and they change when biochemical parameters are varied.

### 2.1.1 Robustness principle on an example of Bacterial Chemotaxis in E. Coli

Chemotaxis is the ability to move toward nutrients (positive chemotaxis) and away from noxious compounds (negative chemotaxis). E. coli bacteria are single-cell organisms, which possess up to six flagella for movement. Flagella are rotary motors powered by ion gradients across the plasma membrane. This motor drives a helical propeller. The basic motor can rotate in either a clock-wise or a counter clock-wise direction. When the flagellum rotates counter clock-wise, "run" mode occurs, flagella form a bundle which helps the bacteria move forward; when rotating clock-wise, the bacteria tumbles. Tumbling allows the bacteria to change direction. When the motor switches back to the counter clock-wise direction, the bacteria swims off in a new direction. Chemicals that attract bacteria to swim towards them are called attractants (i.e sugars), and those that drive bacteria away are called repellents. To sense concentration gradients, E.Coli compares the current attractant concentration to the concentration in the past. When E. coli moves up a gradient of attractant, it detects net positive change in attractant concentration. As a result it reduces the probability of a tumble (it reduces its tumbling frequency) and tends to continue going up the gradient. Similar logic applies to repellents. If the bacterium detects that the concentration of repellent increases with time, the cell increases its tumbling frequency, and thus tends to change direction and avoid swimming toward repellents. Thus, chemotaxis senses the temporal derivative of the concentration of attractants and repellents (Alon, 2007a).

Experimentally, bacteria were observed under a microscope swimming in a liquid with no gradients (Alon, 2007a). The cell displays runs and tumbles, as shown in Figure[1] 2.1. Then the attractant was uniformly added to the liquid, increasing the attractant concentration, but no spatial gradients were formed. The cells sense the increase in attractant levels, no matter which direction they are swimming in, so they suppress tumbling. After a while, tumbling frequency of the cells begins to increase, even though the attractant is still present. This mechanism is adaptation. In the case of

[1]Adapted from Alon *et al.* (1999)

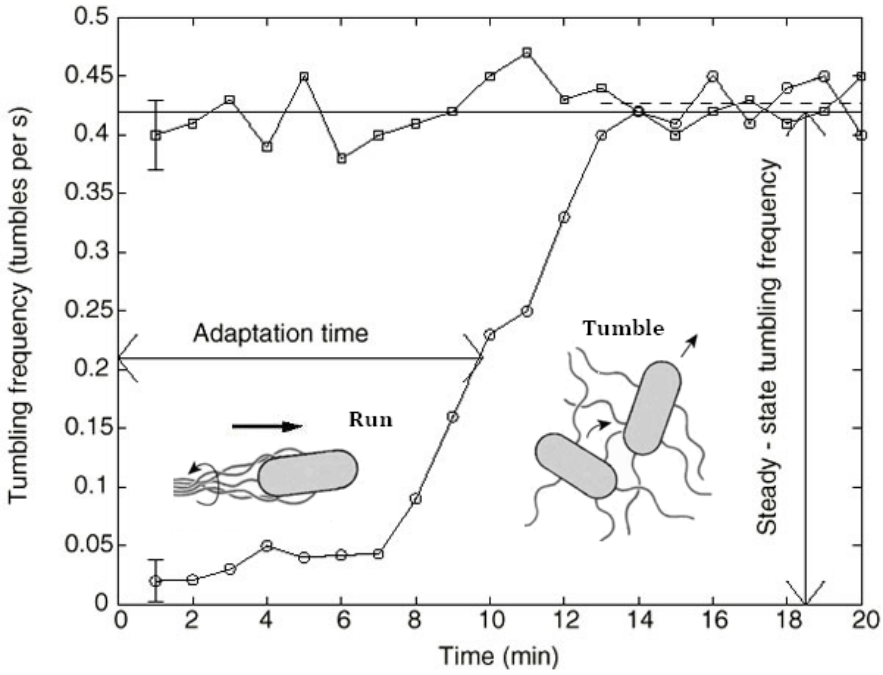



Figure 2.1: Movement in E. Coli, adapted from Alon et. al., Nature, 1999

bacterial chemotaxis, this adaptation is exact adaptation: the tumbling frequency in the presence of the attractant returns to the same level as before the attractant was added. In other words, the steady state tumbling frequency is independent of attractant level. If more attractant is added, the cells show a decrease in tumbling frequency, followed by exact adaptation.

### 2.1.2 Mechanism of exact adaptation by Barkai and Leibler

A mechanism that allows exact adaptation for a wide range of parameters was proposed by Barkai and Leibler. Attractant and repellent compounds are sensed by the cell proteins called receptors. Attractant or repellent molecules bound by the receptor are called ligands. Detecting elevated levels of chemo-attractant decreases the probability of a tumble, thus propelling the bacteria in the favorable direction. This modulation of the length of runs is mediated by a signal transduction pathway consisting of transmembrane receptors (methyl-accepting proteins) and the products of 6 Che genes: *cheA*, *cheB*, *cheR*, *cheW*, *cheY*, and *cheZ*. The receptor forms a complex with the protein CheW and the kinase *CheA*. *CheA* phosphorylates itself and then transfers phosphoryl

group to the messenger protein $CheY$, and this phosphorylated form, $CheY_P$, stimulates tumbling by interacting with the flagellar motor. When chemo-attractant binds receptor, $CheA$ activity is suppressed, the levels of $CheY_P$ decrease, and the bacterium is less likely to tumble. Adaptation results from the methylation of receptor by enzyme $CheR$, which increases $CheA$ activity, promoting $CheY$ phosphorylation. The methylation state of the receptor is balanced by the demethylation enzyme $CheB$. A feedback mechanism is achieved through the $CheA$ mediated phosphorylation of $CheB$, which enhances its demethylation activity.

### 2.1.3 Model of chemotaxis network by Barkai and Leibler

Barkai and Leibler analyzed a simple two-state model of the chemotaxis network (Barkai and Leibler, 1997). The two state model assumes that the receptor complex has two functional states: active and inactive, although with probabilities that depend on both their methylation level and ligand occupancy. The active receptor complex shows a kinase activity: it phosphorylates the response regulator molecules, which then bind to the motors and induce tumbling. The average complex activity can be considered as the output of the network, whereas its input is the concentration of the ligand. When a typical model system is subject to a step-like change in attractant concentration, it is able to respond and to nearly perfectly adapt for all ligand concentrations. The addition (removal) of attractant causes a transient decrease (increase) in system activity, and thus of tumbling frequency.

The adaptation is a robust property of the chemotactic network, and can be described by various characteristics, such as steady-state tumbling frequency, precision of adaptation, sensitivity of adaptation, adaptation time, etc. However, not all the properties are equally insensitive to variations in the network parameters. Robustness is thus a characteristic of specific network properties and not of the network as a whole, whereas some properties are robust, others can show sensitivity to changes in the network parameters.

Consider an enzyme, $E$, which is sensitive to an external signal $l$, such as a ligand, as shown in Figure 2.2. The signal level $l$ affects the equilibrium between two functional

states of the enzyme (active or inactive): a change in $l$ causes a rapid response of the system by shifting this equilibrium. Thus, $l$ is the input of this signal transduction system and the concentration of active enzymes, $A$, can be considered as its output. The enzyme $E$ can be reversibly modified, i.e., by addition of methyl or phosphate groups. The modification of $E$ affects the probabilities of the active and inactive states, and hence can compensate for the effect of the ligand. In general:

$$A(l) = \alpha(l)E + \alpha_m(l)E_m, \tag{2.1}$$

where $E_m$ and $E$ are the concentrations of the modified and unmodified enzyme, respectively, and $\alpha_m(l)$ and $\alpha(l)$ are the probabilities that the modified and unmodified enzyme is active.

After an initial rapid response of the system to a change in the input level, $l$, slower changes in the system activity proceed according to the kinetics of enzyme modification.

The system adapts when its steady-state activity, $A_{st}$, is independent of $l$. A mechanism for adaptation can be obtained by assuming a fine-tuned dependence of the biochemical parameters on the signal level, $l$. A mechanism for robust adaptation can be obtained when the rates of the modification and the reverse-modification reactions depend solely on the system activity, $A$, and not explicitly on the concentrations $E_m$ and $E$. This system can be viewed as a feedback system, in which the output $A$ determines the rates of modification reactions, which in turn determine the slow changes in $A$. Therefore, the value of the steady-state activity, $A_{st}$, is independent of the ligand level, and system is adaptive.

Activity-dependent kinetics can be achieved in a variety of ways. As a simple example, consider a system for which only the modified enzyme can be active ($\alpha = 0$); the enzyme $R$, which catalyzes the modification reaction $E \to E_m$, works at saturation, and the enzyme $B$, which catalyzes the reverse-modification reaction $E_m \to E$, can only bind to active enzymes. In this case, the modification rate is constant at all times, whereas the reverse modification rate is a simple function of the activity:

$$\frac{dE_m(t)}{dt} = V^R_{max} - V^B_{max}\frac{A}{A + K_b}, \tag{2.2}$$

where $V^R_{max}$ and $V^B_{max}$ are the maximal velocities of the modification and the reverse-modification reactions, respectively, and $K_b$ is the Michaelis-Menten constant for the reverse modification reaction; we have assumed $V^R_{max} < V^B_{max}$, and that enzymes follow Michaelis-Menten (quasi steady-state) kinetics. The functioning of the feedback can now be analyzed: the system activity is continuously compared to a reference steady-state value:

$$A^{st} = K_b \frac{V^R_{max}}{V^B_{max} - V^R_{max}}. \tag{2.3}$$

For $A < A_{st}$, the amount of modification increases, leading to an increase in $A$; for $A > A_{st}$, the modification decreases, leading to a decrease in $A$. In this way, the system always returns to its steady state value of activity, exhibiting adaptation. Moreover, with these activity-dependent kinetics, the adaptation properties is insensitive to the values of system parameters (such as enzyme concentrations), so adaptation is robust.

However, the steady-state activity itself, which is not a robust property of the network, depends on the enzyme concentrations. Thus, the mechanism presented here still provides a way to control the system activity on long time scales, for example by changing the expression level of the modifying enzymes while preserving adaptation itself on shorter timescales.

Control system that tracks a specific steady-state output value under input disturbance is a problem commonly used in control theory, and corresponds to the Barkai and Leibler model 2.2. In the Barkai-Leibler model chemo-attractant is the input, receptor activity is the output. We can obtain the equation that characterizes the integral control as:

$$\begin{aligned} \dot{x} &= Ax + bu + w \\ \dot{q} &= y = Cx \end{aligned} \tag{2.4}$$

The augmented system is given by:

$$\begin{aligned} \begin{bmatrix} \dot{x} \\ \dot{q} \end{bmatrix} &= \begin{bmatrix} A & 0 \\ C & 0 \end{bmatrix} \begin{bmatrix} x \\ q \end{bmatrix} + \begin{bmatrix} w \\ 0 \end{bmatrix} + \begin{bmatrix} b \\ 0 \end{bmatrix} u \\ u &= -kx - k_q q + v, \quad \text{where } q = C \int x \end{aligned} \tag{2.5}$$

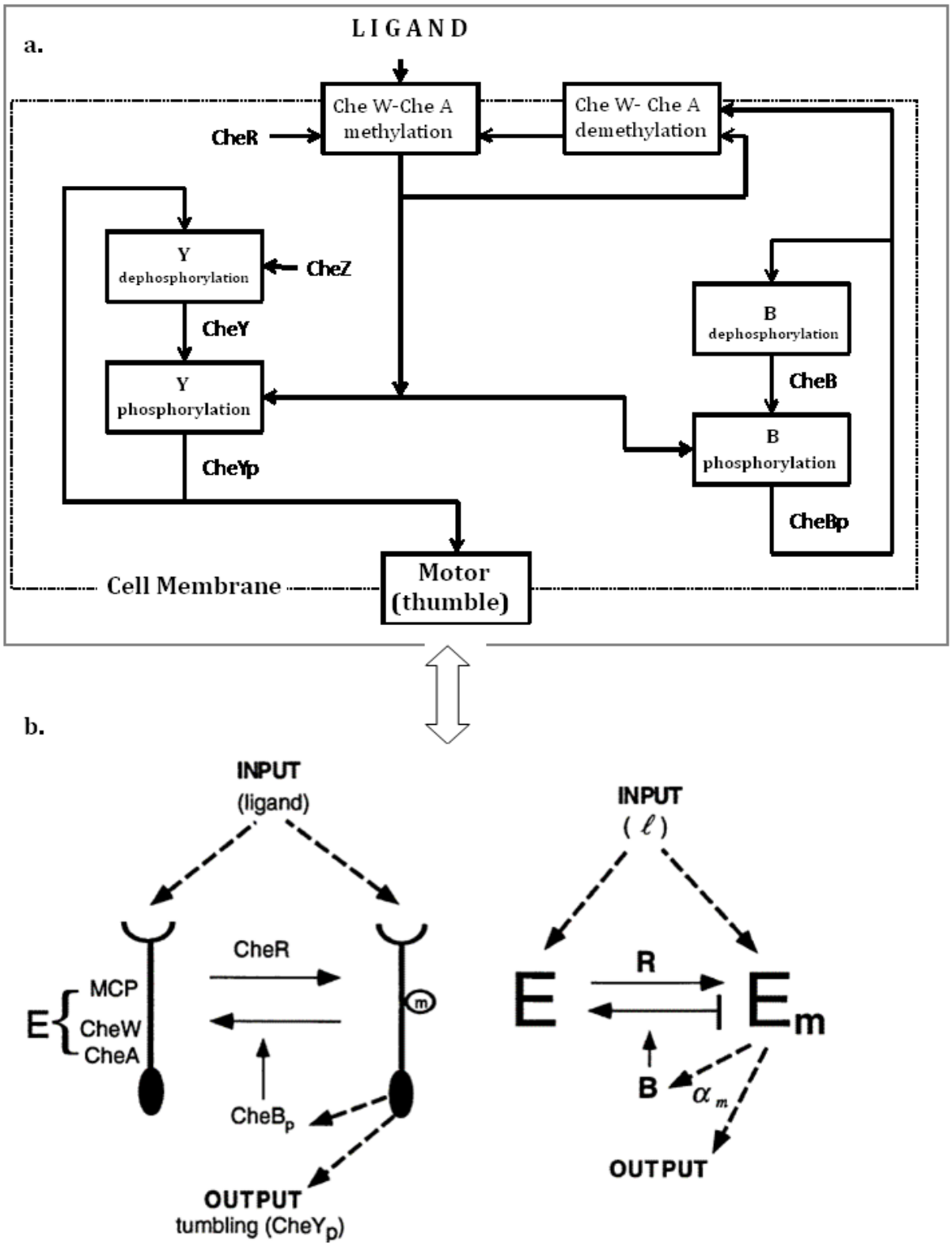



Figure 2.2: a. Block sheme b. The Barkai-Leibler model

In Figure 2.3, block diagram for a simple example of an integral feedback control is given. The paper by Yi *et al.* (2000) discusses the Barkai-Leibler model from a control theory point of view, in more detail.

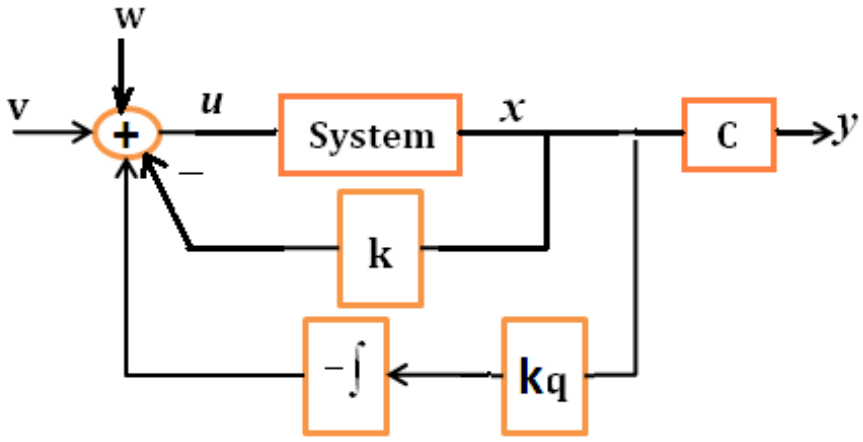



Figure 2.3: A block diagram of an integral feedback control for disturbance rejection

## 2.2 Fold Change Detection

So far, concept of adaptation to the same steady state was considered. On the other hand, the study of transient responses is of a great importance as well, since behavior at the time-scale of signaling may have important consequences for cell survival (Shoval *et al.*, 2010a). Studies by Uri Alon, Marc Kirschner, Berg and others suggest that certain mamalian signaling systems such are ERK2 activation in EGF-stimulated carcinoma cells, and Wnt signaling in cell lines and embryos, as well as in Escherichia coli and possibly other prokaryotic chemotaxis pathways display Fold Change Detection, a response whose entire shape, including amplitude and duration, depends only on fold changes in input, and not on absolute changes (Shoval *et al.*, 2010b; Goentoro *et al.*, 2009). Goentoro *et al.* (2009) consider gene regulation networks composed of different network motifs, such as negative autoregulation, positive autoregulation, single input module, coherent and incoherent feed-forward loops, and show that among them only Incoherent feed-forward loop motif possesses Fold Change Detection property, assuming activator in this motif operates in linear regime and repressor saturates the promoter of target gene. Additionally, FCD provides ability to maintain sensitivity despite the noise in the activator activity. Shoval *et al.* (2010b) show further that although integral feedback in the linear form doesn't provide FCD, nonlinear feedback and logarithmic input with linear feedback both exhibit Fold Change Detection property.

### 2.2.1 Scaling Invariance

Suppose that a system that adapts has had a chance to "pre-adapt" to a certain constant ("background") level $\bar{u}$ of the input, for $t < 0$, and that now the system is presented with the new input $u(t)$, $t \geq 0$. Let $y_1(t)$ be the output function that results. Now the same system is allowed to pre-adapt, to $p\bar{u}$ for $t < 0$, and then the system is presented with $pu(t)$, for $t \geq 0$, where $p$ is some positive scalar. Let $y_2(t)$ be the resulting output. Scale invariance means that the outputs of the two experiments should be the same: $y_1(t) = y_2(t)$ for all $t > 0$. In other words, for any two inputs $u(t)$ and $pu(t)$, as shown

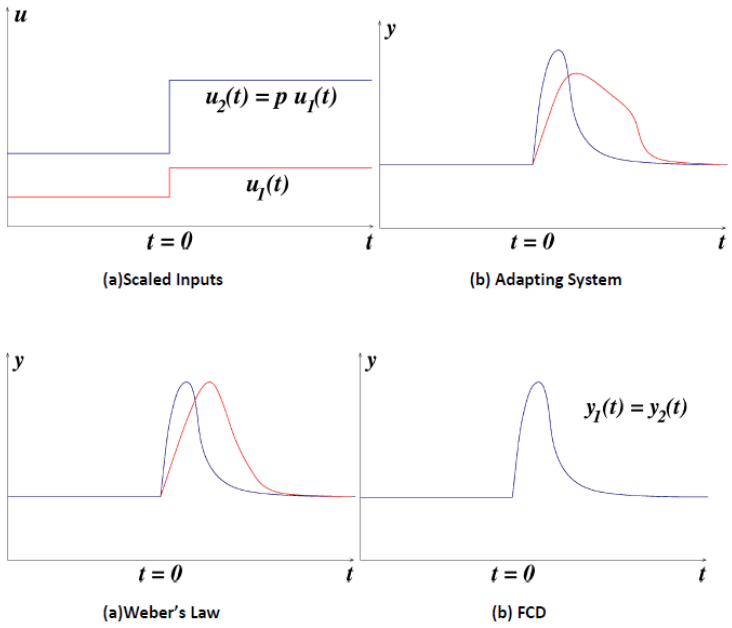



Figure 2.4: Adaptation and FCD

on Figure[1] 2.4, and no matter what positive number $p$ is picked, the entire shape of the response, including amplitude and duration, is identical. Thus, a step change in input from, say, level 1 to 2, gives precisely the same dynamical output as a step from level 2 to 4, since the steps have the same fold change ($p = 1/2$) (Shoval *et al.*, 2010a). FCD represents a particular type of adaptation with robustness to scaling of the input. A weak version of FCD is Weber's logarithmic sensing law: many sensory systems (for weight, sound intensity and pitch, and light intensity) produce responses whose maximal amplitude only depends on the ratio between the stimulus and a background or starting value. Weber hypothesized that the ratio between the smallest perceptual change in a stimulus ($\Delta S_{min}$) and the background level of the stimulus ($S_{background}$) is constant: $\frac{\Delta S_{min}}{S_{background}} = k$. In a Weber's law world, stimuli are perceived not in absolute terms, but in relative terms- fold change in magnitude relative to the background level of stimulus (Ferrell, 2009).

FCD entails exact adaptation and Weber's law, but is not guaranteed by having both (Shoval *et al.*, 2010b). FCD entails exact adaptation, because FCD by definition

[1] Adapted from (Shoval *et al.*, 2010a)

means that for any two constant inputs $u_1$ and $u_2 = pu_1$, the steady-state output must be the same. Exact adaptation does not entail FCD, as shown in Figure 2.4 on panel (b). FCD also generally entails Weber's law, however, Weber's law (even together with exact adaptation) does not necessarily entail FCD, as shown in Figure 2.4. In this example, amplitude depends on relative change in input as in Weber's law, but FCD is not found because the adaptation time varies with the absolute input strength (Shoval *et al.*, 2010a).

### 2.2.2 Sufficient Condition for Fold Change Detection (FCD)

The theory of FCD was first introduced in paper by Shoval *et al.* (2010b).

Consider a system with:

$$\begin{aligned} \dot{x} &= f(x, y, u) \\ \dot{y} &= g(x, y, u) \end{aligned} \tag{2.6}$$

where $x$ represents internal variables, $u$ input signal and $y$ output signal. FCD holds if system (2.6) is stable, shows exact adaptation to a steady state output $y = y_0$, and if $g$ and $f$ satisfy the following homogeneity conditions for any $p > 0$:

$$\begin{aligned} f(px, y, pu) &= pf(x, u, y) \\ g(px, y, pu) &= g(x, u, y) \end{aligned} \tag{2.7}$$

If $f$ is linear then this condition is also necessary for FCD (Shoval *et al.*, 2010b).

## 2.3 Conclusion

In this Chapter, two biologically important functions were introduced, exact adaptation and Fold Change Detection, as an extension of an exact adaptation with additional conditions. Experimental results motivated further investigation of the latter phenomenon in different biological systems, and in Chapters 3 and 4, we will test this property on enzymatic networks. Using the knowledge of the properties of different network motifs, we will give mathematical explanation of the obtained results.

# Chapter 3

# Three-node enzymatic networks that can achieve perfect adaptation

## 3.1 Introduction

In the previous chapters, two important concepts were introduced: network motifs as patterns (sub-graphs) of interconnections occurring in complex networks much more often then expected at random (Alon, 2007a), which will be used as a tool in order to understand the dynamics of the entire network based on the dynamics of its core elements, motifs, and concept of perfect adaptation as a system's ability to respond to a change in the input stimulus and then return to its pre-stimulated level, even when the change in input persists (Ma *et al.*, 2009). Additionally, fold change detection was introduced as a special property of adapting systems, where the output is invariant under the scaling of input (Shoval *et al.*, 2010a,b; Goentoro *et al.*, 2009). In the following two chapters, we will analyze all possible 3-node enzymatic networks.

In chapter 3, the paper by Ma *et al.* (2009) is used as a main reference, and adaptation results obtained by the authors are analyzed. All their simulations and calculations are redone. Here the authors were looking for the simplest motifs contained in all the networks that are capable of achieving adaptation, and their results indicated that such networks fall into two architectural classes: Negative FeedBack Loop with a Buffering node (NFBLB) and Incoherent Feed-Forward Loop with a Proportioner node (IFFLP). More complex circuits that are able to achieve adaptation, all contain at least one of these topologies at their core. These results are used as a starting point and motivation for this thesis.

In Chapter 4, we will test Fold Change Detection property, (FCD) on all the adaptive circuits from Chapter 3, and analyze the obtained results. We will give analytical

explanation of their behavior in Chapter 4, and also provide comparison of the analytical results with obtained numerical solutions, for all the circuits that satisfy the property, in the Appendix of this thesis. Additionally, we captured another interesting behavior among non FCD circuits, namely, shifted version of exact FCD, where the transient behavior and exact adaptation are satisfied with a delay. In future research, we will further analyze this phenomenon.

## 3.2 Enzymatic Networks and Modeling of Enzymatic Networks

Enzymes are proteins that catalyze chemical reactions, i.e., by increasing the rate of the reaction. Enzymes are not consumed by the reactions they catalyze, nor do they alter the equilibrium of these reactions. Their activity can be affected by other molecules-inhibitors that decrease enzyme activity; or activators that increase it. Activity is also affected by temperature, chemical environment, and the concentration of substrate (Garrett and Grisham, 1999). Enzyme kinetics (binding of enzymes to substrate, forming an enzyme-substrate product and catalyzing the reaction obtaining the reaction product) is described by Michaelis-Menten type of equations described in Chapter 1. Several enzymes can work together in a specific order, creating metabolic pathways. More than one enzyme can catalyze the same reaction in parallel, allowing more complex regulation network.

Robust adaptation, as an important concept in biological systems, was explored on simple 3-node enzyme networks. In the paper by Ma *et al.* (2009) and in the further chapters, the nodes are interacting: node $A$ receives the input signal, node $C$ transmits the output and node $B$ is intermediary node and can play different regulatory roles. There can be up to nine directed links among the three nodes ($A-A$, $A-B$, $A-C$, $B-A$, $B-B$, $B-C$, $C-A$, $C-B$, $C-C$), where each link has three possibilities: positive regulation, negative regulation, or no regulation. For every network, a value is assigned to each of the nine links: 1 for positive regulation, -1 for negative regulation and 0 for no regulation. Therefore a network (or topology) is described by a sequence of nine numbers corresponding to those nine links. Thus, there are $3^9 = 19{,}683$ possible topologies. Topologies consisting of only two nodes, A and C, cannot show adaptation, and this observation justifies the need of having at least three node networks.

Each node in the model network has a fixed total concentration that can be interconverted between two forms (active and inactive) by other active enzymes in the network or by basal enzymes. The links between the nodes are modeled using ordinary differential equations characterized by the Michaelis-Menten constants ($K_M$'s) and catalytic rate constants ($k_{cat}$'s) of the enzymes.

The input acts on node $A$ and the output is the active concentration of node $C$. There are 3,645 topologies that have no direct or indirect links from the input to the output. All the remaining 16,038 topologies (networks) were used in the study (Ma *et al.*, 2009). The set of ordinary differential equations that describe the dynamics of the network is as follows:

$$\begin{aligned}
\frac{dA(t)}{dt} &= k_{IA} I \frac{1-A}{1-A+K_{IA}} + \sum_i k_{X_iA} X_i \frac{1-A}{1-A+K_{X_iA}} - \sum_i k_{Y_iA} Y_i \frac{A}{A+K_{Y_iA}} \\
\frac{dB(t)}{dt} &= \sum_i k_{X_iB} X_i \frac{1-B}{1-B+K_{X_iB}} - \sum_i k_{Y_iB} Y_i \frac{B}{B+K_{Y_iB}} \\
\frac{dC(t)}{dt} &= \sum_i k_{X_iC} X_i \frac{1-C}{1-C+K_{X_iC}} - \sum_i k_{Y_iC} Y_i \frac{C}{C+K_{Y_iC}},
\end{aligned} \tag{3.1}$$

where $X_i = A, B, C, E_A, E_B$ or $E_C$ are activating enzymes, $Y_i = A, B, C, F_A, F_B$ or $F_C$ are deactivating enzymes, and $I$ is the input signal.

Here, $A$ represents the concentration of active state, and $(1-A)$ is the concentration of the inactive state, similarly for nodes $B$ and $C$. The enzymatic regulation converts its target node between the two forms. For example, a positive regulation of node $B$ by node $A$ ("activation of $B$ by $A$"), denoted by $A \rightarrow B$ would mean that the active $A$ converts $B$ from its inactive to its active form and would be modeled by the rate $R(B_{inactive} \rightarrow B_{active}) = k_{AB}\ A\ \frac{1-B}{1-B+K_{AB}}$, where $A$ is the concentration of the active form of node $A$ and $(1-B)$ the concentrations of the inactive form of node $B$. Similarly, $A \dashv B$ means that the active $A$ catalyzes $B$ from its active to its inactive form, with a rate $R(B_{active} \rightarrow B_{inactive}) = k_{AB}\ A\ \frac{B}{(B+K_{AB})}$. When there are multiple regulations of the same sign on a node, the effect is additive. For example, if node $C$ is positively regulated by node $A$ and node $B$, $R(Cinactive \rightarrow Cactive) = k_{AC}\ A\ \frac{1-C}{1-C+K_{AC}} + k_{BC}\ B\ \frac{1-C}{1-C+K_{BC}}$. It is assumed that interconversion between active and inactive forms of a node is reversible. Thus, if a node i has only positive incoming links, it is assumed that there is a background (constitutive) deactivating enzyme $F_i$ of a constant concentration (set to be 0.5) to catalyze the reverse reaction. Similarly, a background activating enzyme $E_i = 0.5$ is added for the nodes that have only negative

incoming links. In the equation for node $A$, an input term is added to the right hand-side of the equation: $I\ k_{IA}\ \frac{1-A}{1-A+KIA}$.

## 3.3 Parameter Sampling using the Latin Hypercube Sampling method

### 3.3.1 Motivation for using the LHS

Once network topologies are defined as in Section 3.2, the goal is to determine how many of them show adaptation and, for how many parameter sets adaptation holds. Therefore, for each network topology, 10,000 parameter sets were sampled uniformly in logarithmic scale, using the Latin hypercube sampling method (Iman et al., 1980). The sampling ranges of the parameters are $k_{cat}$=0.1-10 and $K_m$=0.001-100. Here "Circuit" is defined as a network or topology with a particular choice of parameters. Therefore, there are 10,000 circuits for each topology, making it a total of $16,038 \cdot 10,000 \approx 1.6 \cdot 10^8$ different circuits to examine.

The Latin Hypercube Sampling (LHS) Method was developed by W. J. Conover as a method of improving the efficiency of the Monte Carlo algorithm used to characterize the uncertainty in inputs to computer models (Wyss and Jorgensen, 1998; Iman, 2001). Software implementing the LHS strategy was developed in 1975 by R. L. Iman, and formally released in 1980 (Iman, Davenport and Zeigler). The LHS uses a stratified sampling scheme to improve the coverage of the input space (Iman, 2001). Consider a variable $Y = f(X_1, X_2, \ldots, X_k)$. Applying the Monte Carlo algorithm, sampling is done repeatedly from the assumed joint probability density function of the $X$'s and $Y$ is evaluated for each sample, together with its distribution, mean and higher moments. The LHS software developed by Iman et. al. supports this approach for generating samples of the $X$'s. The program output, for $n$ Monte Carlo repetitions, is a set of $n$ $k$-dimensional vectors of input variables. In other words, $n$ different values from each of $k$ variables of $X$ were sampled. Each input vector can then be evaluated by the function or program to generate $n$ values of the result $Y$ ($Y$ may be a scalar or a vector whose dimension is determined by the function or program of interest). This approach provides reasonable estimates for the distribution of $Y$ if the value of $n$ is quite

large. However, since a large value of $n$ requires a large number of computations from the function or a program of interest, which is potentially a very large computational expense, additional methods of uncertainty estimation were sought.

### 3.3.2 Latin Hypercube Sampling Theory

The Latin Hypercube Sampling selects n different values from each of $k$ variables ($X_1$, ..., $X_k$) in the following manner. The range of each variable is divided into $n$ non overlapping intervals on the basis of equal probability; one value from each interval is selected at random with respect to the probability density in the interval. The $n$ values thus obtained for $X_1$ are paired in a random manner (equally likely combinations) with the $n$ values of $X_2$. These $n$ pairs are combined in a random manner with the $n$ values of $X_3$ to form $n$ triplets, and so on, until $n$ $k$-tuplets are formed. This is the Latin hypercube sample. In general, a set of $n$ Latin hypercube sample points in $k$-dimensional Euclidean space contains one point in each of the intervals for each of the $k$ variables.

### 3.3.3 An example of the Latin Hypercube Sampling

To illustrate how the specific values of a variable are obtained in a Latin hypercube sample, the following simple example given in (Wyss and Jorgensen, 1998) was observed. Suppose it is desired to obtain a Latin hypercube sample of size $n = 5$ from a normal distribution with a mean of 5.0 and a variance of 2.618 as indicated in Figure 3.1[1]. The density characteristics of the normal distribution allow for the definition of the equal probability intervals. These intervals are shown in Figure 3.1 in terms of a density function. The next step is to randomly select an observation within each of the intervals. This selection is done relative to the probability density function distribution being sampled (in this case, the normal distribution). This is equivalent to uniformly sampling the vertical axis of the Cumulative Distribution Function, (CDF) and then "inverting" the CDF to obtain the actual distribution values shown in Figure 3.1. Therefore to get the specific values, $n = 5$ numbers, $U_m$, where $m = 1, 2, 3, 4, 5$, are randomly selected from the standard uniform distribution (uniformly distributed between 0 and 1). For

this particular example the values are: 0.08, 0.61, 0.525, 0.935, 0.62, respectively. These values will be used to select distribution values randomly from within each of the $n = 5$ intervals. To accomplish this, each of the random numbers $U_m$ is scaled to obtain a corresponding cumulative probability, $P_m$, so that each $P_m$ lies within the $m$-th interval. Thus, for $n = 5$, $P_m = (\frac{1}{5})U_m + \frac{m-1}{5}$. This ensures that exactly one probability, $P_m$, will fall within each of the five intervals (0, 0.2), (0.2, 0.4), (0.4, 0.6), (0.6, 0.8) and (0.8, 1). The values $P_m$ are used with the inverse normal distribution function to produce the specific values to be used in the final Latin hypercube sample. Exactly one observation is taken from each interval shown in Figure (3.1).

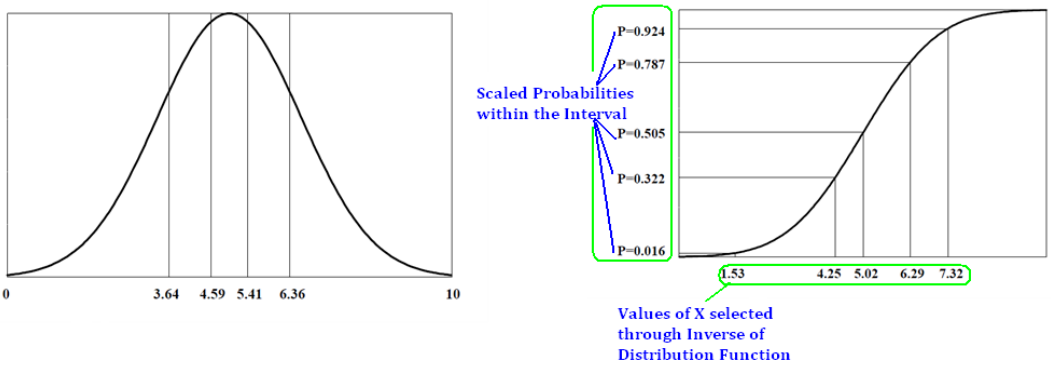



Figure 3.1: Input variable with a normal distribution sampled with the Latin Hypercube Sampling method

This procedure is repeated for each input variable, each time working with the corresponding cumulative distribution function. If a random sample is desired, then it is not necessary to divide the vertical axis into n intervals of equal width. Rather, $n$ random numbers between 0 and 1 are obtained and each is mapped through the inverse distribution function to obtain the specific values. The final step in the sampling process involves pairing the selected values.

### 3.3.4 The Latin Hypercube Sampling Algorithm

The LHS can be summarized as follows:

- Divide the cumulative distribution of each variable into $n$ equiprobable intervals;

[1]The figure was taken from Iman (2001) and adapted

- From each interval select a value randomly, for the $i$-th interval, the sampled cumulative probability can be written as: $Prob_i = (\frac{1}{n})r_u + \frac{i-1}{n}$ where $r_u$ is a uniformly distributed random number ranging from 0 to 1;

- Transform the probability values sampled into value $x$ using the inverse of the distribution function $F^{-1}$: $x = F^{-1}Prob$

- The $n$ values obtained for each variable $x$ are paired randomly (equally likely combinations) with the $n$ values of other variables. The method is based on the assumption that the variables are independent of each other.

## 3.4 Searching for circuits capable of adaptation: Sensitivity and Precision as two parameters in determining adaptation

In order to determine whether a particular topology exhibits adaptation, two functional quantities are observed: the circuit's sensitivity to input stimulus and its adaptation precision. These two concepts were introduced in the paper by Ma *et al.* (2009), and slightly modified in this work.

Sensitivity is defined as the height of output response relative to the initial steady-state value. This function takes into account the hight of the first peak, in the case where the output profile contains more than one peak. Precision represents the difference between the pre-and post-stimulus steady states.

A particular circuit architecture/parameter configuration is considered to be "functional" for adaptation if it shows a strong initial response (Sensitivity $> 1$) combined with strong adaptation (Precision $> 10$). In this part of the simulations, nonzero initial input was given ($I_1 = 0.5$) and then changed by 20% ($I_2 = 0.6$). For any particular topology, it was tested how many parameter sets can perform adaptation - a topology is considered to be more robust if a larger number of parameter sets yield the behavior defined above.

Two typical outputs from node C are response that contains overshoot followed by undershoot or vice-versa, and these cases are shown in Figure 3.2 and Figure 3.3. Sensitivity function takes only the first peak into account. For the notation introduced

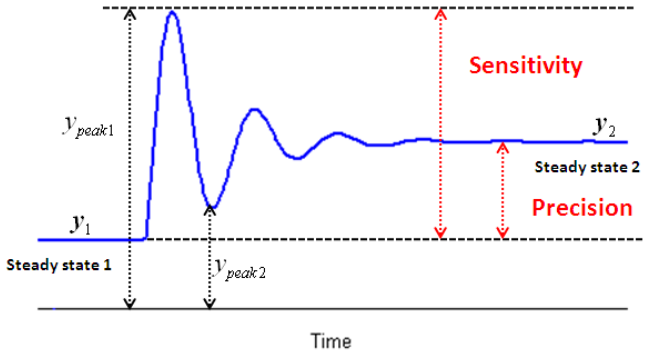



Figure 3.2: Output curve defining Sensitivity and adaptation: Example 1

in Figure 3.2, firstly the size of the peak was compared to the steady state value, $y_{peak1}$. If $y_{peak1} - y_1 < 10^{-5}$ it is assumed that there's no overshoot at all, so sensitivity function will take into the account the undershoot and the corresponding sensitivity equation [1] will be:

$$S = \left| \frac{\frac{y_{peak2} - y_1}{y_1}}{\frac{I_2 - I_1}{I_1}} \right|^2. \tag{3.2}$$

If $y_{peak1} - y_1 \geq 10^{-5}$ overshoot value is used to calculate the sensitivity function according to:

$$S = \left| \frac{\frac{y_{peak1} - y_1}{y_1}}{\frac{I_2 - I_1}{I_1}} \right|^2 \tag{3.3}$$

Similarly, for the notation introduced in Figure 3.3, if $|y_{peak1} - y_1| < 10^{-5}$, it is assumed that there's no undershoot at all, so sensitivity function will take into the account the overshoot and the corresponding sensitivity equation will be given by (3.2), or if $|y_{peak1} - y_1| \geq 10^{-5}$ sensitivity will be given by (3.3).

Precision function takes into account the relative change in the new and the old steady-state values:

$$\text{Precision}^{-1} = \left| \frac{\frac{y_2 - y_1}{y_1}}{\frac{I_2 - I_1}{I_1}} \right|^2. \tag{3.4}$$

.

[1] Paper by Ma *et al.* (2009) defines Sensitivity function without sqaring the expression

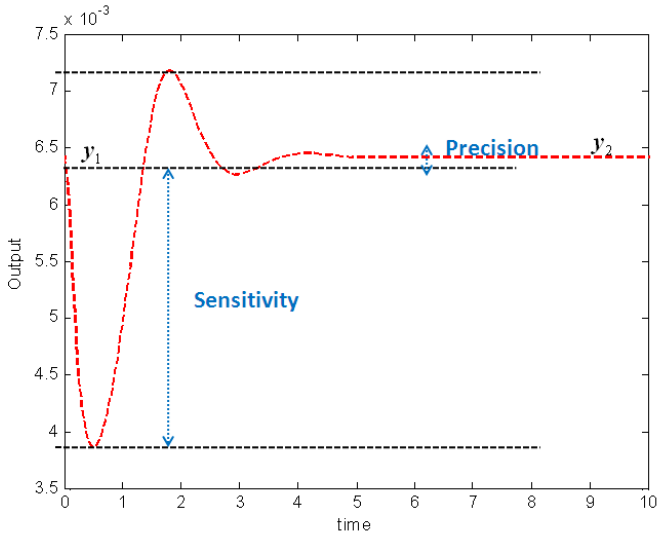



Figure 3.3: Output curve defining Sensitivity and adaptation: Example 2

Circuits with low sensitivity and high precision are classified as "No response circuits", those with high sensitivity and low precision as "Non-adaptive", and are therefore discarded from further analysis. Only circuits with both high precision and sensitivity show adaptation. Moreover, if for a given topology more then 10 parameter sets $(K_{M_i}, K_{cat_i})$ (out of possible 10,000 for each topology) yield adaptation, given topology shows robust adaptation. In this thesis, all adaptive circuits are taken as candidates for further analysis of Fold Change Detection, and robustness of parameters is not taken into account, since the robustness of parameters is not focus of this research.

In the paper by Ma *et al.* (2009) it was shown that out of $1.6 \cdot 10^8$ possible circuits (16,038 topologies each sampled with 10,000 parameters) only 395 topologies are robust (satisfy prescribed sensitivity and precision requirements and have more than ten, out of 10,000 functional parameter sets). In this work, firstly, we redid and verified all the computations, and 16,304 circuits showed adaptation. If the condition on robustness is imposed, we found that the total number of circuits that show adaptation within prescribed specs of sensitivity and precision, is 10210, moreover, we found out that 393 different topologies are adaptive, each having more than 10 functional parameter sets. This result is very similar to what was found in Ma *et al.* (2009) where 395 topologies are obtained.

One of the main results in the paper by Ma *et al.* (2009) is the classification of all 3-node circuits that are capable of adaptation into two topological classes: the Negative FeedBack Loop motif with Buffering node, (NFBLB) and the Incoherent Feed-Forward Loop motif with a Proportioner node, (IFFLP) in both of which control node B plays a defined role. Therefore, two major classes of minimal adaptive networks emerge from the analysis.

Negative feedback loop is defined as a topology whose links, starting from any node in the loop, lead back to that node with the cumulative sign of regulatory links within the loop being negative. The sign of of loop is determined by multiplying the signs of individual regulations along the path. Incoherent feed-forward loop is defined according to the definition given in Chapter 2. Negative feedback loop topologies differ widely in their ability to perform adaptation. There is only one class of simple negative feedback loops that can robustly achieve adaptation,namely NFBLB (Ma *et al.*, 2009). In this class of circuits, the output node must not directly feedback to the input node. Rather, the feedback must go through an intermediate node ($B$), which serves as a buffer. Some examples of Negative Feedback loops and Negative feedback-loops with the buffer node $B$ are given in Figure[1] 3.4.

Among feed-forward loops, incoherent feed-forward loops also differ in their performance. Of these, only the circuit topology in which the output node $C$ is subject to direct inputs of opposing signs (one positive and one negative) appears to be highly preferred. This topology structure will be referred to as Incoherent Feed-forward Loop with Proportioner Node, where node $B$ serves as a proportioner for node $A$, i.e., node $B$ is activated in proportion to the activation of node $A$ and exerts opposing regulation on node $C$ Ma *et al.* (2009). Examples of Incoherent Feed-Forward Loops and Incoherent Feed-Forward Loops with Proportioner node are given in Figure[1] 3.5. The following two sections briefly summarize results from (Ma *et al.*, 2009).

[1]Generated using the publicly available applet: http://tang.ucsf.edu/applets/Adaptation/Adaptation.html

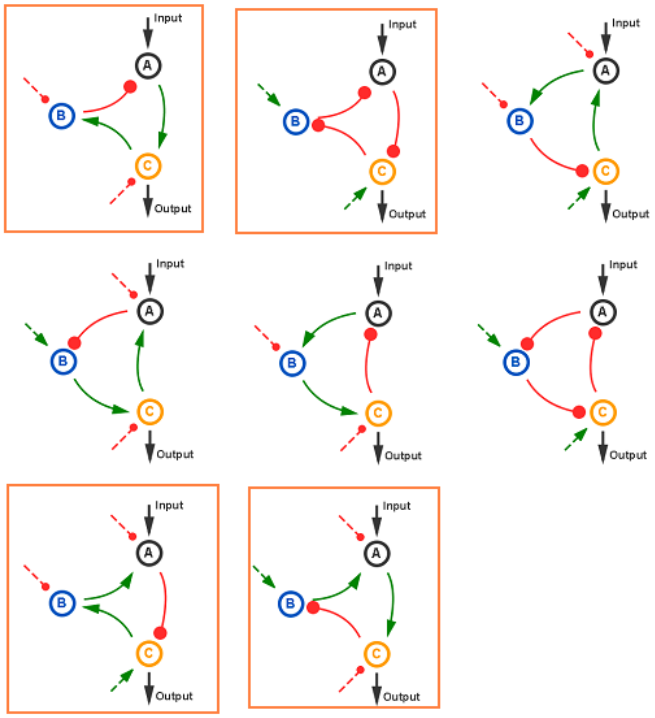



Figure 3.4: Negative Feedback Loops and Negative Feed-Back Loops with Buffer node (in squares)

### 3.4.1 Negative Feedback Loop with a Buffer Node

A simple example of Negative feedback loop is given by:

$$\begin{aligned}
\frac{dA(t)}{dt} &= k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{F_A A} F_A \frac{A}{A+K_{F_A A}} \\
\frac{dB(t)}{dt} &= k_{CB} C \frac{1-B}{1-B+K_{CB}} - k_{F_B B} F_B \frac{B}{B+K_{F_B B}} \\
\frac{dC(t)}{dt} &= k_{AC} A \frac{1-C}{1-C+K_{AC}} - k_{BC} B \frac{C}{C+K_{BC}}.
\end{aligned} \tag{3.5}$$

The NFBLB class of topologies has multiple realizations in three-node networks, all featuring regulation node $B$ that functions as a buffer, as shown in Figure [1]3.6. The first example from the Figure is described by (3.5), has a negative feedback loop between the regulation node $B$ and the output transmitting node $C$. The mechanism by which this NFBLB topology adapts and achieves high sensitivity can be determined by the analysis of the kinetic equations where $F_A$ and $F_B$ represent the concentrations of basal enzymes that carry out the reverse reactions on nodes $A$ and $B$, respectively (they oppose the active network links that activate $A$ and $B$). Analysis of the parameter sets

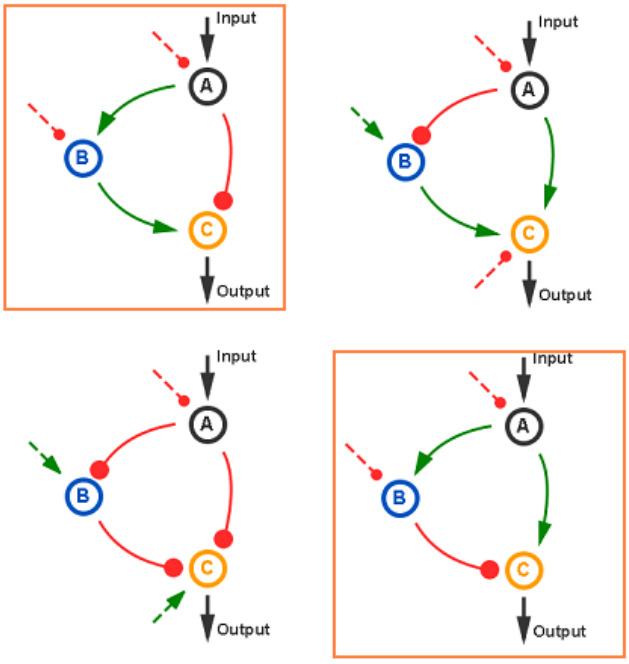



Figure 3.5: Incoherent Feed-Forward Loops and Incoherent Feed-Forward Loops with Proportioner node (in squares)

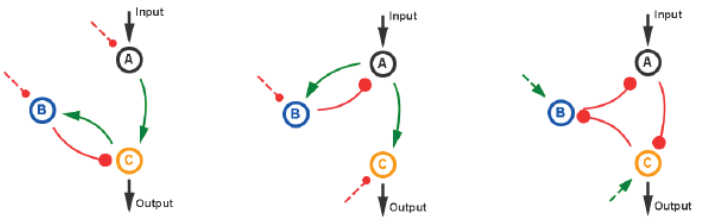
that enabled this topology to adapt, indicates that the two constants $K_{CB}$ and $K_{F_BB}$ (Michaelis-Menten constants for activation of $B$ by $C$ and inhibition of $B$ by the basal enzyme) tend to be small, suggesting that the two enzymes acting on node $B$ must approach saturation to achieve adaptation. Indeed, it can be shown that in the case of saturation this topology can achieve perfect adaptation. Under saturation conditions, i.e., $(1 - B) \gg K_{CB}$ and $B \gg K_{F_BB}$, the rate equation for $B$ can be approximated by the following:

$$\frac{dB(t)}{dt} = k_{CB}C - k_{F_BB}F_B. \tag{3.6}$$

Steady State solution is given by:

$$C^* = \frac{F_B k_{CB}}{k_{F_BB}}, \tag{3.7}$$

which is independent of the input level $I$. The output $C$ of the circuit can still transiently respond to changes in the input but eventually settles to the same steady state




| B→C | C→B | A→C |
|---|---|---|
| + | - | + or - |
| - | - | + |

| A→B | B→A | A→C |
|---|---|---|
| + | - | + or - |

| A→C | C→B | B→A |
|---|---|---|
| + | + | - |
| + | - | + |
| - | + | + |
| - | - | - |

Figure 3.6: Minimal Adaptation Networks: Negative Feedback Loops

determined by Equation (3.7). Equation (3.6) can be rewritten as:

$$\begin{aligned} \frac{dB(t)}{dt} &= k_{CB}(C - C^*) \\ B &= B^*(I_0) + k_{CB} + \int_0^t (C - C^*)\,\mathrm{d}\tau \end{aligned} \tag{3.8}$$

Thus, the buffer node $B$ integrates the difference between the output activity $C$ and its input-independent steady-state value. Therefore, this NFBLB motif, $C \dashv B \rightarrow C$, implements integral control which is a common mechanism for adaptation. All minimal NFBLB topologies use the same integral control mechanism for perfect adaptation.

### 3.4.2 Incoherent Feed-forward Loop with a Proportioner Node

$$\begin{aligned} \frac{dA(t)}{dt} &= k_{IA} I \frac{1 - A}{1 - A + K_{IA}} - k_{F_A A} F_A \frac{A}{A + K_{F_A A}} \\ \frac{dB(t)}{dt} &= k_{AB} A \frac{1 - B}{1 - B + K_{AB}} - k_{F_B B} F_B \frac{B}{B + K_{F_B B}} \\ \frac{dC(t)}{dt} &= k_{AC} A \frac{1 - C}{1 - C + K_{AC}} - k_{BC} B \frac{C}{C + K_{BC}} \end{aligned} \tag{3.9}$$

The other minimal topological class sufficient for adaptation is the incoherent feed-forward loop with a proportional node (IFFLP). The output node $C$ is subject to two regulations, both originating from the input but with opposing cumulative signs in the two pathways, as shown in Figure 3.5. The feed-forward circuit "anticipates" the output from a direct reading of the input. Node $B$ monitors the input and exerts an opposing force on node $C$ to cancel the output's dependence on the input. Therefore, node $B$ is a "proportioner". The adaptation mechanism is mathematically captured in

the equation for node $C$: if the steady-state concentration of the negative regulator $B$ is proportional to that of the positive regulator $A$, the equation determining the steady-state value of $C$, $dC/dt = 0$, would be independent of $A$ and hence of the input $I$. In this case, the equation for node $B$ generates the condition under which the steady-state value $B^*$ would be proportional to $A^*$: the first term in $dB/dt$ equation should depend on $A$ only and the second term on $B$ only. The condition can be satisfied if the first term is in the saturated region ($(1 - B) \gg K_{AB}$) and the second in the linear region ($B \ll K_{F_B B}$) so that:

$$B^* = A^* \cdot k_{AB} \cdot K_{F_B B}/(F_B \cdot k_{F_B B}). \tag{3.10}$$

This relationship shows that the steady-state concentration of active $B$ is proportional to the steady-state concentration of active $A$. Thus $B$ will negatively regulate $C$ in proportion to the degree of the pathway input.

## 3.5 Significance of the results obtained by Ma et.al. and formulation of Fold Change Detection Problem

The theory of network motifs can help to identify a particular function in a variety of different complex networks by recognizing a particular set of simple motifs whose mathematical behavior is known to give rise to that particular function. Work by Ma *et al.* (2009) gave the guidelines for understanding the dynamics of complex networks in terms of its motifs, and motivated our analysis of another mathematically interesting property, Fold Change Detection, that assumes perfect adaptation as explained in Chapter 2. The FCD property found its justification in experimental work related to bacterial chemotaxis in E. coli (Shimizu *et al.*, 2010) where the experiments showed that receptor activity which determines the rate of tumbles that guide the bacteria up chemoattractant gradients will be insensitive to scaling in attractant source. The contribution of this thesis is extending the work of Ma *et al.* (2009) to verifying how many circuits defined in their work will exhibit Fold Change Detection property, and moreover analytically finding the underlying mechanism of such a behavior in 3-node enzymatic networks.

## 3.6 Numerical Procedure for Testing Fold Change Detection

In order to check Fold Change Detection property for all 3-node enzymatic networks given by Ma *et al.* (2009), we formed 2 data-bases. The first one, "0.3-0.36", contains simulation results for the input change from $I = 0.3$ to $I = 0.36$, and the second one, "0.5-0.6", contains simulation results in the case of input change $I = 0.5$ to $I = 0.6$, as shown in Figure 3.7. The latter is checked, and it is shown that the number of robust topologies (more than 10 functional parameters) is 393, which is comparable to 395 Ma *et al.* (2009) had in the paper. In this thesis, however, we are interested in all adaptive topologies, without imposing the robustness condition. Both data-bases contain matrices that describe the dynamics of the nodes $A$, $B$ and $C$ for corresponding input change for all topologies and same parameter sets used in the paper by Ma *et al.* (2009), obtained by the Latin Hypercube Sampling Method. Moreover, information about steady states of all the nodes and the size of maxima and minima in the dynamics for node $C$, is recorded for all the tested circuits. Sensitivity and Precision functions are defined in the same manner as before, and also included in the data-base.

Data mining proceeded, and the adaptation property of circuits in "0.3-0.36" data-base is compared to the adaptive circuits in "0.5 to 0.6".

The total number of adaptive circuits in "0.3-0.36" data base is 15820. New circuits that emerged in this data-base, and those that showed adaptation in "0.5-0.6" data-base, but failed to adapt in the "0.3-0.36", are removed from both data-bases. The remaining number of circuits in both data-bases in now 15818. This leads to the conclusion, that the range of the input must be determined first. It is done by checking the adaptation to constant inputs, and determining the approximate range to be 0.2-0.6. Other choices for range of the input signal led to the increasing number of circuits that are not able to show adaptation even for that constant input. So far, we have only checked adaptation within data-base "0.3-0.36". In other words, whether the steady states for $I = 0.36$ is the same, or within the prescribed precision tolerance, as in the case for $I = 0.3$. More circuits had to be removed from the data-base because they

didn't satisfy sensitivity specs, therefore being classified as "non-responsive". Comparing the data-bases, 15425 circuits in "0.5-0.6" passed sensitivity test, and 10,436 in "0.3-0.36". Leaving only those that satisfy sensitivity specs in both databases, 10,290 circuits remain as candidates for further testing.

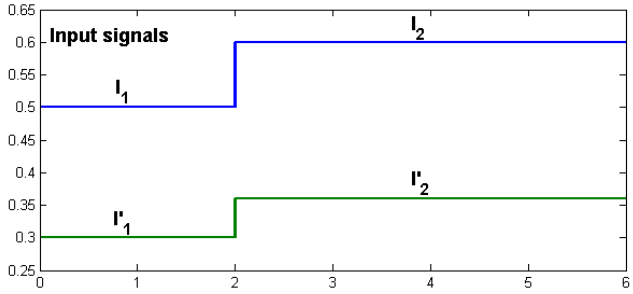



Figure 3.7: Input Fold

Adaptation values for data in the two data-bases that now have all the same circuits are compared to make sure the circuits adapt to identical steady state values. If change in input from 0.3 to 0.36 led to exact adaptation or adaptation within prescribed sensitivity/precision specs but is different from the value obtained for input change from 0.5 to 0.6 in terms of relative difference that is greater than 0.1 those circuits are again removed from both databases and not considered for further analysis. The number of circuits that passed the test is 10,063. At this point, it can be observed that none of the two node circuits (having an input receiving node, A, and output transmitting node, C, without regulatory node B) can achieve adaptation within the prescribed specs for Sensitivity and Precision, and this conclusion is consistent with the conclusion given in the paper by Ma *et al.* (2009).

Lastly, equal peak response analysis is done. Fold Change Detection imposes conditions on same exact adaptation values in both data-bases but also requires the transient behaviors as well as maximal responses (Weber's law) to be identical. By checking peak responses large portion of circuits is eliminated from further analysis. The following hypothesis is tested:

$$\begin{aligned} \frac{minpeak(y_{0.6}) - minpeak(y_{0.36})}{max(minpeak(y_{0.6}), minpeak(y_{0.36}))} &< 0.1 \\ \frac{maxpeak(y_{0.6}) - maxpeak(y_{0.36})}{max(maxpeak(y_{0.6}), maxpeak(y_{0.36}))} &< 0.1 \end{aligned} \tag{3.11}$$

which specifies that peak responses for both minima and maxima in the two cases must be the same or within specified tolerance range.

### 3.6.1 Numerical Results

We found that 216 circuits passed the "equal peak response" test and are candidates for Fold Change Detection. After plotting these circuits, two types of behavior emerge: Fold Change Detection in the case of 25 circuits (21 different topologies); and Shifted Fold Change Circuits, where the shift is with respect to time axis ("Time Shifting FCD"), in the case of 44 circuits (43 different topologies). The remaining circuits show a "weaker" form of FCD, due to the condition of satisfying Precision specs with the relative tolerance less than 0.1. These circuits show almost exact adaptation (Sontag, 2008) in both databases, therefore showing "Magnitude Shifting FCD", and are excluded from further analysis. Example of such circuit is given in Figure 3.8, and shifting with respect to magnitude axis the FCD property is obtained, as shown in Figure 3.9.

However, since the objective of this thesis is determining mechanisms that give rise to FCD, Magnitude Shifted FCD circuits are eliminated from further analysis.

Figures 3.10, 3.11, 3.12 and 3.13 show examples of Time Shifting FCD and Exact FCD, respectively.

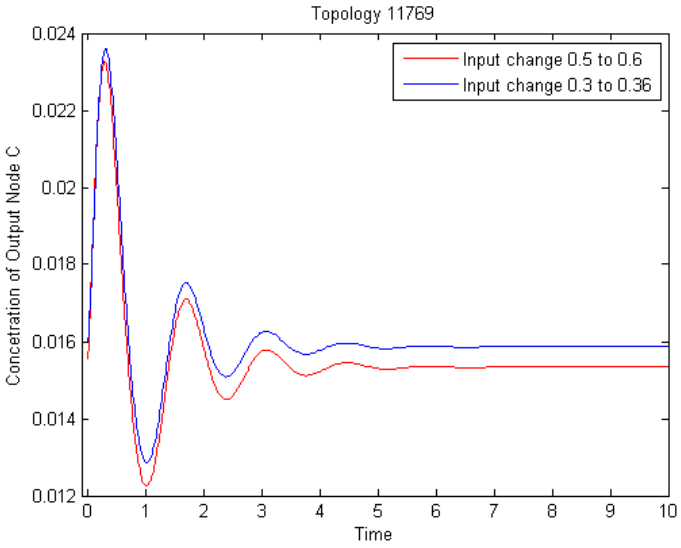



Figure 3.8: Example of Magnitude Shifting FCD.

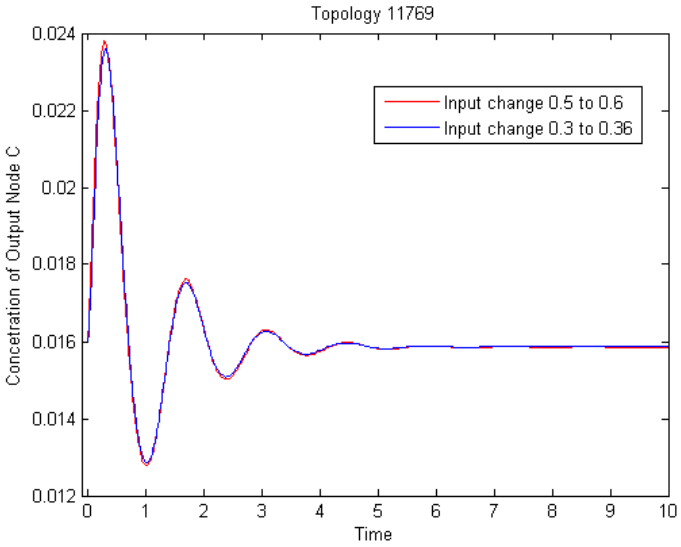



Figure 3.9: Example of Magnitude Shifting FCD. Circuit would exibit exact FCD by shifting one of the outputs by the amount of the difference between the steady states. In this case the shift equals to 0.00052

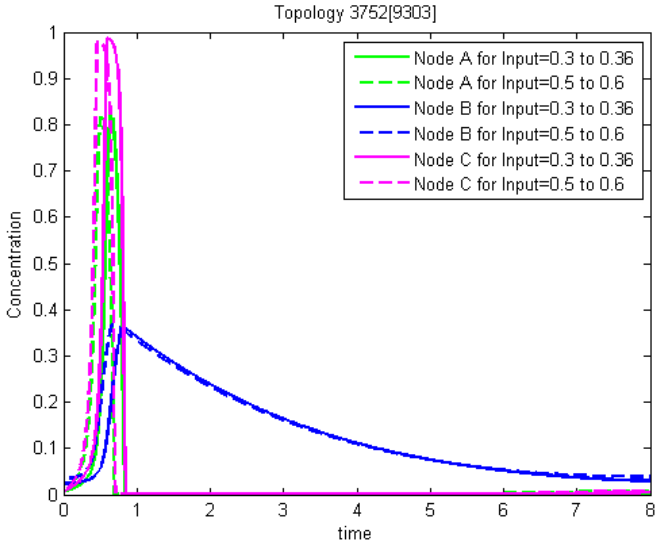



Figure 3.10: Example of Time Shifting FCD Property

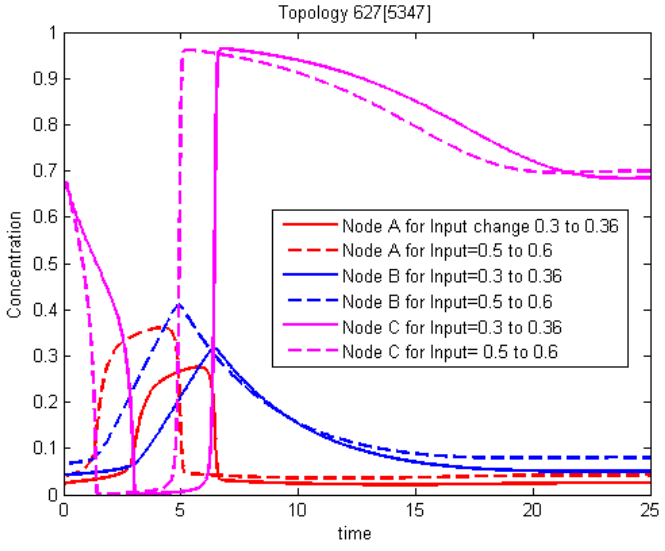



Figure 3.11: Example of Time Shifting FCD Property

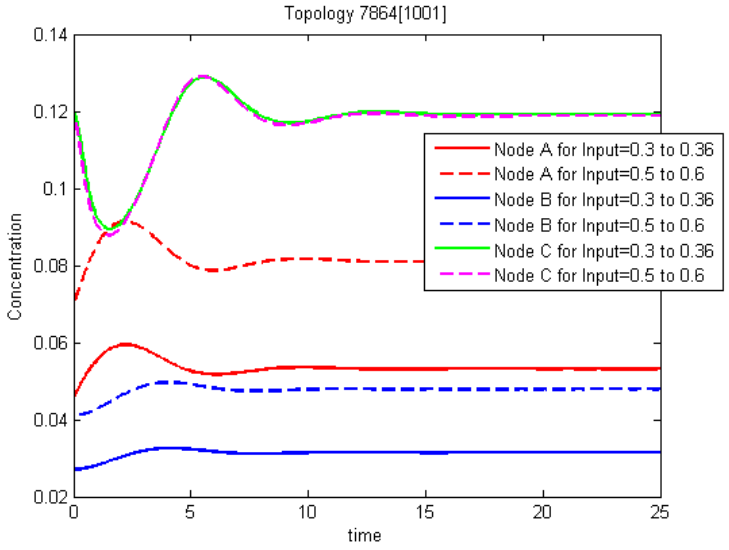



Figure 3.12: Example of FCD Property

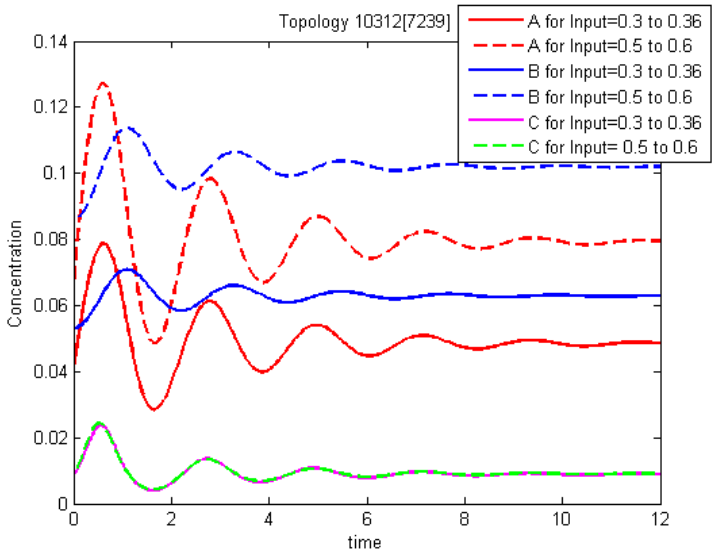



Figure 3.13: Example of FCD Property

## 3.7 Conclusion

In this chapter, we gave an overview of the results by Ma *et al.* (2009),we redid all the computations and corrected some minor errors. Then, work by Ma *et al.* (2009) is extended to the examination of the Fold Change Detection property for adaptive circuits. Our results indicate that among 10,063 adaptive circuits only 25 (21 different topologies) are able to achieve Fold Change Detection property. Additionally, the phenomenon of Time Shifting FCD was detected among 44 circuits (43 different topologies). The mechanism of exact FCD will be further explored in Chapter 4.

# Chapter 4

# Results

## 4.1 Analytical Results

As a starting point, let us look at the simplest of the 25 circuits that we had found. The set of differential equations describing the circuit is given by:

$$\begin{aligned} f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} \\ f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} - k_{F_B B} F_B \frac{B}{B+K_{F_B B}} \\ f_C = \quad & \frac{dC(t)}{dt} = k_{AC} A \frac{1-C}{1-C+K_{AC}} - k_{BC} B \frac{C}{C+K_{BC}} \end{aligned} \tag{4.1}$$

Let us observe the equation that describes dynamics of node $C$. We will use this as an equation that describes the dynamics of the output and denote $C(t) = y(t)$ for consistency with the notation used to explain Fold Change Detection in Chapter 2. Suppose that:

$$\frac{dC}{dt} = A f_1(C) - B f_2(C) \tag{4.2}$$

is "fast" in the sense that if $A$ and $B$ were constant, the dynamics would equilibrate fast. The functions $f_1$ and $f_2$, in the equation, are nonlinear functions of $C$. Setting $dC(t)/d(t) = 0$, and solving for $C$ in terms of $A$ and $B$, we have that $C(t)$ can be thought of as a function $C = P(A, B)$ of the current values of $A$ and $B$, i.e. $C(t) = P(A(t), B(t))$. One finds that there is unique positive solution for $C(t)$. It clearly follows from (4.2) that:

$$P(mA, mB) = P(A, B) \; \forall \, m. \tag{4.3}$$

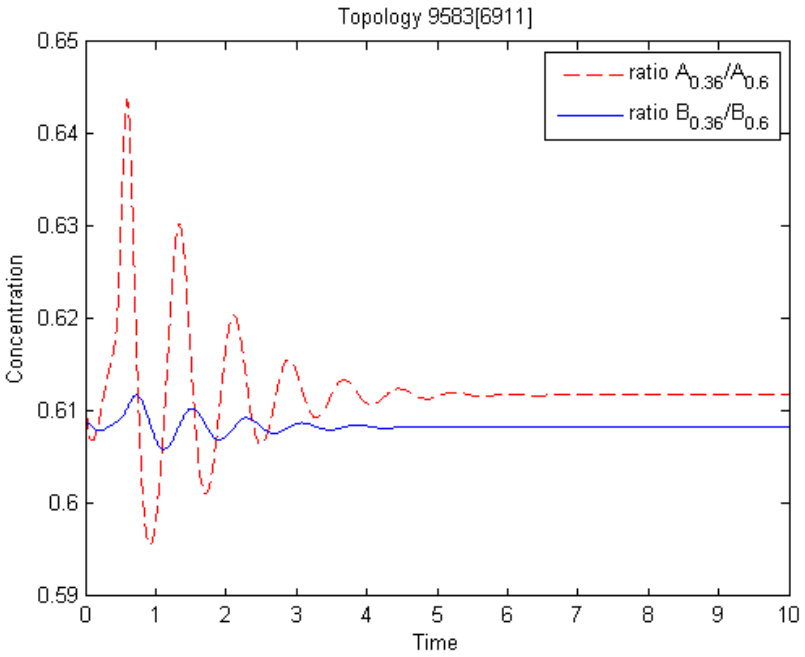



Figure 4.1: Homogeneity Property for A and B

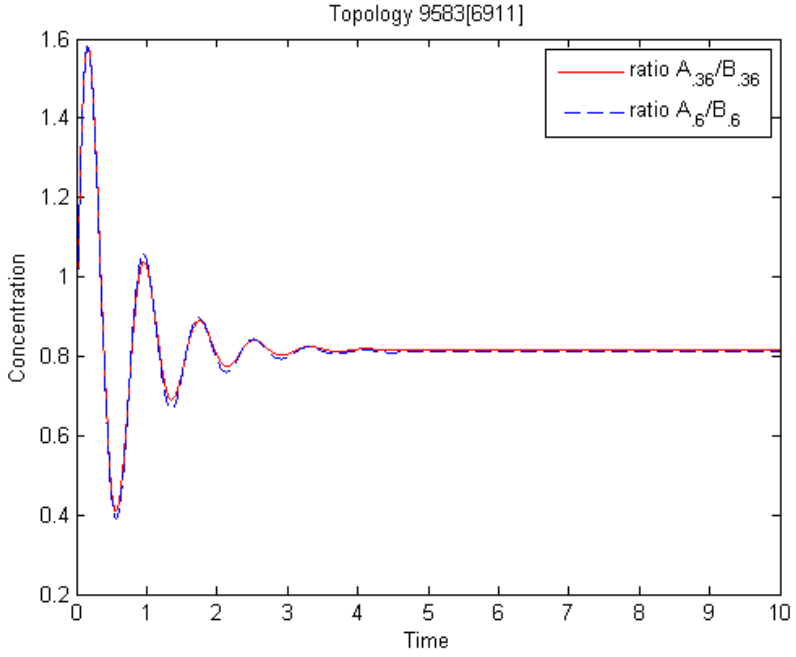



Figure 4.2: A(t)/B(t)

Our key discovery was that when scaling the input, for each of the circuits, there is a factor $m(t)$ such that $A(t)$ and $B(t)$ scale as $m(t)A(t)$ and $m(t)B(t)$, respectively. So, if $A(t)$ is replaced by $m(t)A(t)$ and $B(t)$ replaced by $m(t)B(t)$, $C(t)$ will be the same in both cases, which gives the Fold Change Detection Property. The scaling factor $m(t)$, can be be approximated by a constant that equals $\frac{3}{5}$. Therefore, the ratio of concentrations of nodes $A$ and $B$ doesn't depend on the value of the input signal, and quickly converges to a constant value, as shown in Figure 4.2. The following idea was tested for all the circuits that show FCD property, and the simulation results are given in the Appendix. As it can be seen from Figure 4.3 that depicts derivative of the output, $C$ equilibrates fast. Also, the contributions of the terms that constitute the equation for the dynamics of $C$ node are shown in Figure 4.4.

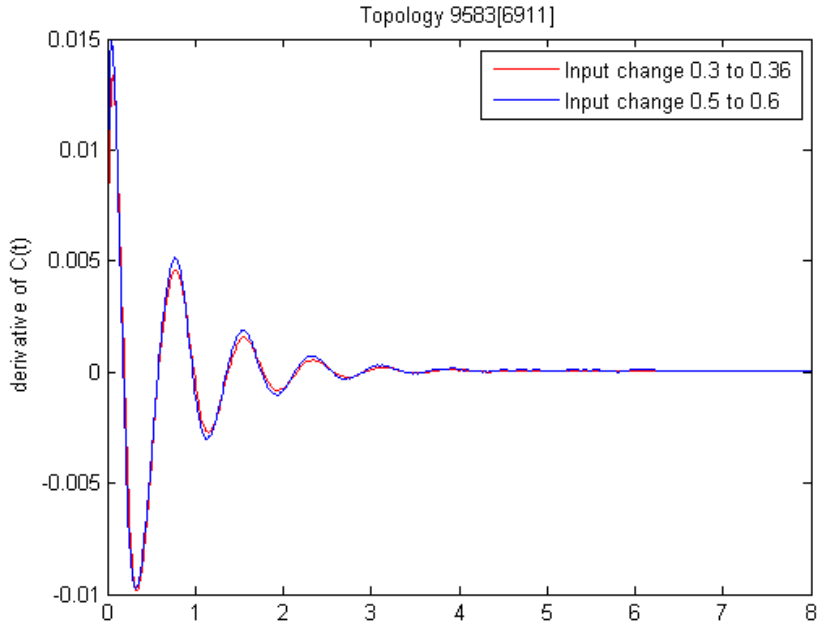



Figure 4.3: Derivative of the output: C node equilibrates fast

We find that when we replace $C(t)$ by $P(A(t), B(t))$, the plot looks the same as for the true nonlinear system (see Appendix). For the particular example discussed in this chapter the solution $C(t)$ is obtained solving the following quadratic equation:

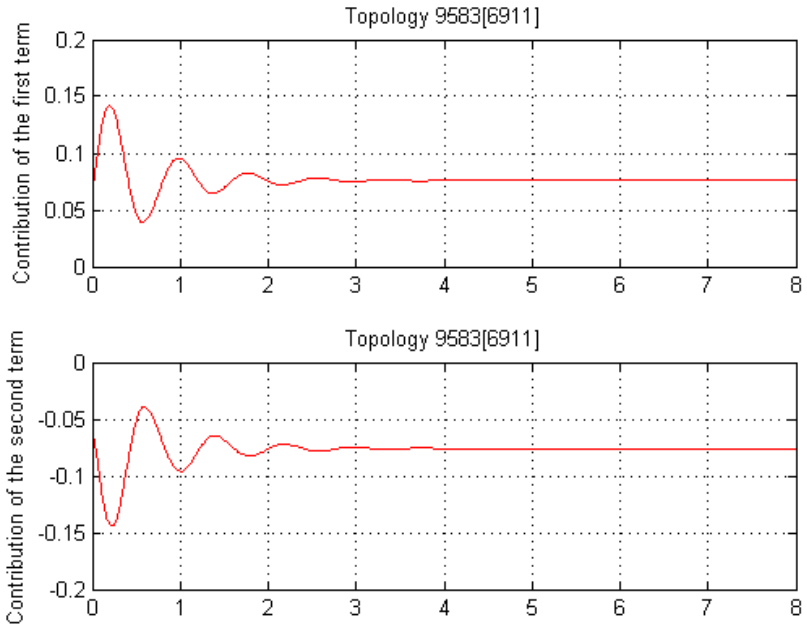



Figure 4.4: Contributions of the two terms in the differential equation for C node

$$C(t) = P(A(t), B(t))$$

$$\text{Setting: } \frac{dC(t)}{dt} = 0$$

$$\frac{A(t)}{B(t)} \frac{1 - C(t)}{1 - C(t) + K_{AC}} = \frac{k_{BC}}{k_{AC}} \frac{C(t)}{C(t) + K_{BC}}$$

$$C^2(t) \left( \frac{A(t)}{B(t)} - \frac{k_{BC}}{k_{AC}} \right) + C(t) \left( \frac{k_{BC}}{k_{AC}}(1 + K_{AC}) - \frac{A(t)}{B(t)}(1 - K_{BC}) \right) - \frac{A(t)}{B(t)} K_{BC} = 0$$

and taking the positive solution. Let us rewrite the differential equation for the system given by (4.1) in the standard singular perturbation form:

$$\begin{aligned} \dot{x} &= f(x, y, u) \\ \epsilon \dot{y} &= g(x, y, u) \end{aligned} \tag{4.4}$$

where $x(t) = (A(t), B(t))$, $y(t) = C(t)$, $u(t) = I$ and $\epsilon \ll 1$

The function $g$ doesn't need to be linear, only the homogeneity property has to apply, as stated above. The output of node $C$ equilibrates fast, and thus we can assume that $g(x(t), y(t), u) = 0$.

Let us review the necessary and sufficient conditions for FCD, summarized as follows:

$$\begin{aligned} \dot{x} &= f(x, y, u) \\ \dot{y} &= g(x, y, u) \end{aligned} \tag{4.5}$$

FCD holds if system (4.5) is stable, shows exact adaptation and $g$ and $f$ satisfy the following homogeneity properties (Shoval *et al.*, 2010b):

$$\begin{aligned} f(px, y, pu) &= pf(x, y, u) \\ g(px, y, pu) &= g(x, y, u) \end{aligned} \tag{4.6}$$

If $f$ is linear this is also a necessary condition for FCD (Shoval *et al.*, 2010b). Notice that in our case (4.6) doesn't hold, because $g(px, y, pu) = pg(x, y, u)$.

### 4.1.1 An approximation

We will see later that our system can be approximated by:

$$\begin{aligned} \dot{x} &= A(y)x + B(y)u \\ \epsilon\dot{y} &= h(x, y) \end{aligned} \tag{4.7}$$

where $x(t) = (A(t), B(t))$, $y(t) = C(t)$.

**Assumption 4.1.1.** *Suppose:* $\dot{x} = A(y)x + B(y)u$ *and* $\epsilon\dot{y} = h(x, y)$ *with* $0 < \epsilon \ll 1$ *and* $h(px, y) = ph(x, y)$.
*There exists* $y \approx \alpha(x)$ *such that* $\alpha(x)$ *is a unique solution of* $h(x, y) = 0$.

**Lemma 4.1.1.** $\alpha(px) = \alpha(x), \ (\forall p, x)$

*Proof.*

Pick x,y:

$h(x, y) = 0$

$h(x, \alpha(x)) = 0$

$ph(x, \alpha(x)) = 0$

$h(px, \alpha(x)) = 0$

$\alpha(px) = \alpha(x)$ by uniqueness of solution.

□

**Theorem 4.1.1.** *Consider a system of the following form:*

$$\begin{aligned} \dot{x} &= A(y)x + B(y)u \\ y &= P(x) \end{aligned} \tag{4.8}$$

*where* $P(px) = P(x), \ \forall \ p$. *Then FCD holds.*

*Proof.*

$$\begin{aligned} \dot{x} &= A(y)x + B(y)u \\ z &= px \\ \dot{z} &= p\dot{x} = A(y)z + B(y)pu \\ y &= P(x) = P(z) \end{aligned}$$

□

Since for small $\epsilon$ this system approximates the true system (4.1), the fact that FCD holds for (4.8), combined with the fact that this system appoximates our system, allows us to conclude that an approximate FCD property holds for the true system. Our main contribution is in noticing this reduction and checking its validity for all the circuits.

### 4.1.2 Linearization

We next show that for every one of the FCD circuits, true system behaves almost like a linear system, and can be approximated by systems of the form: (4.7).

Steady state values of nodes $A$ and $B$ were checked for different constant inputs, ranging from 0.3 to 0.6 with an increment of 0.01. Simulation plots show an approximately linear relationship for both nodes $A$ and $B$. The results justified the idea of linearization of the nonlinear system. The system (4.1) can be seen as first order nonlinear dynamic system in form:

$$\frac{d}{dt}z(t) = \mathcal{F}(z(t), u(t)) \tag{4.9}$$

Assume that $z(t_0)$ is known. Where $z_n(t) = [A_n, B_n, C_n]^T = [A_{0.3}, B_{0.3}, C_{0.3}]^T$ and $z(t) \in \mathbf{R}^n$, $u(t) \in \mathbf{R}^r$ and $\mathcal{F}$ n-dimensional vector function. Nominal system trajectory

$z_n(t)$ is known, and nominal system input that keeps the system on the nominal trajectory is given by $u_n(t)$. Assume that the actual system dynamics in the immediate proximity of the system nominal trajectories can be approximated by the first terms of the Taylor series. That is, starting with:

$$\begin{aligned} z(t) &= z_n(t) + \Delta z(t) \\ u(t) &= u_n(t) + \Delta u(t) \end{aligned} \tag{4.10}$$

and

$$\frac{d}{dt} z_n(t) = \mathcal{F}(z_n(t), u_n(t)) \tag{4.11}$$

we expand $z(t)$ from equation (4.10) into Taylor series as follows:

$$\begin{aligned} \frac{d}{dt} z_n(t) + \frac{d}{dt} \Delta z &= \mathcal{F}(z_n + \Delta z, u_n + \Delta u) \\ \frac{d}{dt} z_n(t) + \frac{d}{dt} \Delta z &= \mathcal{F}(z_n, u_n) + (\frac{\partial \mathcal{F}}{\partial z})|_{z_n(t), f_n(t)} \Delta z + (\frac{\partial \mathcal{F}}{\partial u})|_{z_n(t), f_n(t)} \Delta u \end{aligned} \tag{4.12}$$

Higher order terms are neglected, by choosing $\Delta z$ and $\Delta u$ to be small quantities.

Expanding it around the steady state $(A^*, B^*, C^*) = (A_{0.3}, B_{0.3}, C_{0.3})$ and $u = I = 0.3$ the linearized model is given by:

$$\begin{bmatrix} \frac{d\Delta A}{dt} \\ \frac{d\Delta B}{dt} \\ \frac{d\Delta C}{dt} \end{bmatrix} = \begin{bmatrix} \frac{\partial f_A}{\partial A} & \frac{\partial f_A}{\partial B} & \frac{\partial f_A}{\partial C} \\ \frac{\partial f_B}{\partial A} & \frac{\partial f_B}{\partial B} & \frac{\partial f_B}{\partial C} \\ \frac{\partial f_C}{\partial A} & \frac{\partial f_C}{\partial B} & \frac{\partial f_C}{\partial C} \end{bmatrix} \cdot \begin{bmatrix} \Delta A \\ \Delta B \\ \Delta C \end{bmatrix} + \begin{bmatrix} \frac{\partial f_A}{\partial I} \\ 0 \\ 0 \end{bmatrix} \Delta I. \tag{4.13}$$

Using the following parameters: $K_{AB} = 0.001191$; $k_{AB} = 1.466561$; $K_{AC} = 0.113697$; $k_{AC} = 1.211993$; $K_{BA} = 0.001688$; $k_{BA} = 44.802268$; $K_{BC} = 0.009891$; $k_{BC} = 7.239357$; $K_{IA} = 0.093918$; $k_{IA} = 11.447219$; $k_{AC} = 1.211993$; $K_{AC} = 0.1136927$; $K_{F_B} = 9.424319$; $k_{F_B} = 22.745736$; $I = 0.3$, we obtained the following matrices:

$$\begin{bmatrix} \frac{d\Delta A}{dt} \\ \frac{d\Delta B}{dt} \\ \frac{d\Delta C}{dt} \end{bmatrix} = \begin{bmatrix} -1.7913 & -43.5485 & 0 \\ 1.4647 & -1.1887 & 0 \\ 1.0881 & -0.8892 & -40.3872 \end{bmatrix} \begin{bmatrix} \Delta A \\ \Delta B \\ \Delta C \end{bmatrix} + \begin{bmatrix} 10.4088 \\ 0 \\ 0 \end{bmatrix} \Delta I \tag{4.14}$$

The system will be denoted by the usual state space form:

$$\frac{d}{dt} \Delta z(t) = (\frac{\partial \mathcal{F}}{\partial z})|_{z_n(t), f_n(t)} \Delta z(t) + (\frac{\partial \mathcal{F}}{\partial u})|_{z_n(t), f_n(t)} \Delta u(t) \tag{4.15}$$

Introducing matrices $\bar{A}$ and $\bar{B}$ for corresponding partial derivatives, we have:

$$\Delta \dot{z} = \bar{A}\Delta z + \bar{B}\Delta u \tag{4.16}$$

For the system denoted by (4.1) behavior of the outputs from node $A$ and $B$ was observed and compared the corresponding outputs of the linearized model. The results are shown in Figure 4.5. Based on the plot, it can be inferred that indeed, linearization of nonlinear system around its steady state values, and $u = I = 0.3$ captures the behavior of the system well. Therefore the behavior of the system described by (4.1) can be decomposed into linearized model describing dynamics of nodes $A$ and $B$, and singularly perturbed approximation for the dynamics of node $C$.

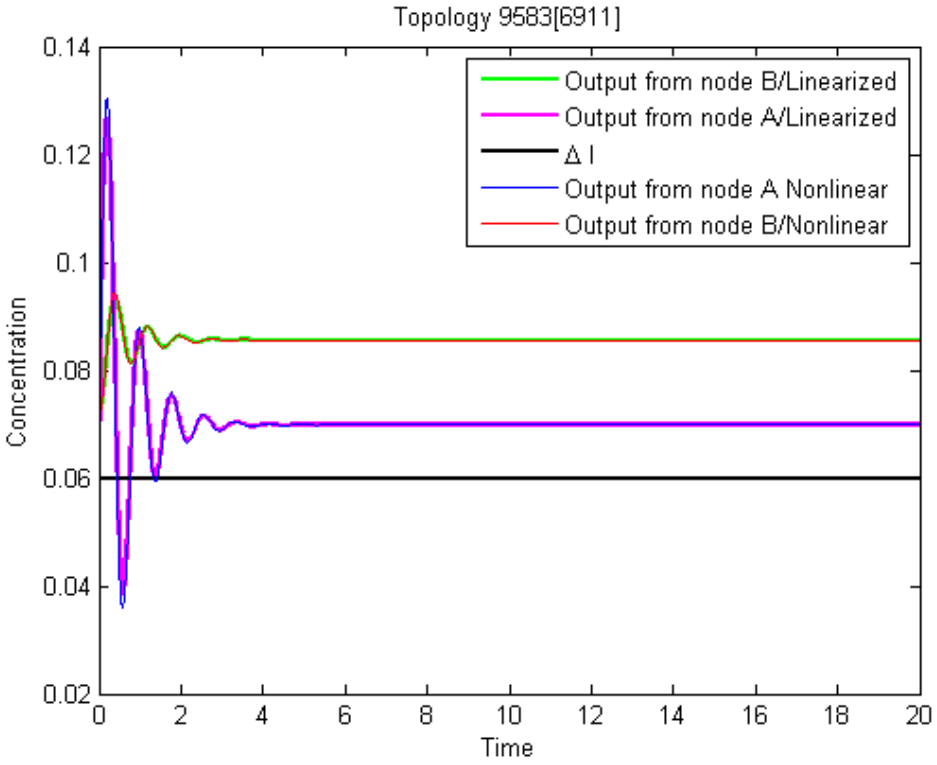



Figure 4.5: Linearized model

When the approach explained above was applied to remaining 24 circuits that show FCD, the following conclusions are obtained:

- Only $A \rightarrow C$ and $B \rightarrow C$ regulations affect the dynamics of output node C. The contribution of positive or negative self-loop on C can be neglected, therefore the dynamics for C node can again be approximated by solving $\frac{dC}{dt} = 0$ for C(t) in terms of $\frac{A(t)}{B(t)}$, neglecting term that reflects self-loop on C. An example is shown in Figure 4.7.

- All circuits that show FCD possess an incoherent feed-forward loop.

- Out of 10,000 parameters from parameter space obtained by the Latin Hypercube sampling method, certain parameter sets have shown to be more favorable then the others. Therefore Figures[1] 4.8, 4.9, 4.10, and 4.11 denote groups of circuits that all show FCD for same parameter sets, 6911, 305, 7239, 3497, respectively. The circuit on the top of every figure is the minimal motif for given parameter set, and the circuits below it, contain the minimal motif and also additional regulatory loops. Additionally, circuits in rectangles don't contain any link where node $C$ feeds back to $A$ or $B$. The circuits on Figure[1] 4.12 share unique parameter sets. Topologies 3022 and 3751 show FCD for two different parameter sets (7239 and 6911), and topology 10312 for three different parameter sets (6911,5228 and 7239) which makes it the most robust[2].

- All circuits satisfy the sensitivity requirement, but differ drastically in the value of sensitivity function, shown in Figure 4.6.

[1]Figures are generated using the publicly available applet:http://tang.ucsf.edu/applets/Adaptation/Adaptation.html

[2]Note that simulation results showed that self-loop on C doesn't affect much the dynamics on the ouput, and its contribution is therefore neglected in all the circuits

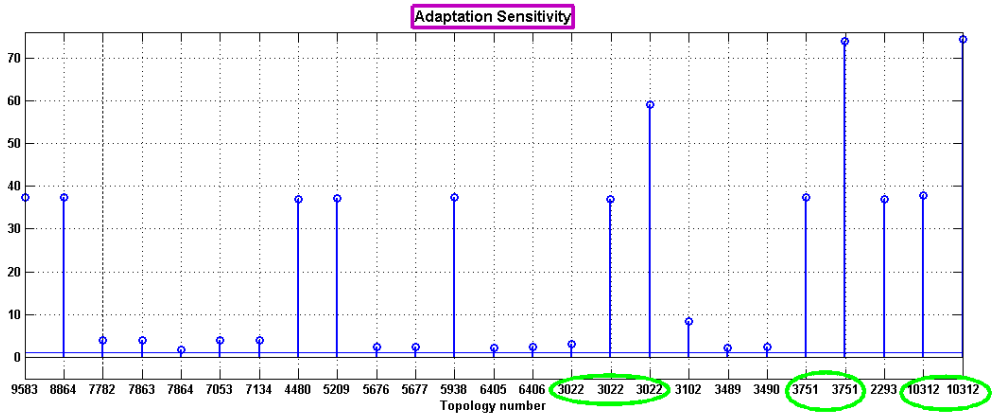



Figure 4.6: Sensitivity function for topologies that show FCD, topologies with multiple parameter sets are circled

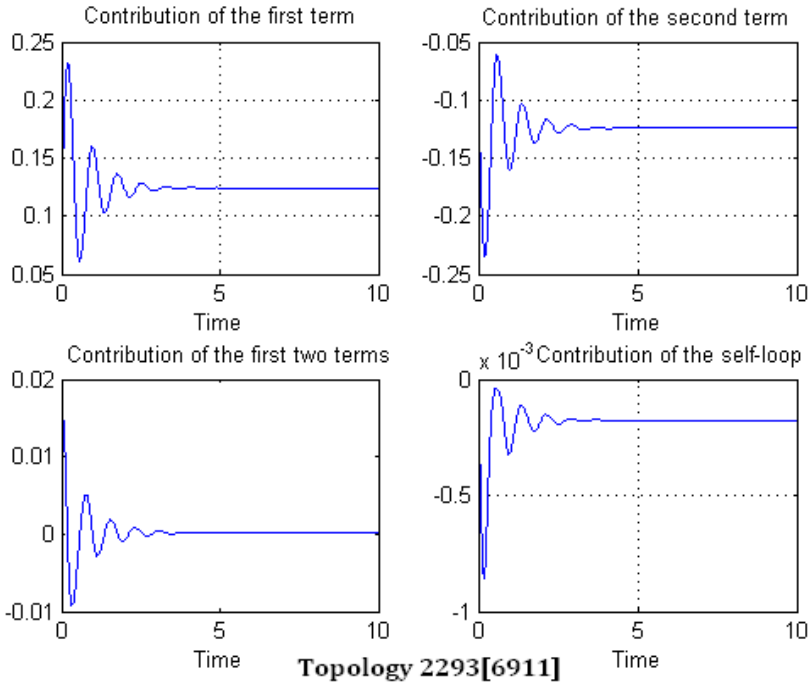



Figure 4.7: Contribution of terms in the equation for node C. Self loop doesn't affect much the dynamics

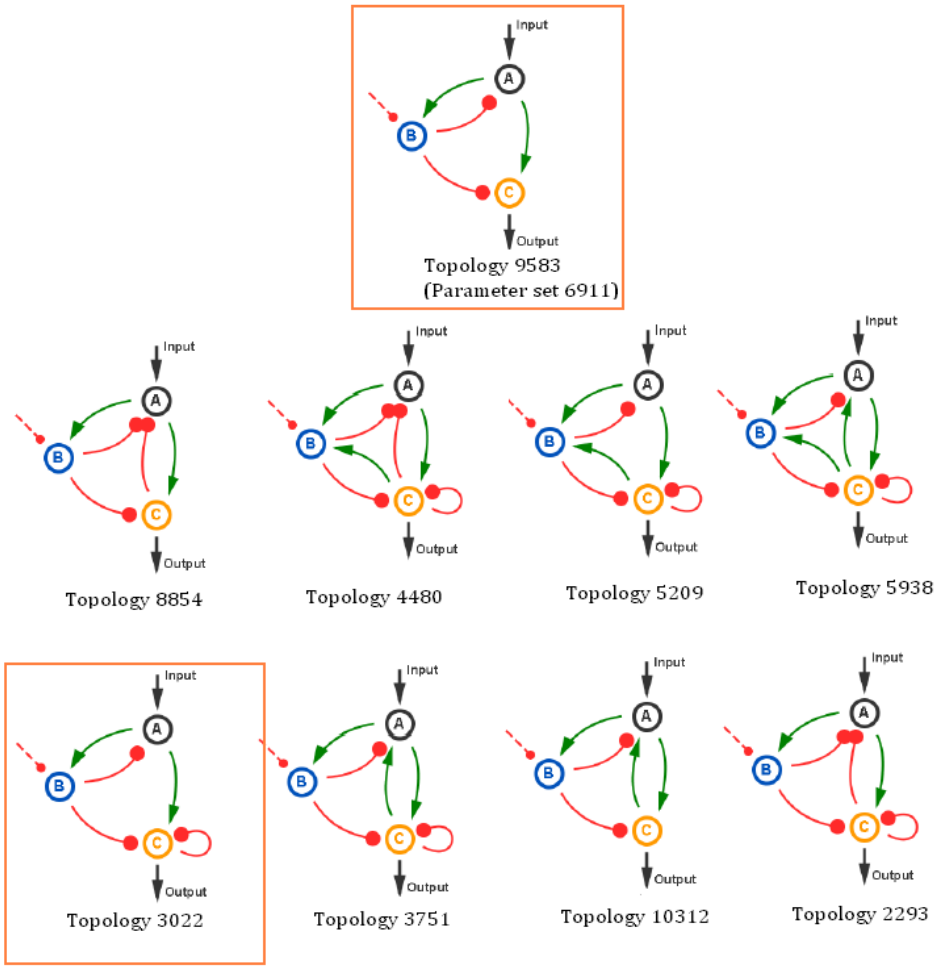



Figure 4.8: Circuits that show FCD

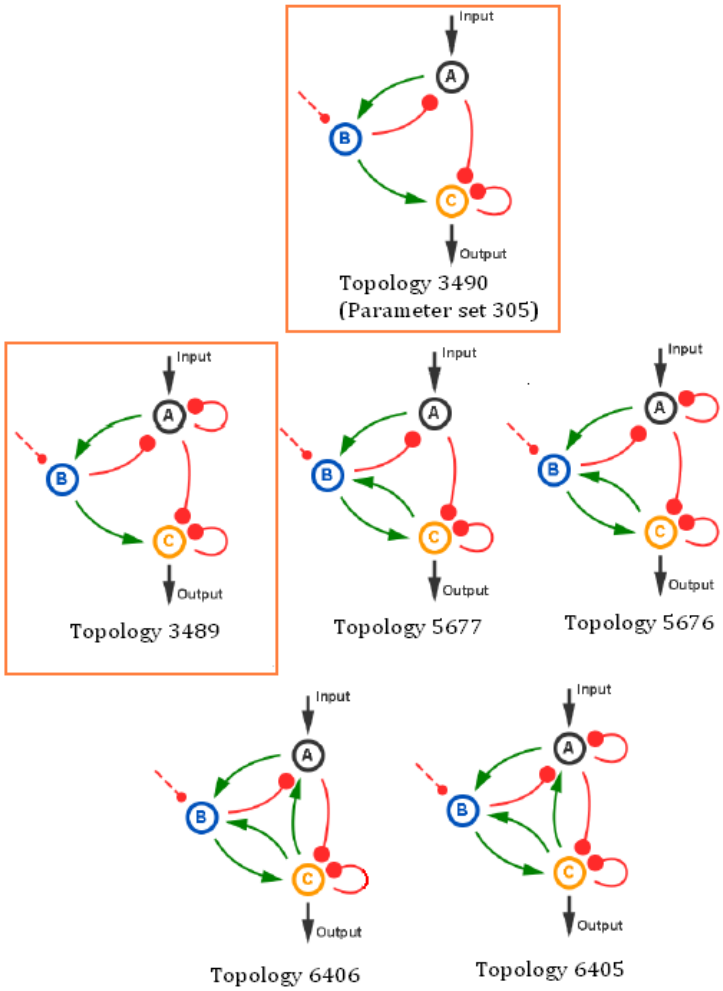



Figure 4.9: Circuits that show FCD

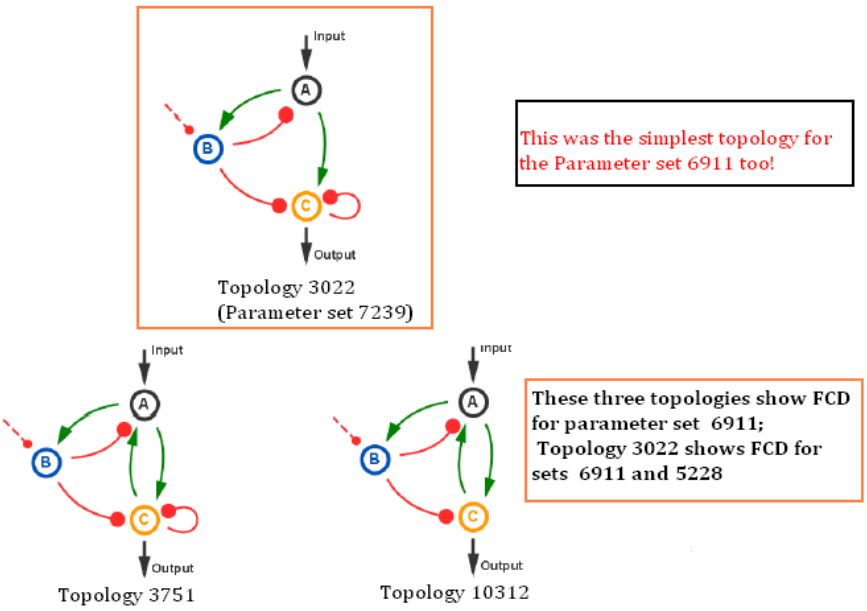



Figure 4.10: Circuits that show FCD

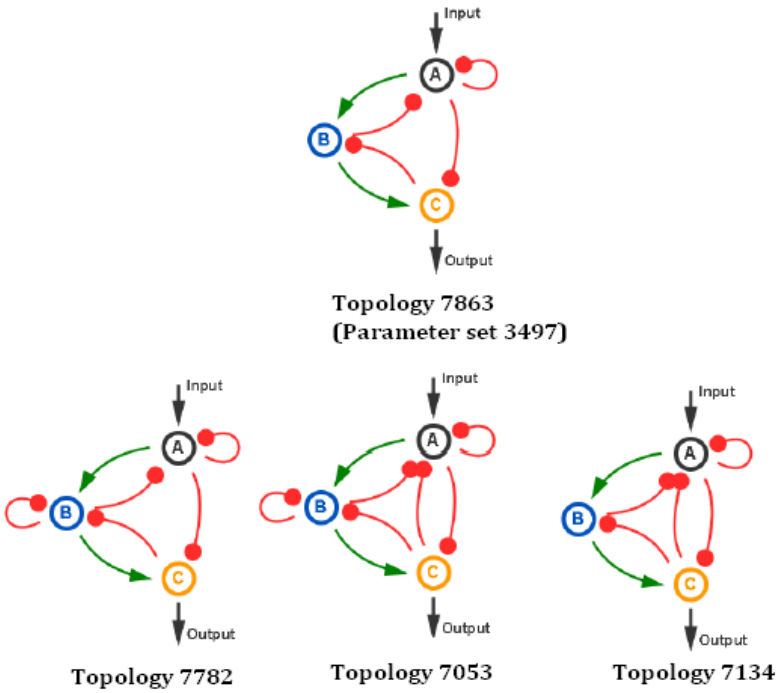



Figure 4.11: Circuits that show FCD

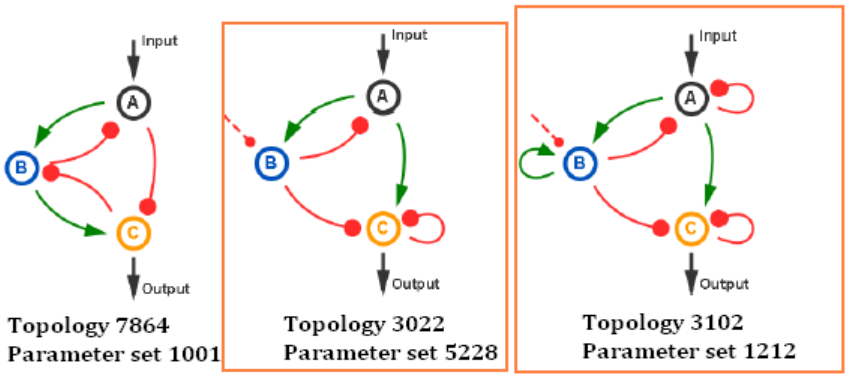



Figure 4.12: Circuits that show FCD

## 4.2 Conclusion

In this Chapter, the linearization of the system that describes the dynamics of nodes A and B together with a singular perturbation approximation of the dynamics of node C was shown to describe well the dynamics of all 3-node enzymatic networks that exhibit FCD property. The theory was developed on the simplest circuit that exhibits FCD, and tested on all other circuits that show FCD. It was also shown that additional self-loops on output node C don't affect the dynamics of the output. Additionally, all FCD circuits contain an incoherent feed-forward loop in their core, suggesting that it might be a necessary motif for Fold Change Detection in 3-node enzymatic networks.

# Chapter 5

# Conclusions and Future Work

## 5.1 Conclusions

The study of network motifs developed by Uri Alon, was used as the main tool to understand the dynamics of all 3-node enzymatic networks capable of achieving Fold Change Detection property. An exhaustive search through 16,038 topologies with 10,000 parameter sets for each topology, led to the conclusion that despite the diversity of enzymatic circuits only a small number of them is capable of achieving FCD property and the mechanism for achieving it can be found in an analytical form. Linearized system around the steady state values of the variables of the system, for a relatively large range of the input signal, in combination with a quadratic approximation for the dynamics of the output, gives an excellent approximation of the mathematical property, Fold Change Detection, recently experimentally discovered in many biological systems.

## 5.2 Future Work

We will explore the extension of the results of this thesis to the examples of Time Shifting Fold Change Detection circuits, and Magnitude Shifting FCD circuits. Further analysis of the former phenomenon might give explanation of the common property all these circuit share, and determine analytical explanation behind the delay. Therefore, we would be provided with a mechanism, or perhaps a minimal motif, all circuit must possess to produce a delay. Moreover, the work can be extended to pulse-like inputs, and testing the FCD property for other type of biological networks, i.e. transcription based networks.

# Chapter 6

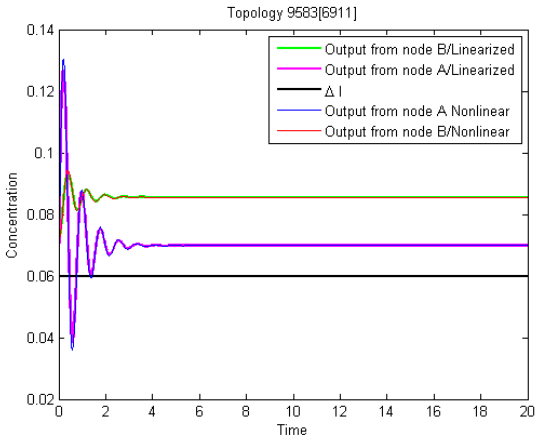
# Appendix

## 6.1 Circuits that exhibit Fold Change Detection

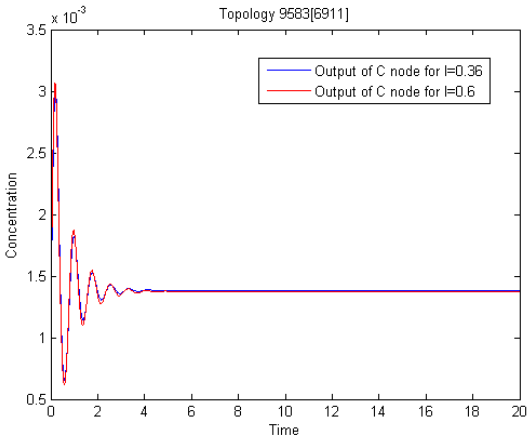
In this section the following results are given: set of differential equations describing nonlinear model for circuit that shows Fold Change Detection property; dynamics of nodes A and B using the Nonlinear model (numerical solution) and using the Linearized Model around operating point; $(A^*, B^*, C^*) = (A_{0.3}, B_{0.3}, C_{0.3})$ and $u = 0.3$; dynamics of the output node C, using nonlinear model (numerical solution) and using quadratic approximation discussed in Chapter 4, and comparison to actual, numerical solution of corresponding differential equation; homogeneity property for nodes A and B; and dynamics of the ratio between concentrations of A and B nodes in time. All the plots contain information of the topology number and parameter set used in simulations. Circuit 1.

$$\begin{aligned} f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} \\ f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} - k_{F_B B} F_B \frac{B}{B+K_{F_B B}} \\ f_C = \quad & \frac{dC(t)}{dt} = k_{AC} A \frac{1-C}{1-C+K_{AC}} - k_{BC} B \frac{C}{C+K_{BC}} \end{aligned} \tag{6.1}$$

Where: $K_{AB} = 0.001191$; $k_{AB} = 1.466561$; $K_{AC} = 0.113697$; $k_{AC} = 1.211993$; $K_{BA} = 0.001688$; $k_{BA} = 44.802268$; $K_{BC} = 0.009891$; $k_{BC} = 7.239357$; $K_{IA} = 0.093918$; $k_{IA} = 11.447219$; $k_{AC} = 1.211993$; $K_{AC} = 0.1136927$; $K_{F_B} = 9.424319$; $k_{F_B} = 22.745736$; $I = 0.3$;

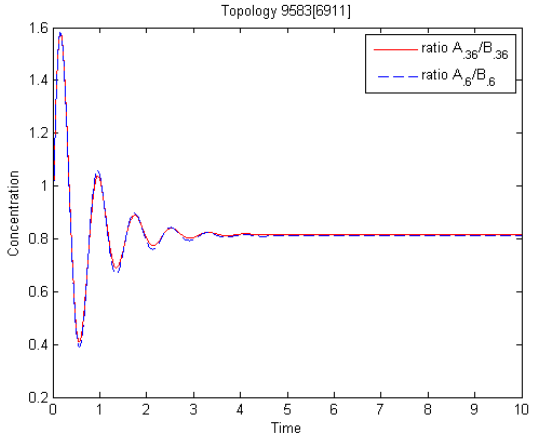



(a) Dynamics of A and B in Linearized model

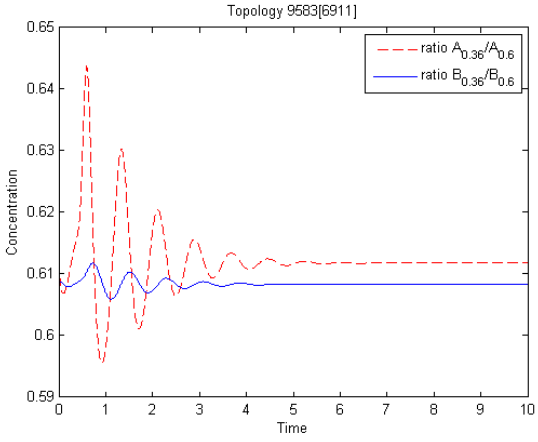



(b) Ouput from C Nonlinear Model

Figure 6.1: Topology 9583

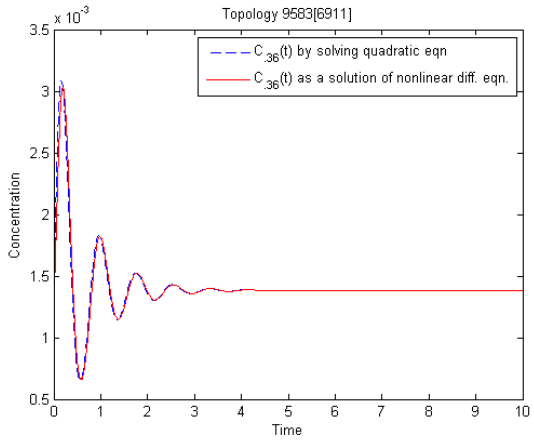



(a) A/B ratio

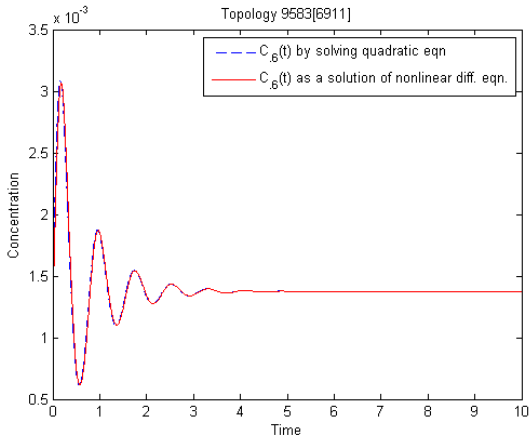



(b) AA and BB ratios

Figure 6.2: Topology 9583




(a) Quadratic approximation and Output of Nonlinear System



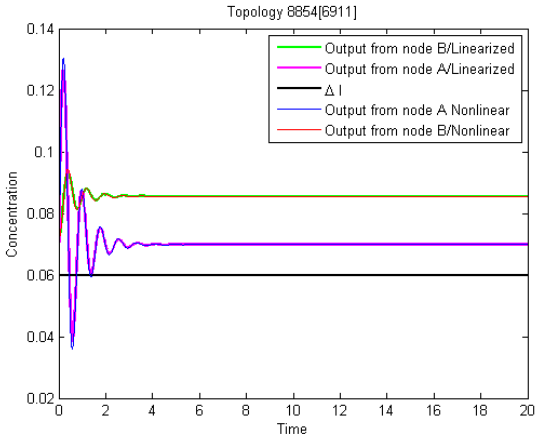

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.3: Topology 9583

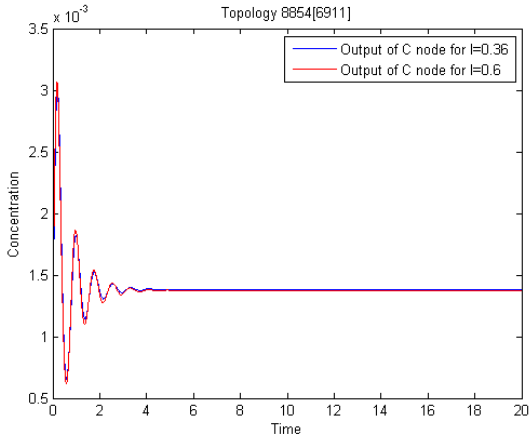
Circuit 2.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} - k_{CA} C \frac{A}{A+K_{CA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} - k_{F_B B} F_B \frac{B}{B+K_{F_B B}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{AC} A \frac{1-C}{1-C+K_{AC}} - k_{BC} B \frac{C}{C+K_{BC}}
\end{aligned}
\tag{6.2}
$$

Where: $K_{IA} = 0.093918$; $k_{IA} = 11.447219$; $K_{BA} = 0.001688$; $k_{BA} = 44.802268$; $K_{CA} = 90.209027$; $k_{CA} = 96.671843$; $K_{AB} = 0.001191$; $k_{AB} = 1.466561$; $K_{F_B} = 9.424319$; $k_{F_B} = 22.745736$; $K_{AC} = 0.113697$; $k_{AC} = 1.211993$; $K_{BC} = 0.009891$; $k_{BC} = 7.239357$;

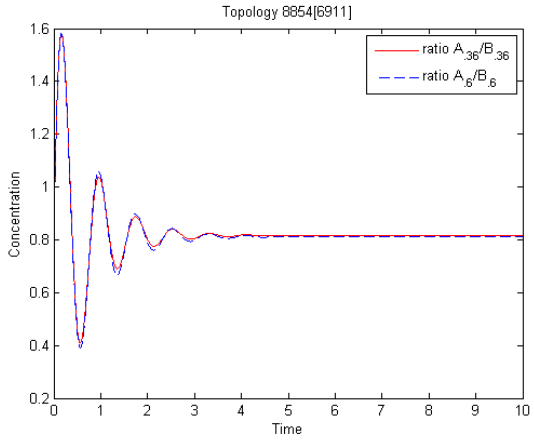



(a) Dynamics of A and B in Linearized model

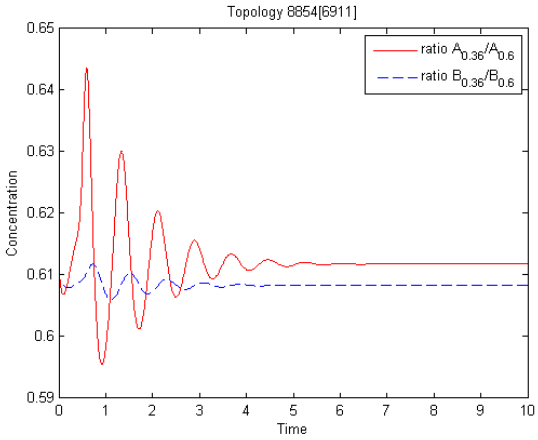



(b) Ouput from C Nonlinear Model

Figure 6.4: Topology 8854

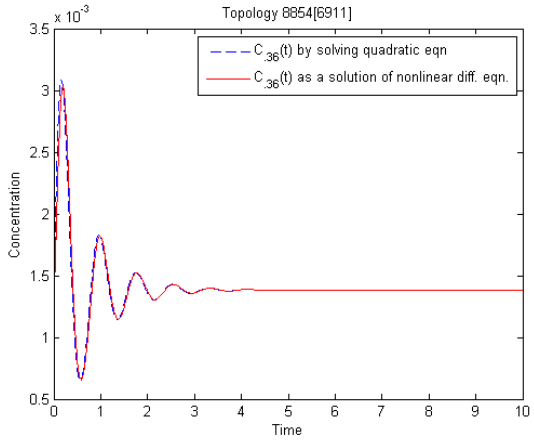



(a) A/B ratio

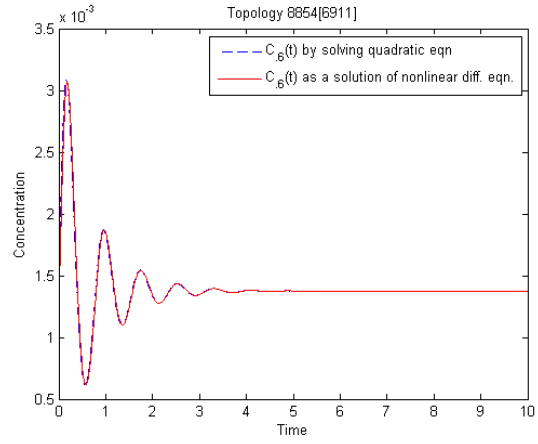



(b) AA and BB ratios

Figure 6.5: Topology 8854




(a) Quadratic approximation and Output of Nonlinear System



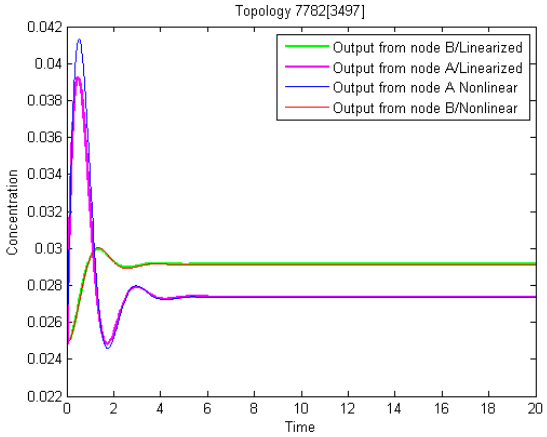

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.6: Topology 8854

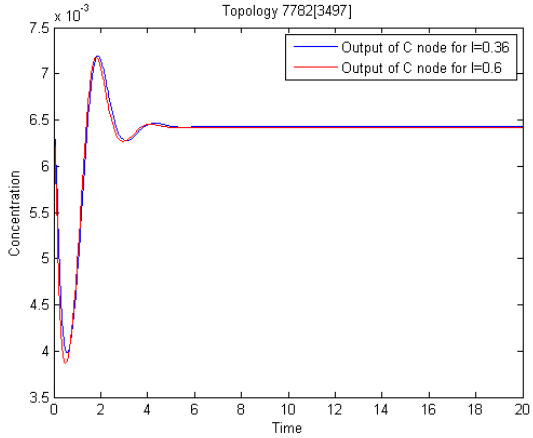
Circuit 3.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} - k_{AA} A \frac{A}{A+K_{AA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} - k_{CB} B \frac{B}{B+K_{CB}} - k_{BB} B \frac{B}{B+K_{BB}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{BC} B \frac{1-C}{1-C+K_{BC}} - k_{AC} A \frac{C}{C+K_{AC}}
\end{aligned}
\tag{6.3}
$$

Where: $K_{AA} = 7.633962$; $k_{AA} = 86.238263$; $K_{AB} = 20.265158$; $k_{AB} = 5.428752$; $K_{AC} = 0.258375; k_{AC} = 62.416585$; $K_{BA} = 0.003960$; $k_{BA} = 17.705166$; $K_{BB} = 31.604578$; $k_{BB} = 3.692326$; $K_{BC} = 44.386408$; $k_{BC} = 65.027941$; $K_{CB} = 0.701052$; $k_{CB} = 26.091557$; $K_{IA} = 0.464248$; $k_{IA} = 1.882348$;

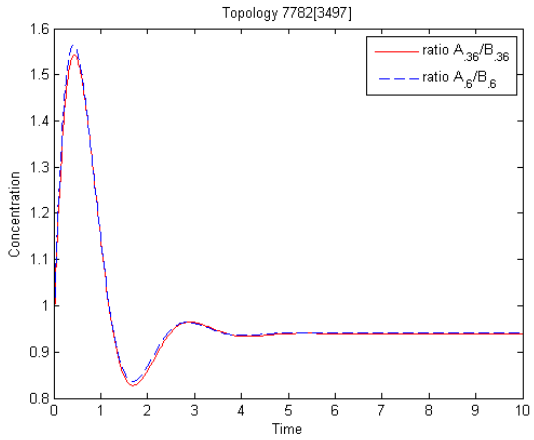



(a) Dynamics of A and B in Linearized model

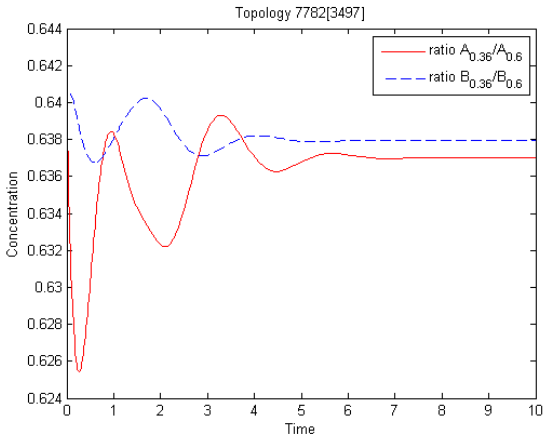



(b) Ouput from C Nonlinear Model

Figure 6.7: Topology 7782

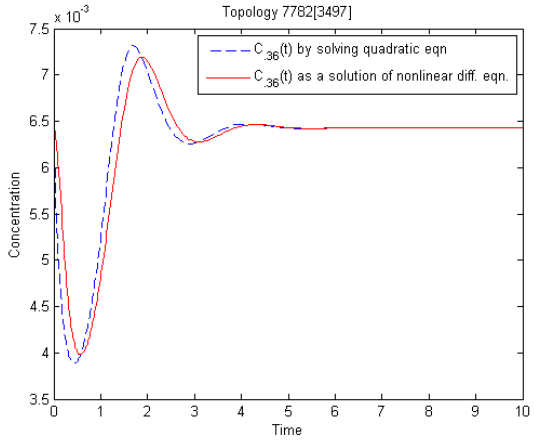



(a) A/B ratio

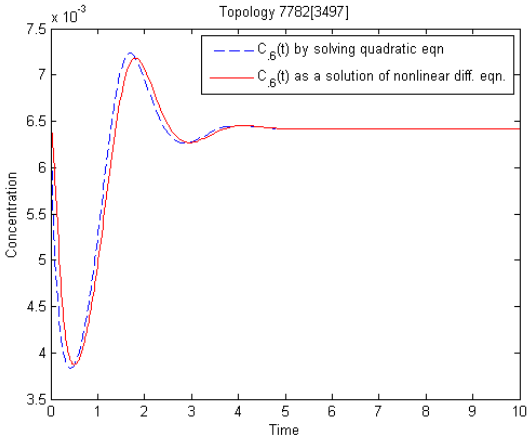



(b) AA and BB ratios

Figure 6.8: Topology 7782




(a) Quadratic approximation and Output of Nonlinear System



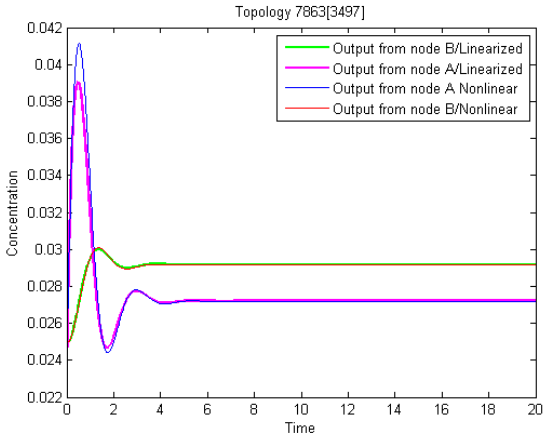

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.9: Topology 7782

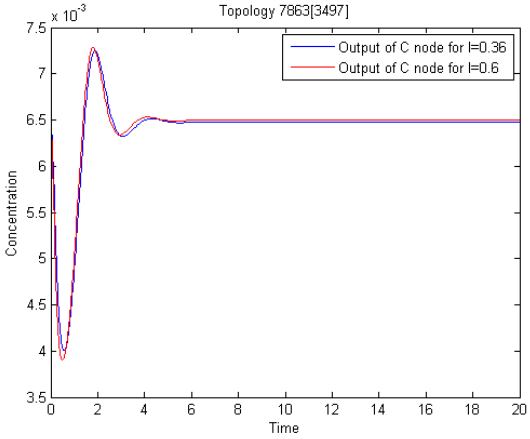
Circuit 4.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} - k_{AA} A \frac{A}{A+K_{AA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} - k_{CB} C \frac{B}{B+K_{CB}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{BC} B \frac{1-C}{1-C+K_{BC}} - k_{AC} A \frac{C}{C+K_{AC}}
\end{aligned}
\tag{6.4}
$$

Where: $Kaa = 7.633962$; $kaa = 86.238263$; $K_{AB} = 20.265158$; $k_{AB} = 5.428752$; $K_{AC} = 0.258375$; $k_{AC} = 62.416585$; $K_{BA} = 0.003960$; $k_{BA} = 17.705166$; $K_{BC} = 44.386408$; $k_{BC} = 65.027941$; $K_{CB} = 0.701052$; $k_{CB} = 26.091557$; $K_{IA} = 0.464248$; $k_{IA} = 1.882348$;

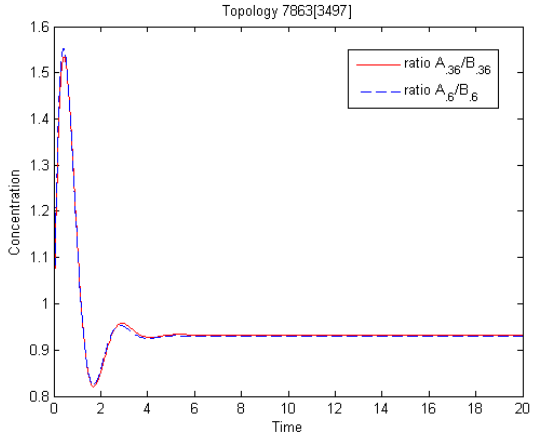



(a) Dynamics of A and B in Linearized model

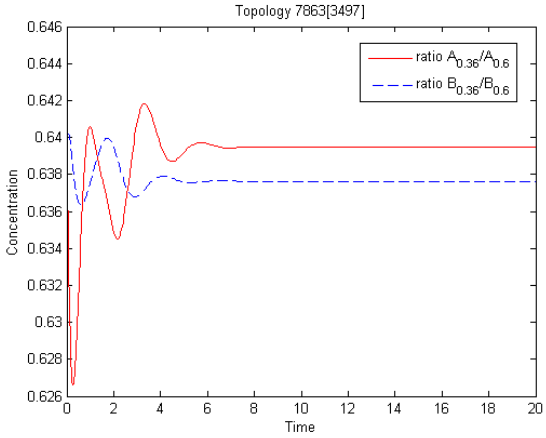



(b) Ouput from C Nonlinear Model

Figure 6.10: Topology 7863

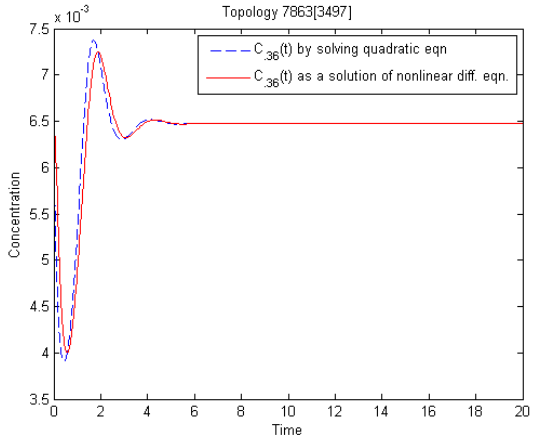



(a) A/B ratio

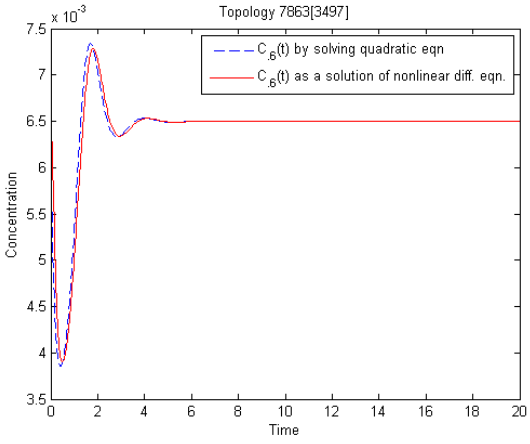



(b) AA and BB ratios

Figure 6.11: Topology 7863




(a) Quadratic approximation and Output of Nonlinear System



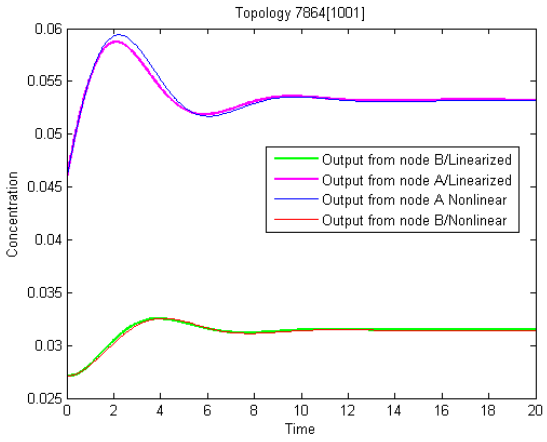

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.12: Topology 7863

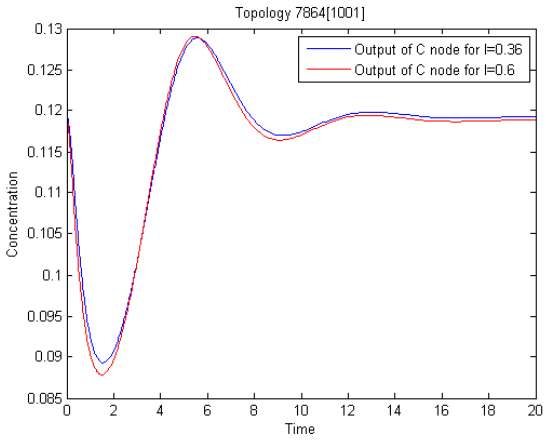
Circuit 5.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} - k_{CB} C \frac{B}{B+K_{CB}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{BC} B \frac{1-C}{1-C+K_{BC}} - k_{AC} A \frac{C}{C+K_{AC}}
\end{aligned} \tag{6.5}
$$

Where:$K_{AB} = 63.277600$; $k_{AB} = 6.638959$; $K_{AC} = 0.133429$; $k_{AC} = 55.731406$; $K_{BA} = 0.011188$; $k_{BA} = 2.749793$; $K_{BC} = 0.013374$; $k_{BC} = 45.175191$; $K_{CB} = 1.457975$; $k_{CB} = 2.114949$; $K_{IA} = 24.589517$; $k_{IA} = 5.346875$;

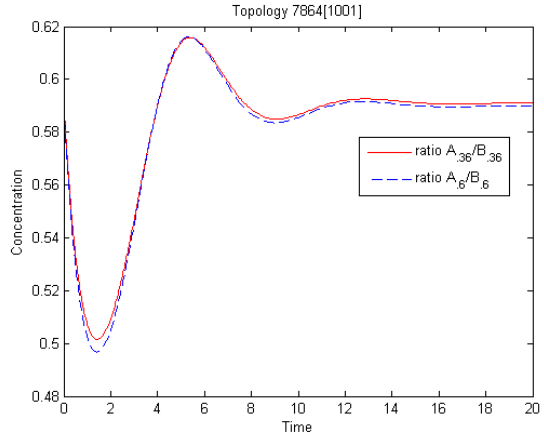



(a) Dynamics of A and B in Linearized model

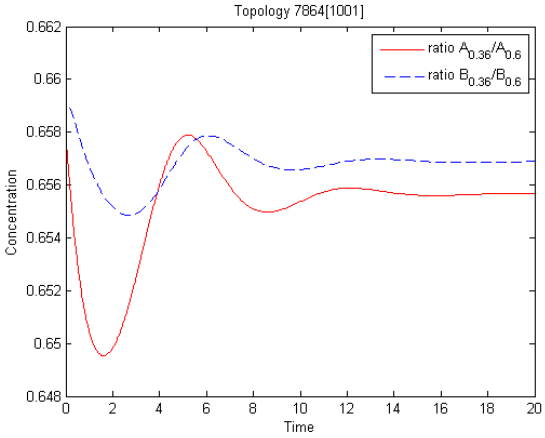



(b) Ouput from C Nonlinear Model

Figure 6.13: Topology 7864

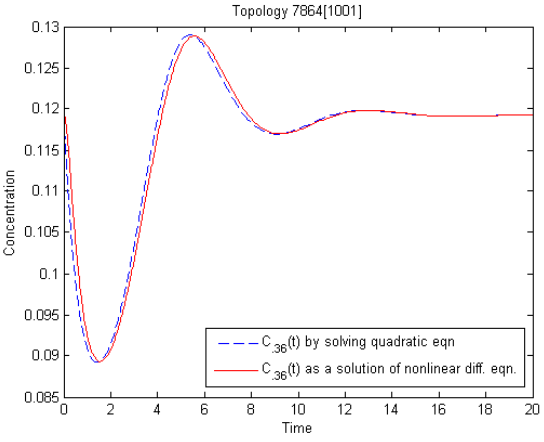



(a) A/B ratio

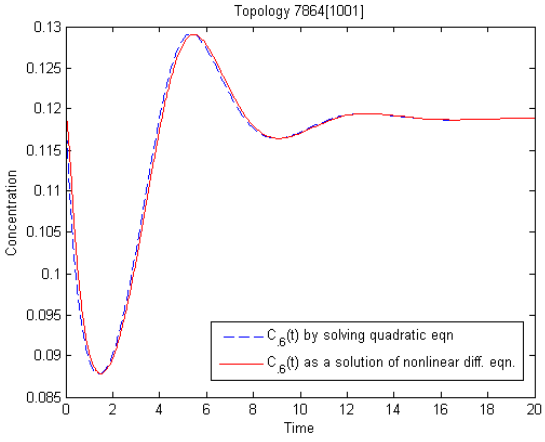



(b) AA and BB ratios

Figure 6.14: Topology 7864




(a) Quadratic approximation and Output of Nonlinear System



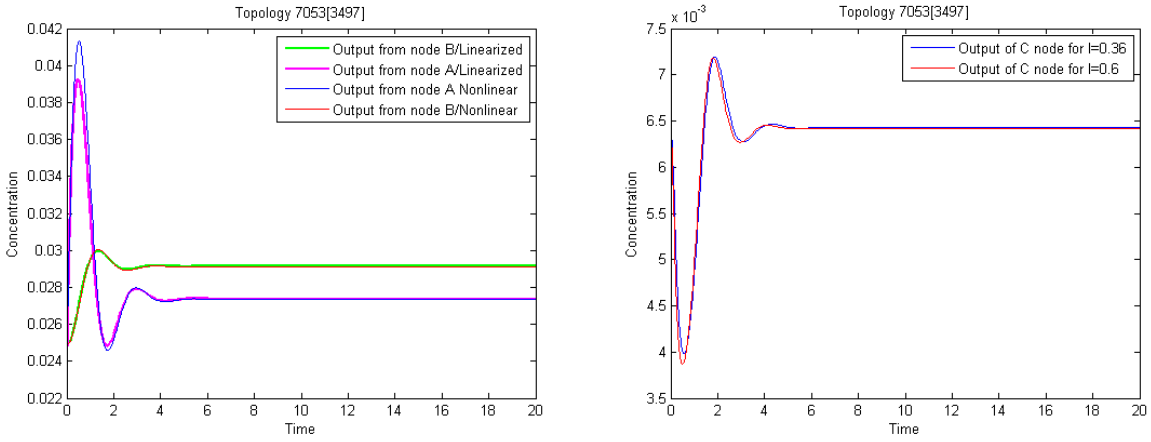

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.15: Topology 7864

Circuit 6.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} - -k_{AA} A \frac{A}{A+K_{AA}} - k_{CA} C \frac{A}{A+K_{CA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} - k_{CB} C \frac{B}{B+K_{CB}} - k_{BB} B \frac{B}{B+K_{BB}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{BC} B \frac{1-C}{1-C+K_{BC}} - k_{AC} A \frac{C}{C+K_{AC}}
\end{aligned}
\tag{6.6}
$$

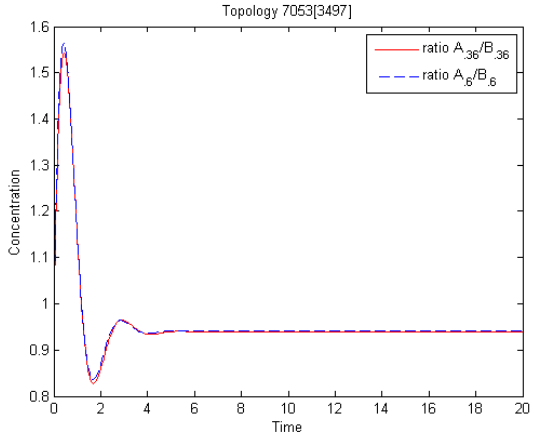
Where: $K_{AA} = 7.633962$; $k_{AA} = 86.238263$; $K_{AB} = 20.265158$; $k_{AB} = 5.428752$; $K_{AC} = 0.258375$; $k_{AC} = 62.416585$; $K_{BA} = 0.003960$; $k_{BA} = 17.705166$; $K_{BB} = 31.604578$; $k_{BB} = 3.692326$; $K_{BC} = 44.386408$; $k_{BC} = 65.027941$; $K_{CA} = 26.714681$; $k_{CA} = 2.806080$; $K_{CB} = 0.701052$; $k_{CB} = 26.091557$; $K_{IA} = 0.464248$; $k_{IA} = 1.882348$;



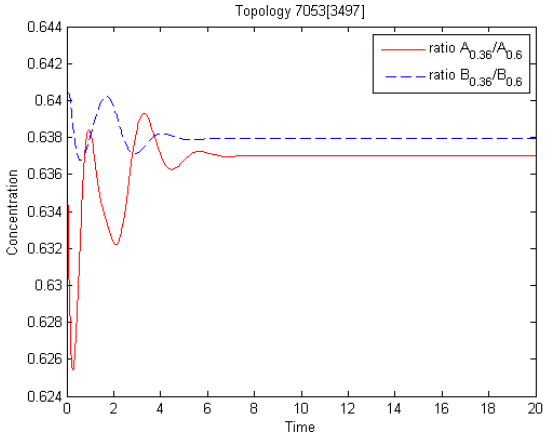

(a) Dynamics of A and B in Linearized model

(b) Ouput from C Nonlinear Model

Figure 6.16: Topology 7053

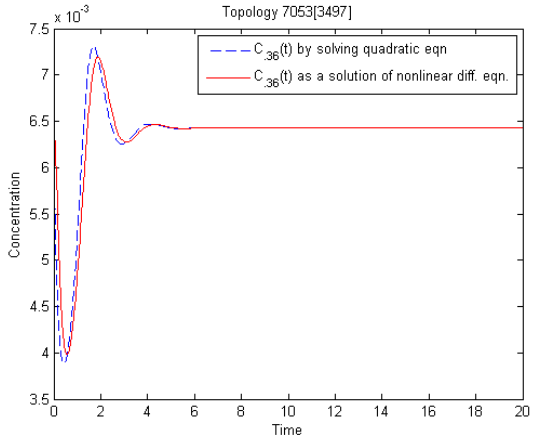



(a) A/B ratio

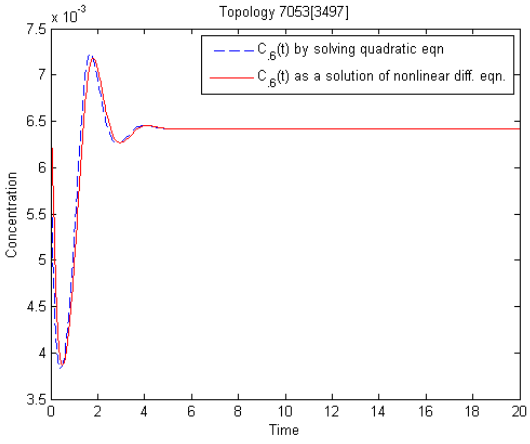



(b) AA and BB ratios

Figure 6.17: Topology 7053




(a) Quadratic approximation and Output of Nonlinear System



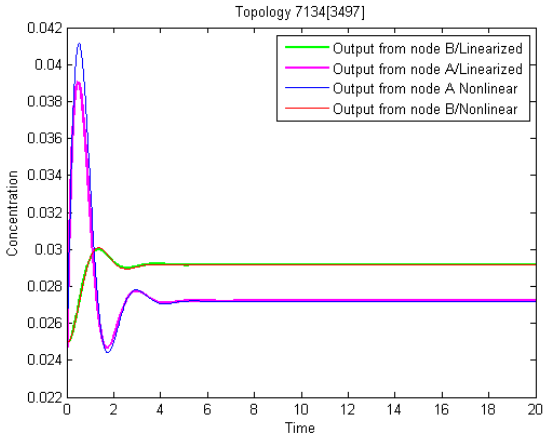

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.18: Topology 7053

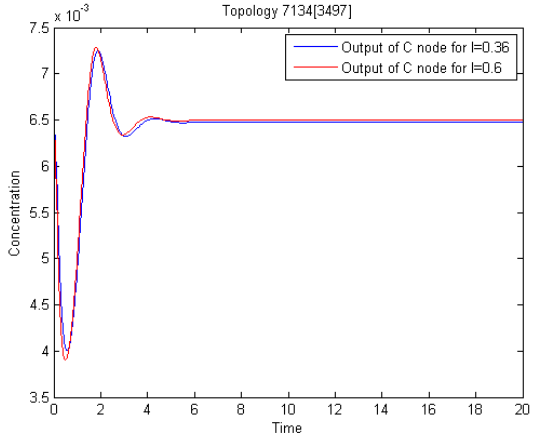
Circuit 7.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} - k_{AA} A \frac{A}{A+K_{AA}} - k_{CA} C \frac{A}{A+K_{CA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} - k_{CB} C \frac{B}{B+K_{CB}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{BC} B \frac{1-C}{1-C+K_{BC}} - k_{AC} A \frac{C}{C+K_{AC}}
\end{aligned}
\tag{6.7}
$$

Where $K_{AA} = 7.633962$; $k_{AA} = 86.238263$; $K_{AB} = 20.265158$; $k_{AB} = 5.428752$; $K_{AC} = 0.258375$; $k_{AC} = 62.416585$; $K_{BA} = 0.003960$; $k_{BA} = 17.705166$; $K_{BC} = 44.386408$; $k_{BC} = 65.027941$; $K_{CA} = 26.714681$; $k_{CA} = 2.806080$; $K_{CB} = 0.701052$; $k_{CB} = 26.091557$; $K_{IA} = 0.464248$; $k_{IA} = 1.882348$;

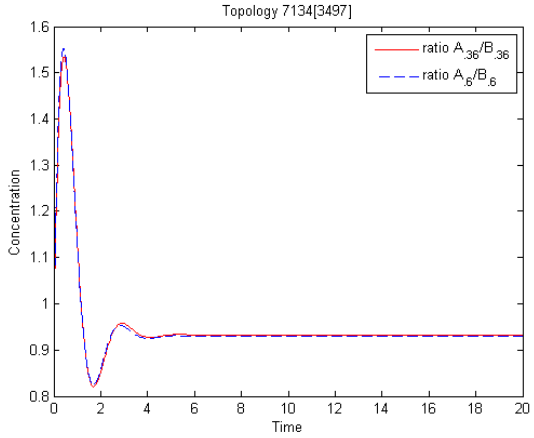



(a) Dynamics of A and B in Linearized model

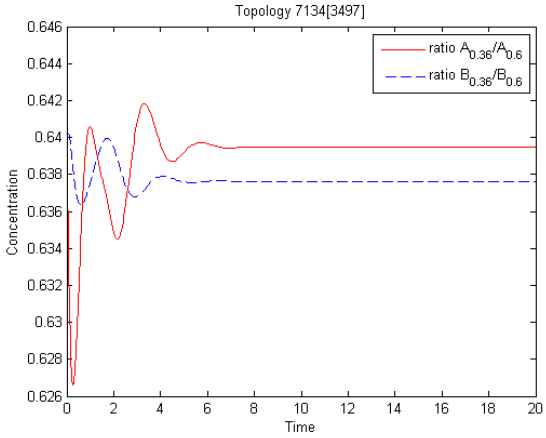



(b) Ouput from C Nonlinear Model

Figure 6.19: Topology 7134

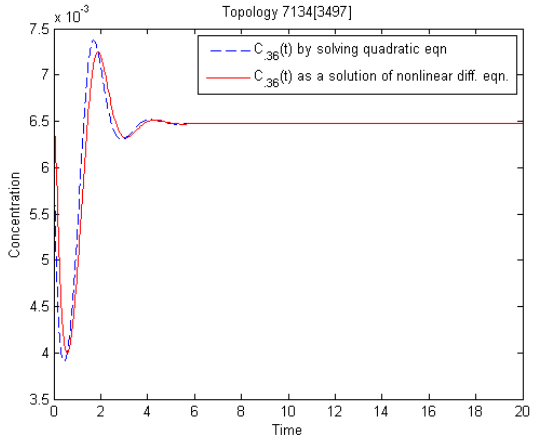



(a) A/B ratio

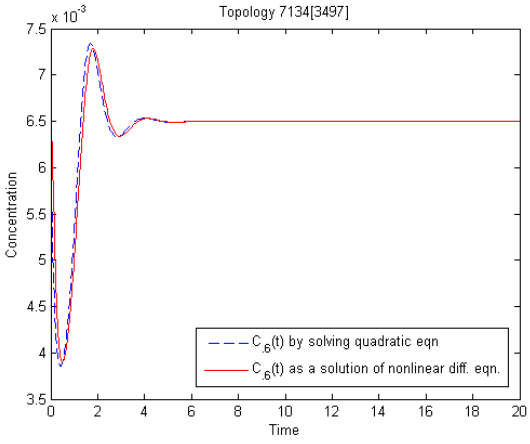



(b) AA and BB ratios

Figure 6.20: Topology 7134




(a) Quadratic approximation and Output of Nonlinear System



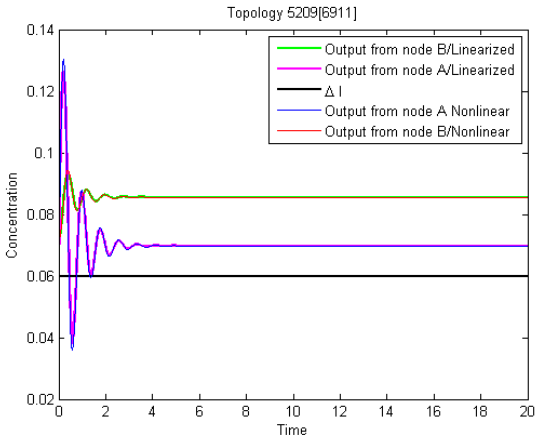

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.21: Topology 7134

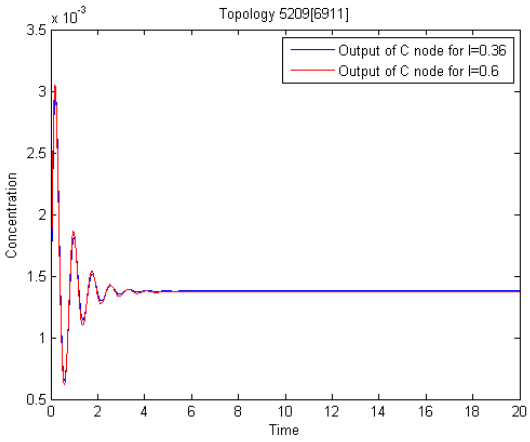
Circuit 8.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} - k_{F_B B} F_B \frac{B}{B+K_{F_B B}} + k_{CB} C \frac{1-B}{1-B+K_{CB}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{AC} A \frac{1-C}{1-C+K_{AC}} - k_{BC} B \frac{C}{C+K_{BC}} - k_{CC} C \frac{C}{C+K_{CC}}
\end{aligned}
\tag{6.8}
$$

Where: $K_{IA} = 0.093918$; $k_{IA} = 11.447219$; $K_{BA} = 0.001688$; $k_{BA} = 44.802268$; $K_{AB} = 0.001191$; $k_{AB} = 1.466561$; $K_{F_B} = 9.424319$; $k_{F_B} = 22.745736$; $K_{AC} = 0.113697$; $k_{AC} = 1.211993$; $K_{BC} = 0.009891$; $k_{BC} = 7.239357$; $K_{CB} = 30.602013$; $k_{CB} = 3.811536$; $K_{CC} = 0.189125$; $k_{CC} = 17.910182$;

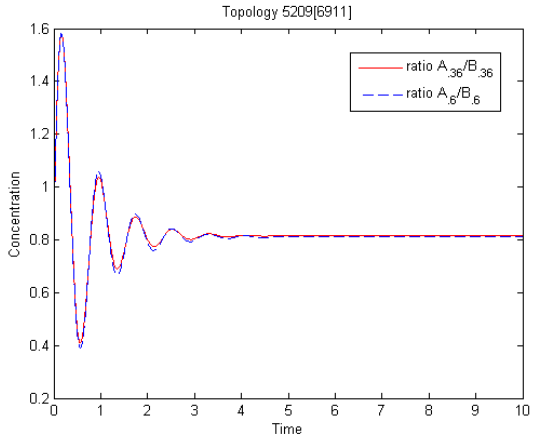



(a) Dynamics of A and B in Linearized model

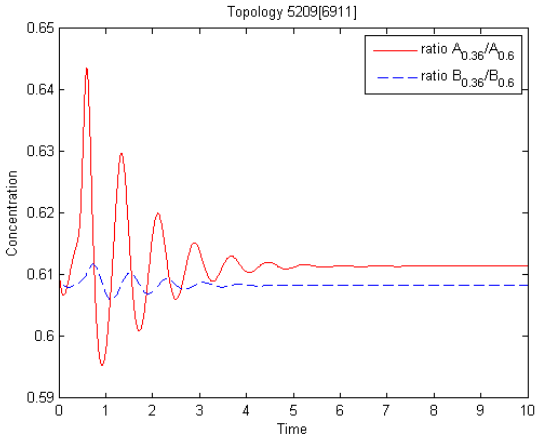



(b) Ouput from C Nonlinear Model

Figure 6.22: Topology 5209

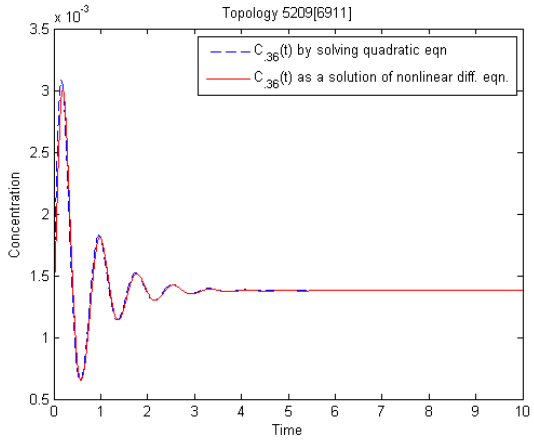



(a) A/B ratio

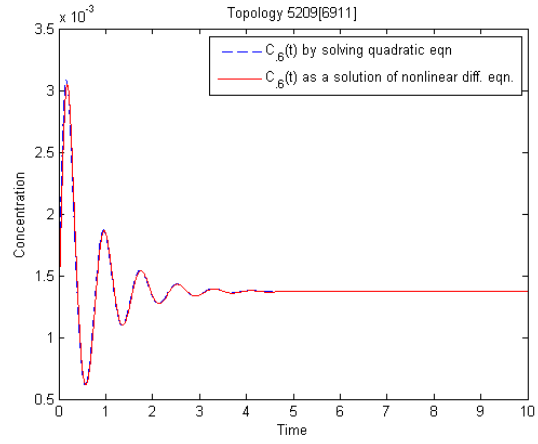



(b) AA and BB ratios

Figure 6.23: Topology 5209




(a) Quadratic approximation and Output of Nonlinear System



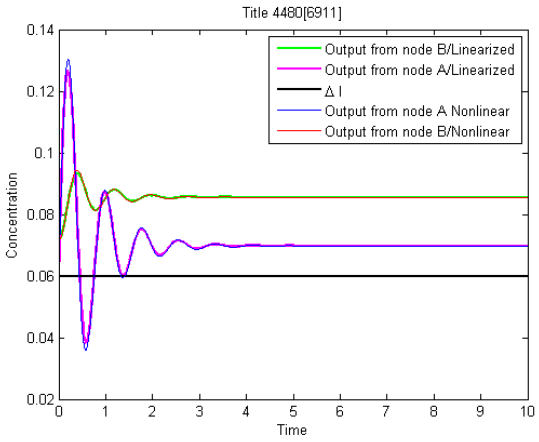

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.24: Topology 5209

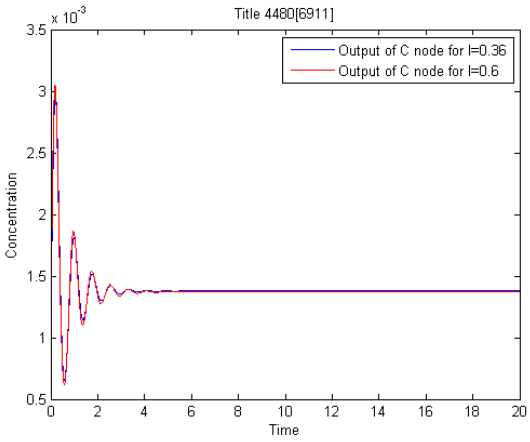
Circuit 9.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} - k_{CA} C \frac{A}{A+K_{CA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} + k_{CB} C \frac{1-B}{1-B+K_{CB}} - k_{F_B B} F_B \frac{B}{B+K_{F_B B}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{AC} A \frac{1-C}{1-C+K_{AC}} - k_{BC} B \frac{C}{C+K_{BC}} - k_{CC} C \frac{C}{C+K_{CC}}
\end{aligned}
\tag{6.9}
$$

Where: $K_{IA} = 0.093918$; $k_{IA} = 11.447219$; $K_{BA} = 0.001688$; $k_{BA} = 44.802268$; $K_{CA} = 90.209027$; $k_{CA} = 96.671843$; $K_{AB} = 0.001191$; $k_{AB} = 1.466561$; $K_{F_B} = 9.424319$; $k_{F_B} = 22.745736$; $K_{Ac} = 0.113697$; $k_{AC} = 1.211993$; $K_{BC} = 0.009891$; $k_{BC} = 7.239357$; $K_{CB} = 30.602013$; $k_{CB} = 3.811536$; $K_{CC} = 0.189125$; $k_{CC} = 17.910182$;

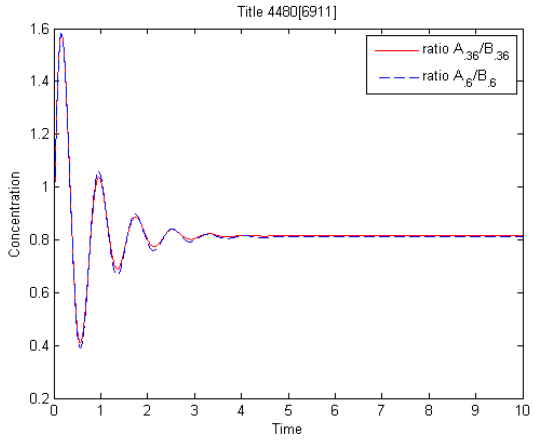



(a) Dynamics of A and B in Linearized model

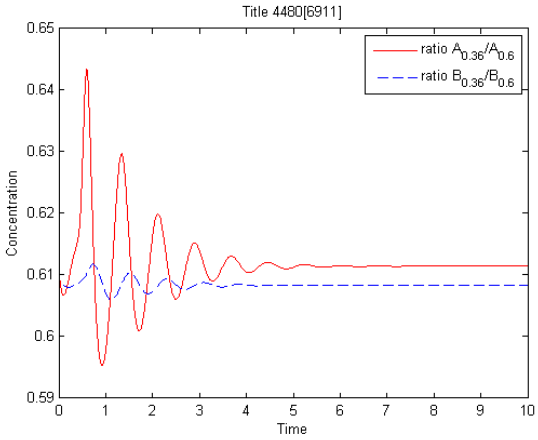



(b) Ouput from C Nonlinear Model

Figure 6.25: Topology 4480

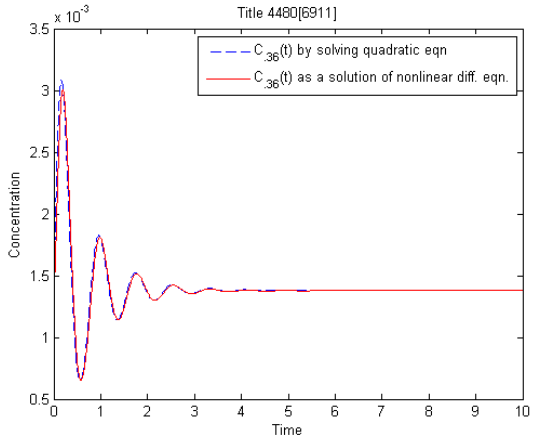



(a) A/B ratio

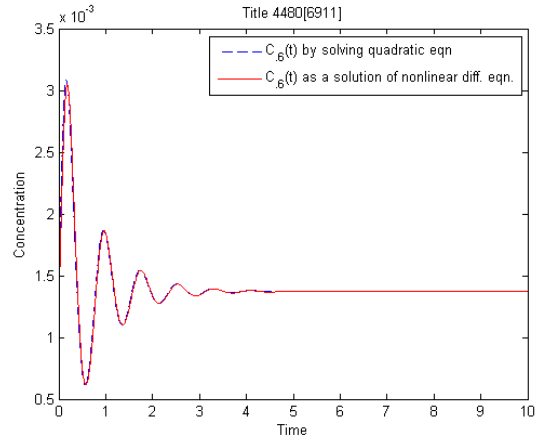



(b) AA and BB ratios

Figure 6.26: Topology 4480




(a) Quadratic approximation and Output of Nonlinear System



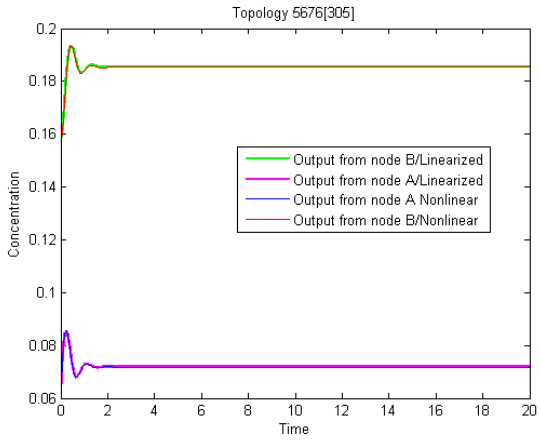

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.27: Topology 4480

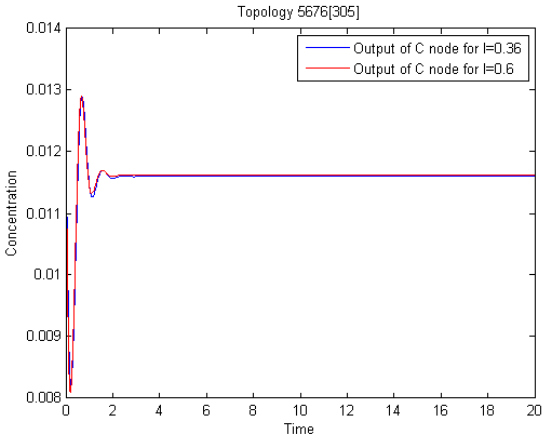
Circuit 10.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} - k_{AA} A \frac{A}{A+K_{AA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} + k_{CB} C \frac{1-B}{1-B+K_{CB}} - k_{F_B B} F_B \frac{B}{B+K_{F_B B}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{BC} B \frac{1-C}{1-C+K_{BC}} - k_{AC} A \frac{C}{C+K_{AC}} - k_{CC} C \frac{C}{C+K_{CC}}
\end{aligned}
\tag{6.10}
$$

Where: $K_{AA} = 24.989065$; $k_{AA} = 53.174082$; $K_{AB} = 0.444375$; $k_{AB} = 12.053134$; $K_{F_B} = 1.716920$; $k_{F_B} = 11.601122$; $K_{AC} = 0.013988$; $k_{AC} = 8.521185$; $K_{BA} = 0.005461$; $k_{BA} = 7.103952$; $K_{BC} = 51.850148$; $k_{BC} = 80.408137$; $K_{CB} = 5.392001$; $k_{CB} = 3.086740$; $K_{CC} = 1.962230$; $k_{CC} = 17.382010$; $K_{IA} = 4.387832$; $k_{IA} = 19.638124$;

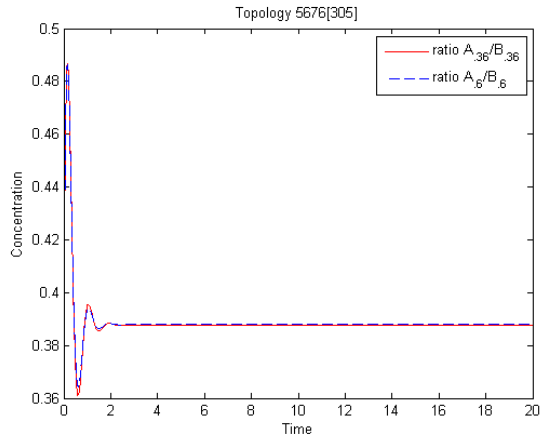



(a) Dynamics of A and B in Linearized model

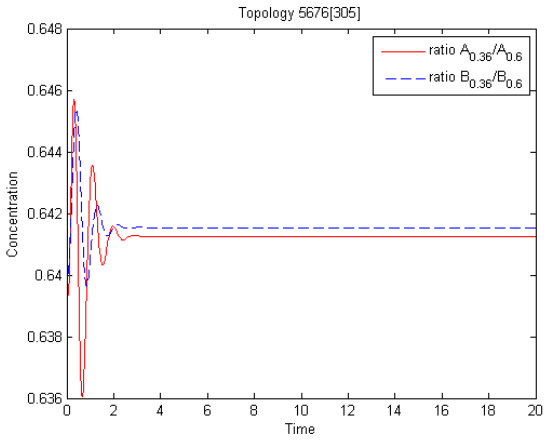



(b) Ouput from C Nonlinear Model

Figure 6.28: Topology 5676

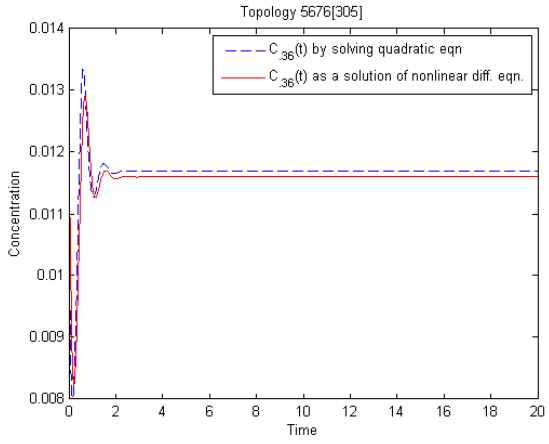



(a) A/B ratio

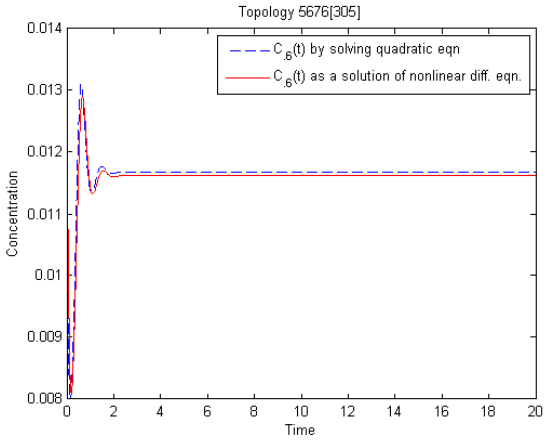



(b) AA and BB ratios

Figure 6.29: Topology 5676




(a) Quadratic approximation and Output of Nonlinear System



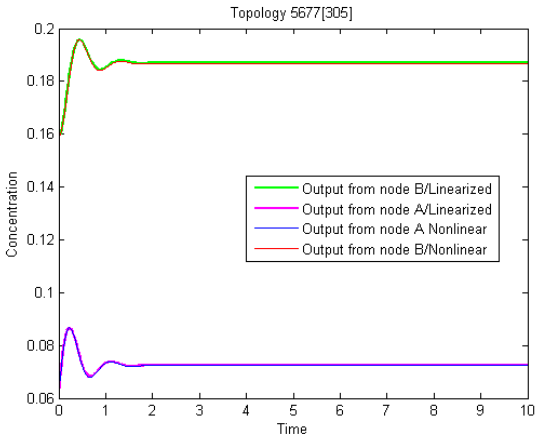

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.30: Topology 5676

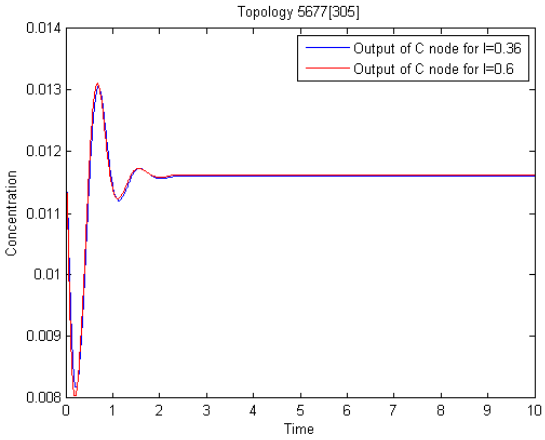
Circuit 11.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} + k_{CB} C \frac{1-B}{1-B+K_{CB}} - k_{F_B B} F_B \frac{B}{B+K_{F_B B}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{BC} B \frac{1-C}{1-C+K_{BC}} - k_{AC} A \frac{C}{C+K_{AC}} - k_{CC} C \frac{C}{C+K_{CC}}
\end{aligned}
\tag{6.11}
$$

Where: $K_{AB} = 0.444375$; $k_{AB} = 12.053134$; $K_{F_B} = 1.716920$; $k_{F_B} = 11.601122$; $K_{AC} = 0.013988$; $k_{AC} = 8.521185$; $K_{BA} = 0.005461$; $k_{BA} = 7.103952$; $K_{BC} = 51.850148$; $k_{BC} = 80.408137$; $K_{CB} = 5.392001$; $k_{CB} = 3.086740$; $K_{CC} = 1.962230$; $k_{CC} = 17.382010$; $K_{IA} = 4.387832$; $k_{IA} = 19.638124$;

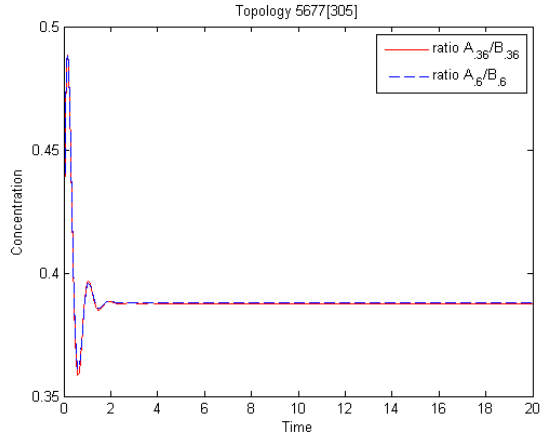



(a) Dynamics of A and B in Linearized model

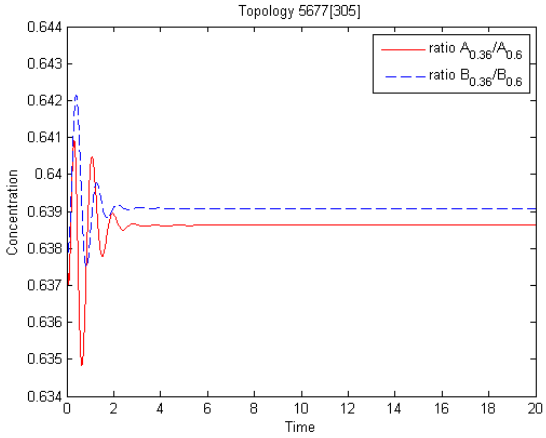



(b) Ouput from C Nonlinear Model

Figure 6.31: Topology 5677

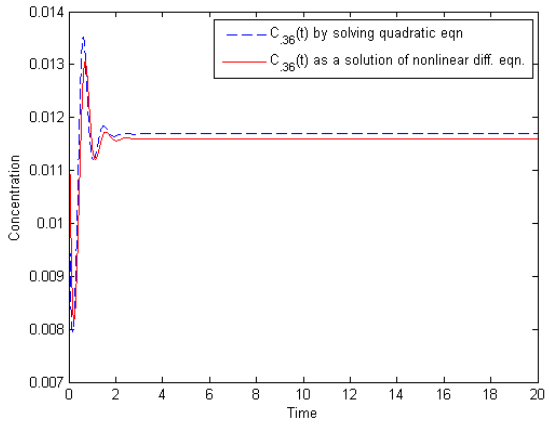



(a) A/B ratio

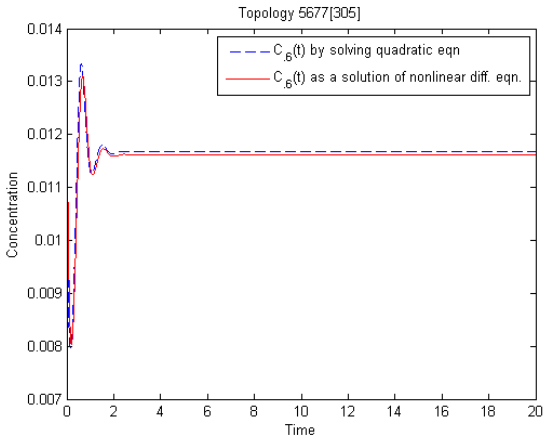



(b) AA and BB ratios

Figure 6.32: Topology 5677




(a) Quadratic approximation and Output of Nonlinear System



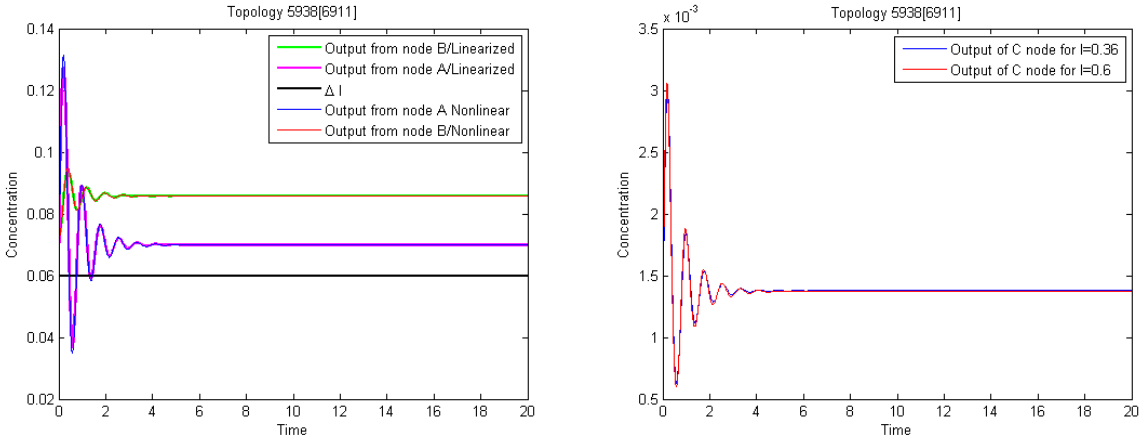

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.33: Topology 5677

Circuit 12.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} + k_{CA} C \frac{1-A}{1-A+K_{CA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} + k_{CB} C \frac{1-B}{1-B+K_{CB}} - k_{F_B B} F_B \frac{B}{B+K_{F_B B}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{AC} A \frac{1-C}{1-C+K_{AC}} - k_{BC} B \frac{C}{C+K_{BC}} - k_{CC} C \frac{C}{C+K_{CC}}
\end{aligned}
\tag{6.12}
$$

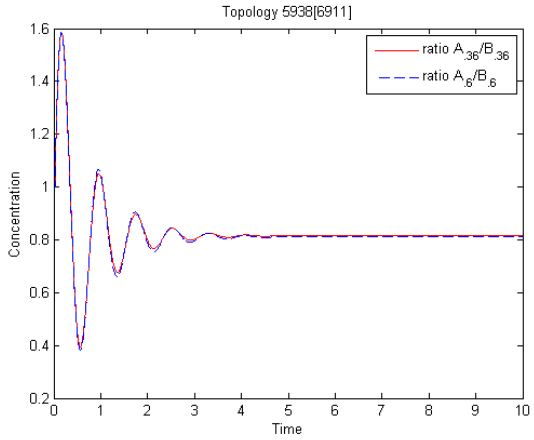
Where: $K_{IA} = 0.093918$; $k_{IA} = 11.447219$; $K_{BA} = 0.001688$; $k_{BA} = 44.802268$; $K_{CA} = 5.026318$; $k_{CA} = 45.803641$; $K_{AB} = 0.001191$; $k_{AB} = 1.466561$; $K_{F_B} = 9.424319$; $k_{F_B} = 22.745736$; $K_{AC} = 0.113697$; $k_{AC} = 1.211993$; $K_{BC} = 0.009891$; $k_{BC} = 7.239357$; $K_{CB} = 30.602013$; $k_{CB} = 3.811536$; $K_{CC} = 0.189125$; $k_{CC} = 17.910182$;



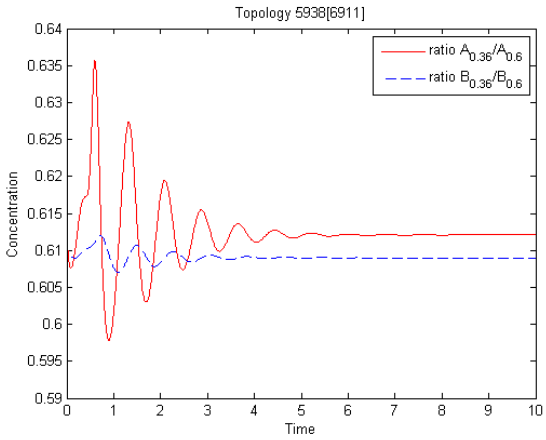

(a) Dynamics of A and B in Linearized model

(b) Ouput from C Nonlinear Model

Figure 6.34: Topology 5938

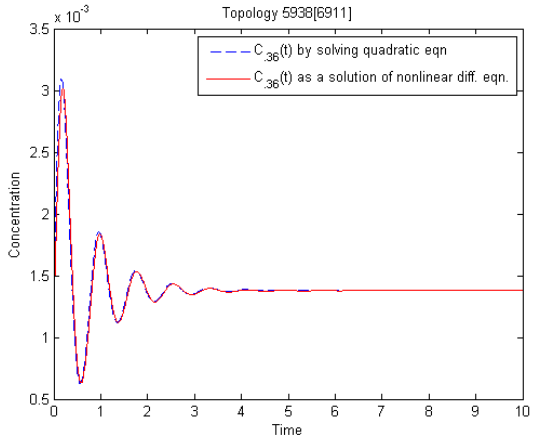



(a) A/B ratio

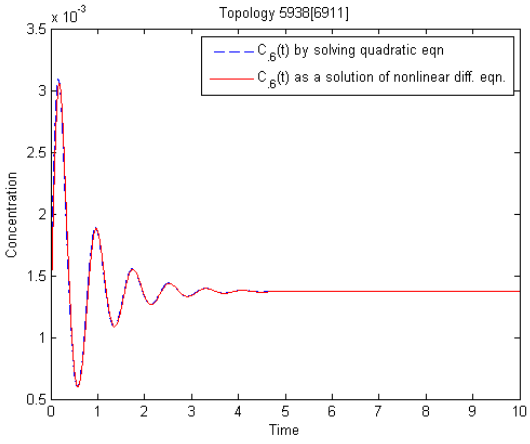



(b) AA and BB ratios

Figure 6.35: Topology 5938




(a) Quadratic approximation and Output of Nonlinear System



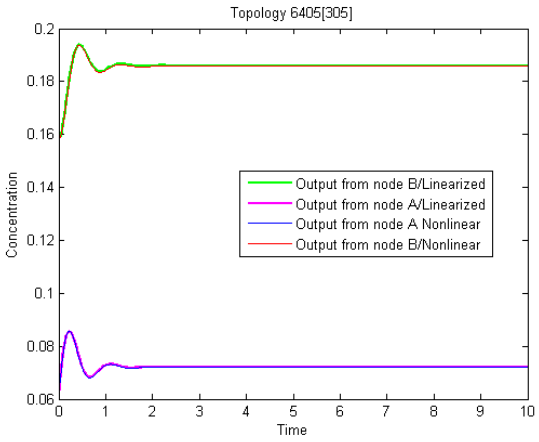

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.36: Topology 5938

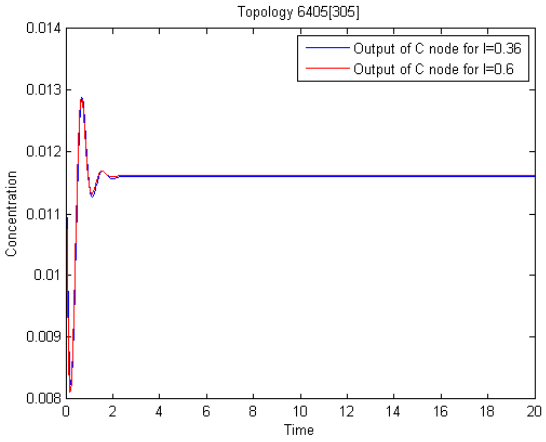
Circuit 13.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} - k_{AA} A \frac{A}{A+K_{AA}} + k_{CA} C \frac{1-A}{1-A+K_{CA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} + k_{CB} C \frac{1-B}{1-B+K_{CB}} - k_{BB} B \frac{B}{B+K_{BB}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{BC} B \frac{1-C}{1-C+K_{BC}} - k_{AC} A \frac{C}{C+K_{AC}} - k_{CC} A \frac{C}{C+K_{CC}}
\end{aligned}
\tag{6.13}
$$

Where: $K_{AA} = 24.989065$; $k_{AA} = 53.174082$; $K_{AB} = 0.444375$; $k_{AB} = 12.053134$; $K_{F_B} = 1.716920$; $k_{F_B} = 11.601122$; $K_{AC} = 0.013988$; $k_{AC} = 8.521185$; $K_{BA} = 0.005461$; $k_{BA} = 7.103952$; $K_{BC} = 51.850148$; $k_{BC} = 80.408137$; $K_{CB} = 5.392001$; $k_{CB} = 3.086740$; $K_{CC} = 1.962230$; $k_{CC} = 17.382010$; $K_{IA} = 4.387832$; $k_{IA} = 19.638124$; $K_{CA} = 15.479253$; $k_{CA} = 4.903430$;

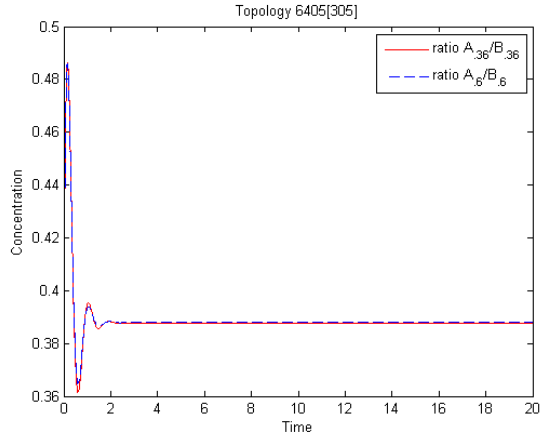



(a) Dynamics of A and B in Linearized model

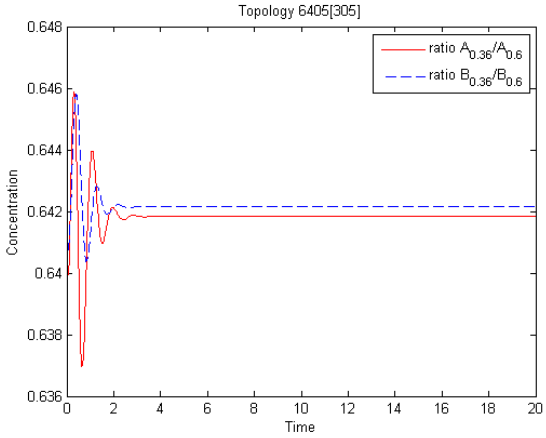



(b) Ouput from C Nonlinear Model

Figure 6.37: Topology 6405

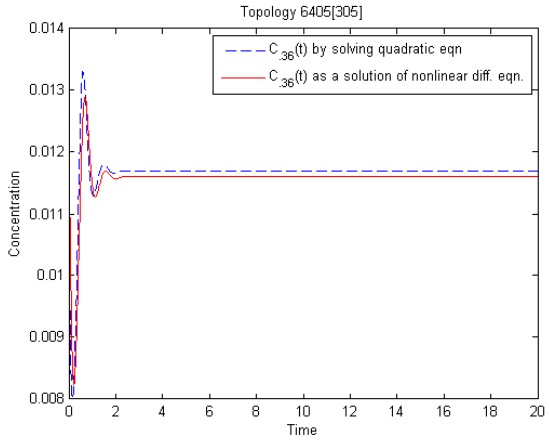



(a) A/B ratio

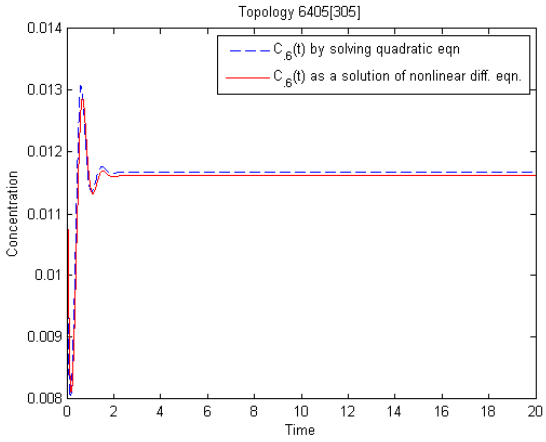



(b) AA and BB ratios

Figure 6.38: Topology 6405




(a) Quadratic approximation and Output of Nonlinear System



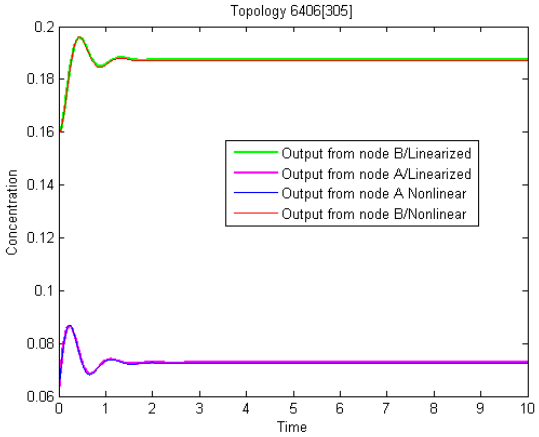

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.39: Topology 6405

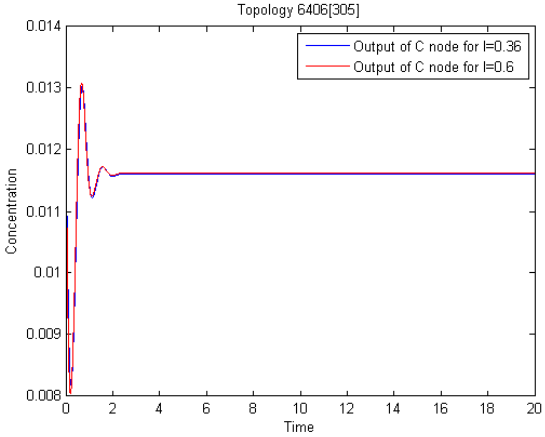
Circuit 14.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} + k_{CA} C \frac{1-A}{1-A+K_{CA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} + k_{CB} C \frac{1-B}{1-B+K_{CB}} - k_{BB} B \frac{B}{B+K_{BB}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{BC} B \frac{1-C}{1-C+K_{BC}} - k_{AC} A \frac{C}{C+K_{AC}} - k_{CC} A \frac{C}{C+K_{CC}}
\end{aligned}
\tag{6.14}
$$

Where: $K_{AB} = 0.444375$; $k_{AB} = 12.053134$; $K_{F_B} 1.716920$; $k_{F_B} = 11.601122$; $K_{AC} = 0.013988$; $k_{AC} = 8.521185$; $K_{BA} = 0.005461$; $k_{BA} = 7.103952$; $K_{BC} = 51.850148$; $k_{BC} = 80.408137$; $Kcb = 5.392001$; $kcb = 3.086740$; $K_{CC} = 1.962230$; $k_{CC} = 17.382010$; $K_{IA} = 4.387832$; $k_{IA} = 19.638124$; $K_{CA} = 15.479253$; $k_{CA} = 4.903430$;

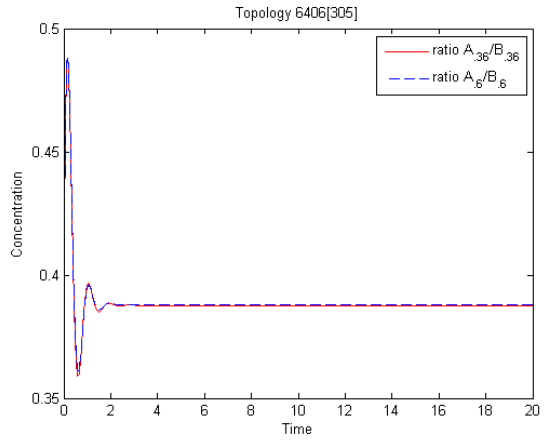



(a) Dynamics of A and B in Linearized model

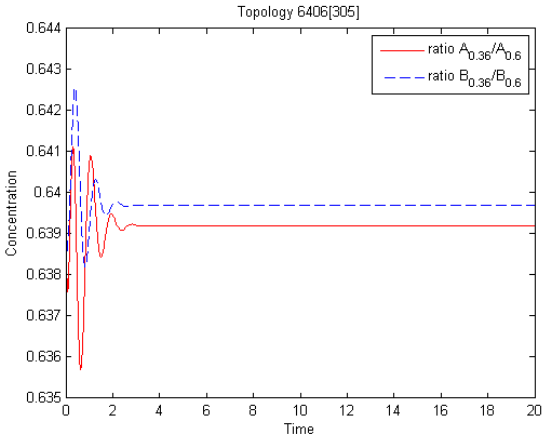



(b) Ouput from C Nonlinear Model

Figure 6.40: Topology 6406

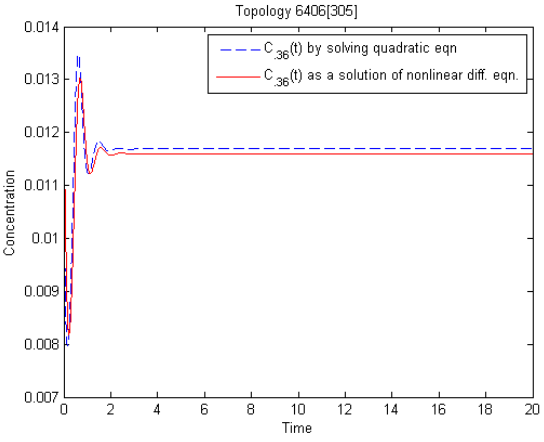



(a) A/B ratio

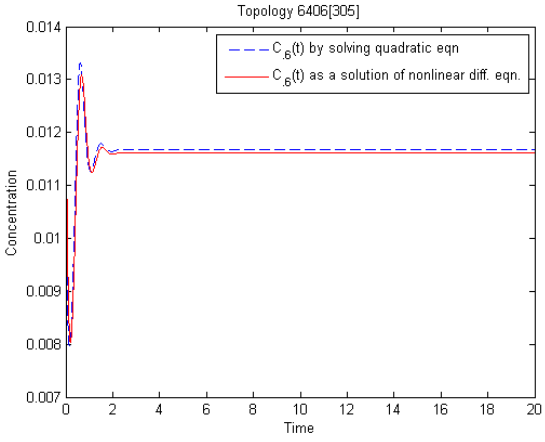



(b) AA and BB ratios

Figure 6.41: Topology 6406




(a) Quadratic approximation and Output of Nonlinear System



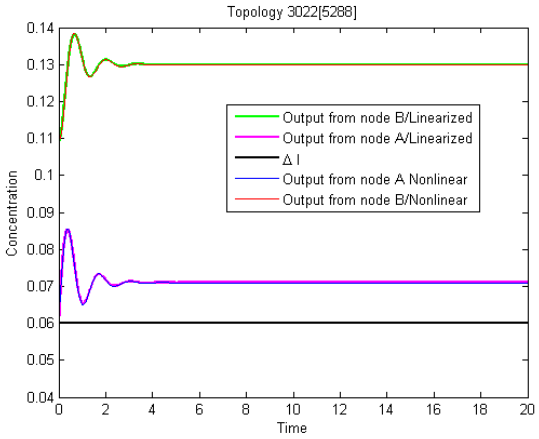

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.42: Topology 6406

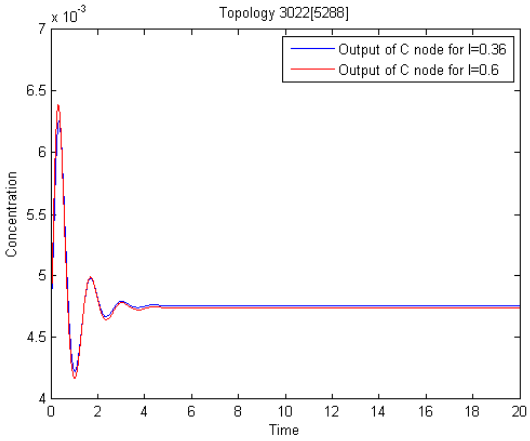
Circuit 15.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} - k_{F_B B} F_B \frac{B}{B+K_{F_B B}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{AC} A \frac{1-C}{1-C+K_{AC}} - k_{BC} B \frac{C}{C+K_{BC}} - k_{CC} A \frac{C}{C+K_{CC}}
\end{aligned}
\tag{6.15}
$$

Where: $K_{AB} = 0.709169$; $k_{AB} = 7.445605$; $K_{F_B} = 1.495375$; $k_{F_B} = 7.282827$; $K_{AC} = 0.002566$; $k_{AC} = 1.115065$; $K_{BA} = 0.002522$; $k_{BA} = 5.753075$; $K_{BC} = 0.017051$; $k_{BC} = 2.777794$; $K_{CC} = 0.195997$; $k_{CC} = 1.480130$; $K_{IA} = 0.225814$; $k_{IA} = 2.492872$;

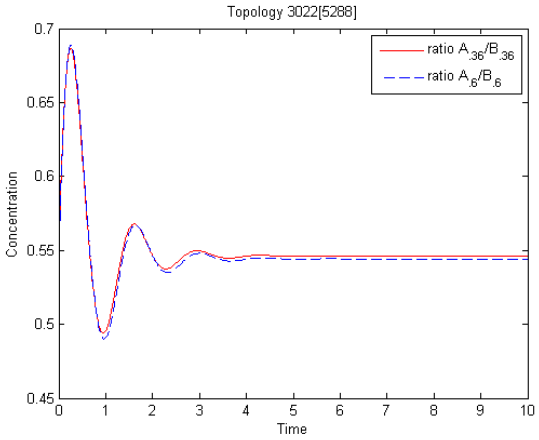



(a) Dynamics of A and B in Linearized model

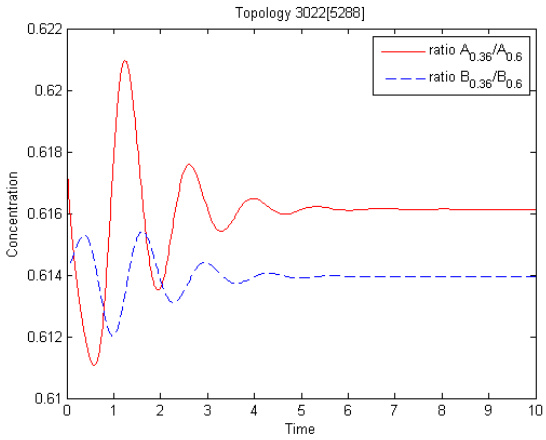



(b) Ouput from C Nonlinear Model

Figure 6.43: Topology 3022

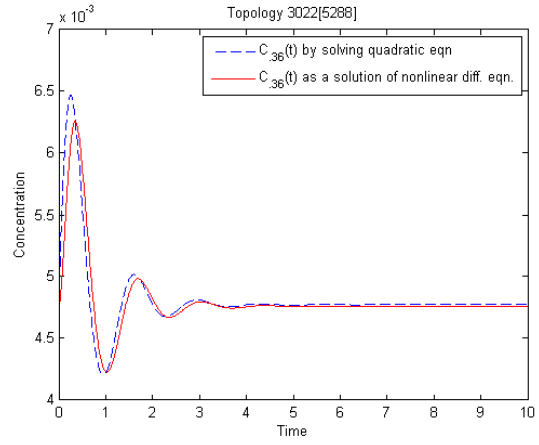



(a) A/B ratio

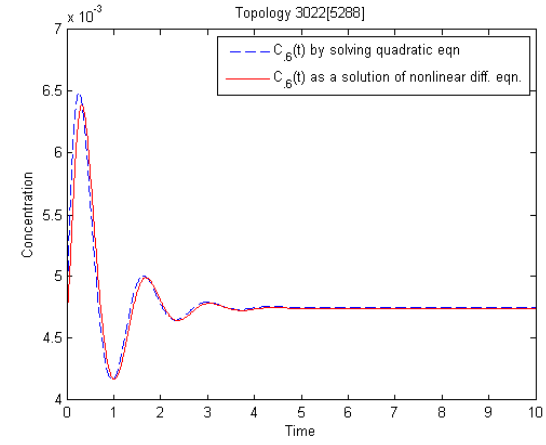



(b) AA and BB ratios

Figure 6.44: Topology 3022




(a) Quadratic approximation and Output of Nonlinear System

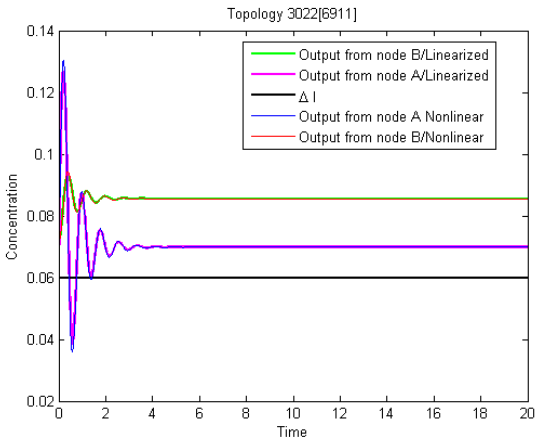



(b) Quadratic approximation and Output of Nonlinear System

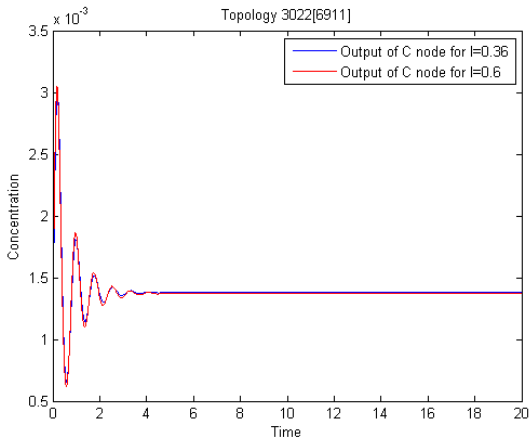
Figure 6.45: Topology 3022

Circuit 16.

This is the same topology as in previous case, only different parameter set was used: $K_{AB} = 0.001191$; $k_{AB} = 1.466561$; $K_{F_B} = 9.424319$; $k_{F_B} = 22.745736$; $K_{AC} = 0.113697$; $k_{AC} = 1.211993$; $K_{BA} = 0.001688$; $k_{BA} = 44.802268$; $K_{BC} = 0.009891$; $k_{BC} = 7.239357$; $K_{CC} = 0.189125$; $k_{CC} = 17.910182$; $K_{IA} = 0.093918$; $k_{IA} = 11.447219$;

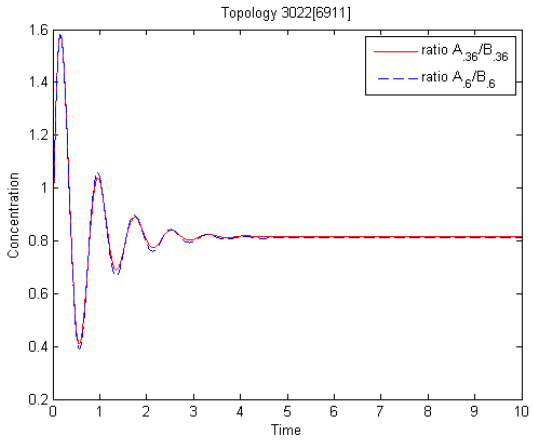



(a) Dynamics of A and B in Linearized model

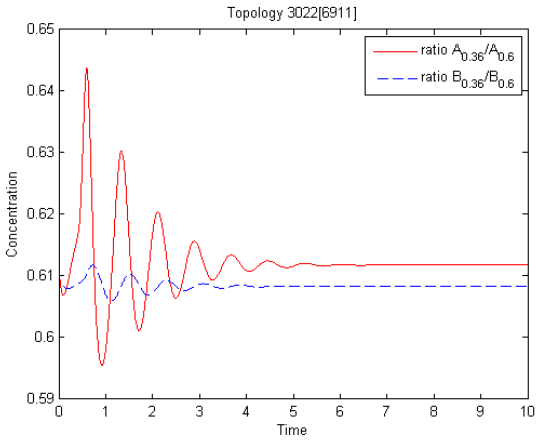



(b) Ouput from C Nonlinear Model

Figure 6.46: Topology 3022

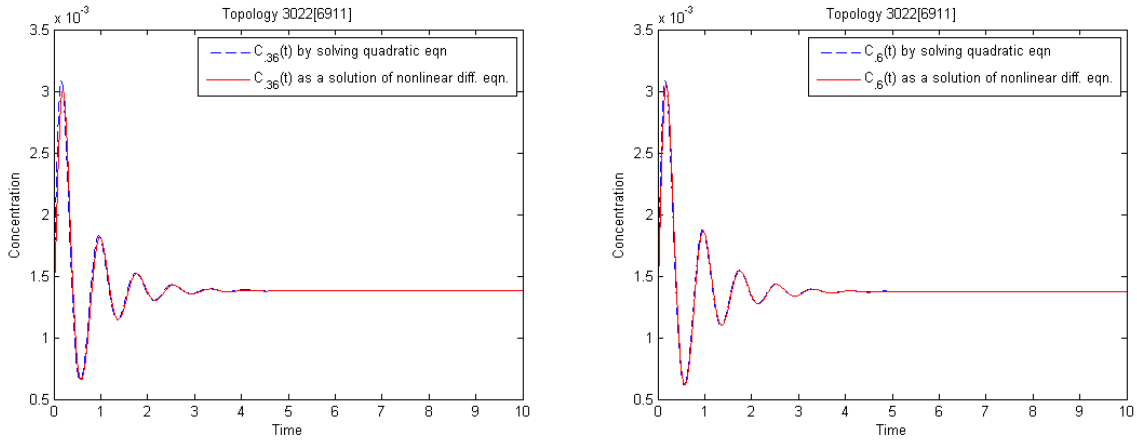



(a) A/B ratio




(b) AA and BB ratios

Figure 6.47: Topology 3022



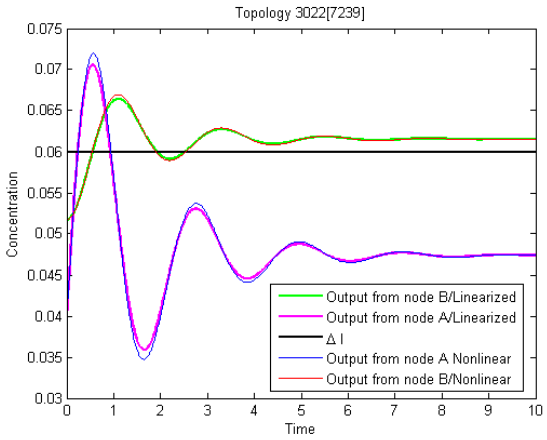

(a) Quadratic approximation and Output of Nonlinear System

(b) Quadratic approximation and Output of Nonlinear System

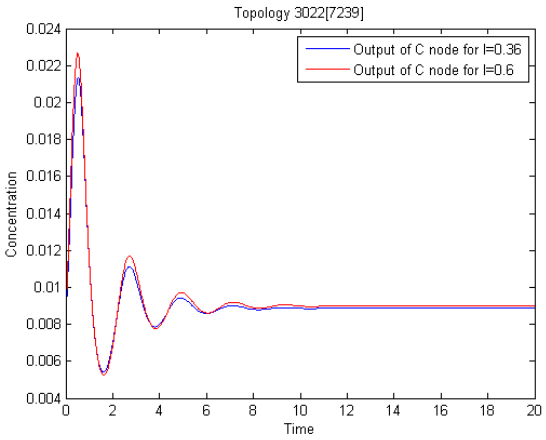
Figure 6.48: Topology 3022

Circuit 17.

This is the same topology as in previous case, only different parameter set was used: $K_{AB} = 1.620877$; $k_{AB} = 2.306216$; $K_{F_B} = 2.012565$; $k_{F_B} = 2.700847$; $K_{AC} = 0.010933$; $k_{AC} = 8.968091$; $K_{BA} = 0.001812$; $k_{BA} = 10.039221$; $K_{BC} = 0.014199$; $k_{BC} = 17.762333$; $K_{CC} = 2.686891$; $k_{CC} = 4.139044$; $K_{IA} = 0.161715$; $k_{IA} = 1.933303$;

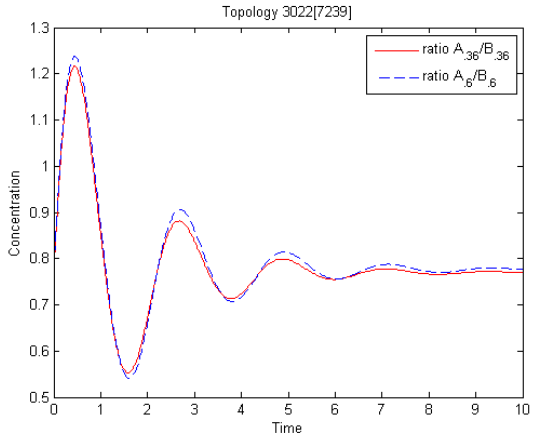



(a) Dynamics of A and B in Linearized model

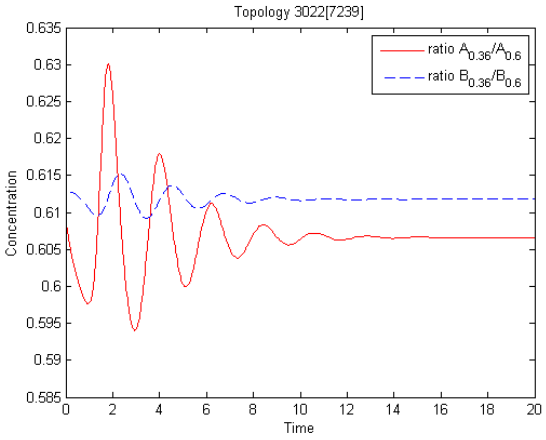



(b) Ouput from C Nonlinear Model

Figure 6.49: Topology 3022

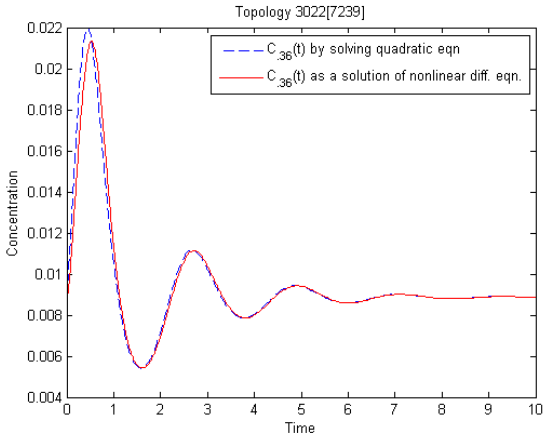



(a) A/B ratio

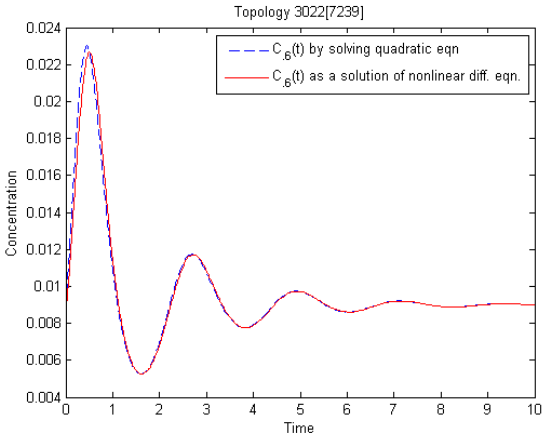



(b) AA and BB ratios

Figure 6.50: Topology 3022




(a) Quadratic approximation and Output of Nonlinear System



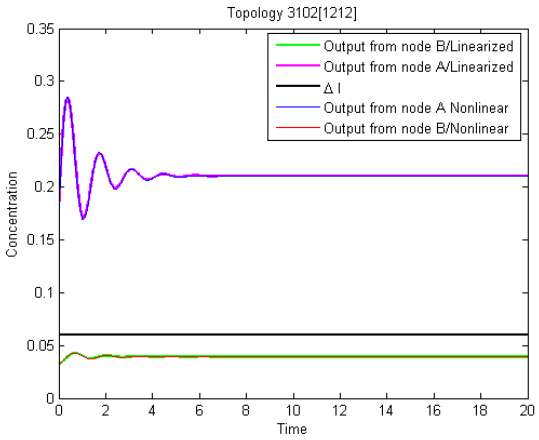

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.51: Topology 3022

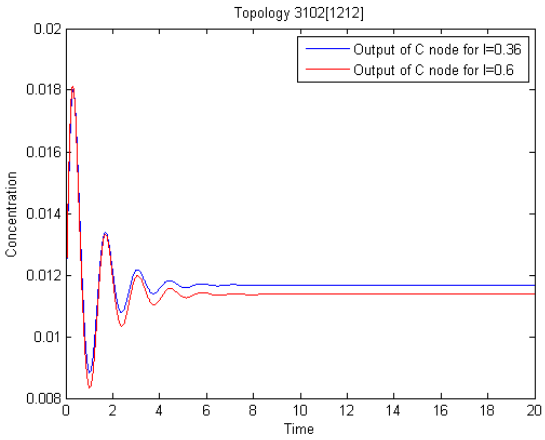
Circuit 18.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} - k_{AA} A \frac{A}{A+K_{AA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} + k_{BB} B \frac{1-B}{1-B+K_{BB}} - k_{F_B B} F_B \frac{B}{B+K_{F_B B}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{AC} A \frac{1-C}{1-C+K_{AC}} - k_{BC} B \frac{C}{C+K_{BC}} - k_{CC} C \frac{C}{C+K_{CC}}
\end{aligned}
\tag{6.16}
$$

Where: $K_{AA} = 17.569120$; $k_{AA} = 2.198366$; $K_{AB} = 9.435176$; $k_{AB} = 3.134007$; $K_{F_B} = 0.469083$; $k_{F_B} = 1.934194$; $K_{AC} = 0.062914$; $k_{AC} = 2.742206$; $K_{BA} = 0.003245$; $k_{BA} = 75.352905$; $K_{BB} = 27.463128$; $k_{BB} = 10.551155$; $K_{BC} = 0.041615$; $k_{BC} = 61.333818$; $K_{CC} = 0.039332$; $k_{CC} = 4.756637$; $K_{IA} = 0.005167$; $k_{IA} = 8.186533$;

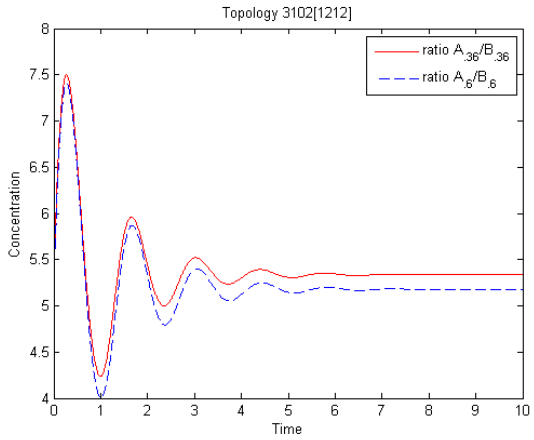



(a) Dynamics of A and B in Linearized model

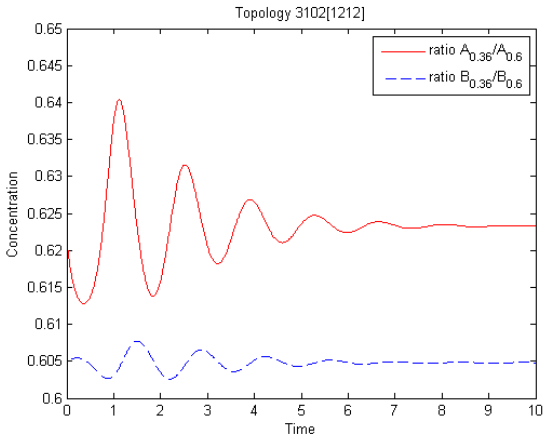



(b) Ouput from C Nonlinear Model

Figure 6.52: Topology 3102

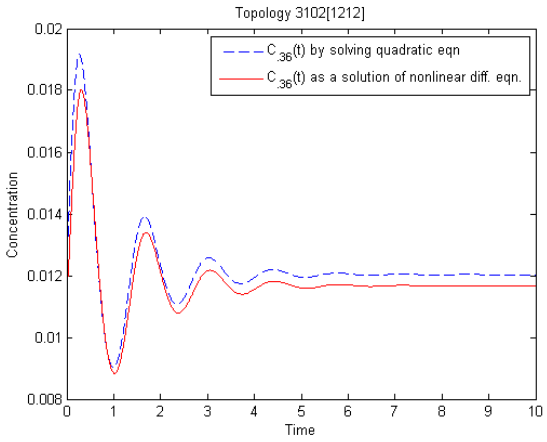



(a) A/B ratio

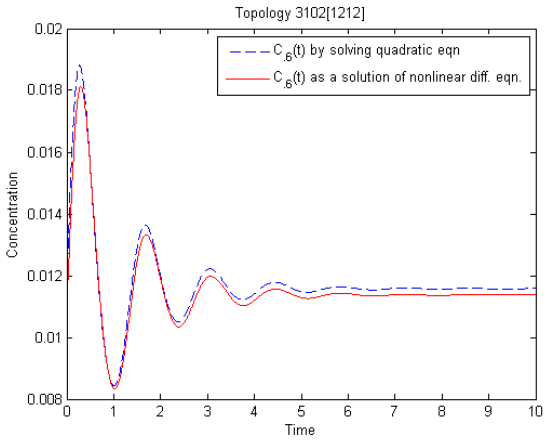



(b) AA and BB ratios

Figure 6.53: Topology 3102




(a) Quadratic approximation and Output of Nonlinear System



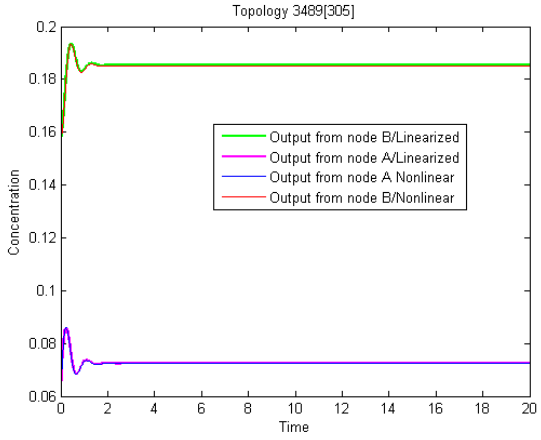

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.54: Topology 3102

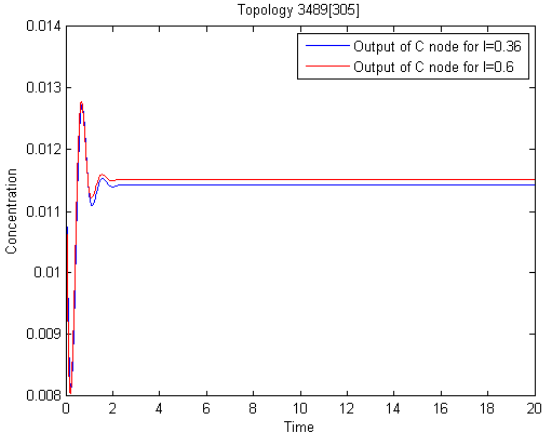
Circuit 19.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} - k_{AA} A \frac{A}{A+K_{AA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} - k_{F_B B} F_B \frac{B}{B+K_{F_B B}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{BC} B \frac{1-C}{1-C+K_{BC}} - k_{AC} A \frac{C}{C+K_{AC}} - k_{CC} C \frac{C}{C+K_{CC}}
\end{aligned}
\tag{6.17}
$$

Where: $K_{IA} = 4.387832$; $k_{IA} = 19.638124$; $K_{BA} = 0.005461$; $k_{BA} = 7.103952$; $K_{AA} = 24.989065$; $k_{AA} = 53.174082$; $K_{AB} = 0.444375$; $k_{AB} = 12.053134$; $K_{F_B} = 1.716920$; $k_{F_B} = 11.601122$; $K_{BC} = 51.850148$; $k_{BC} = 80.408137$; $K_{AC} = 0.013988$; $k_{AC} = 8.521185$; $K_{CC} = 1.962230$; $k_{CC} = 17.382010$;

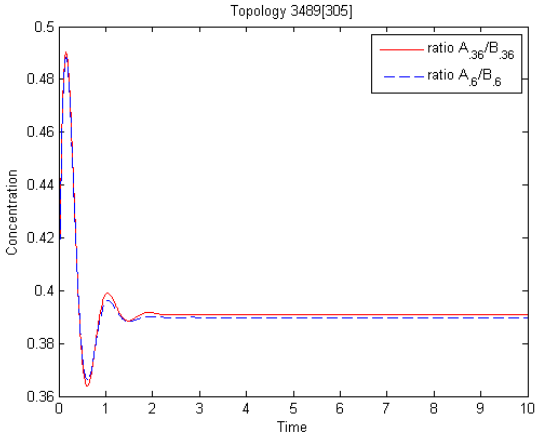



(a) Dynamics of A and B in Linearized model

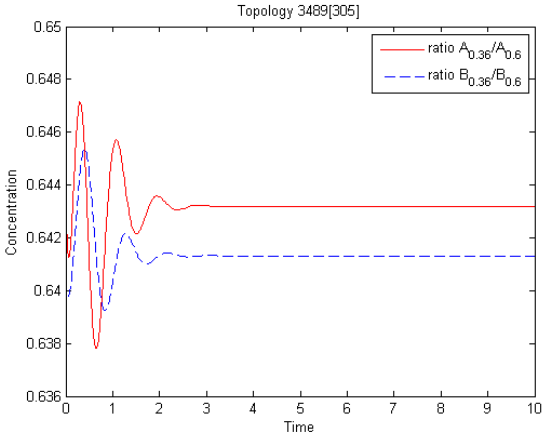



(b) Ouput from C Nonlinear Model

Figure 6.55: Topology 3489

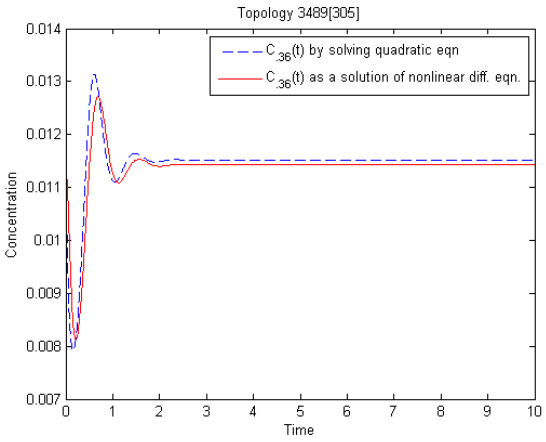



(a) A/B ratio

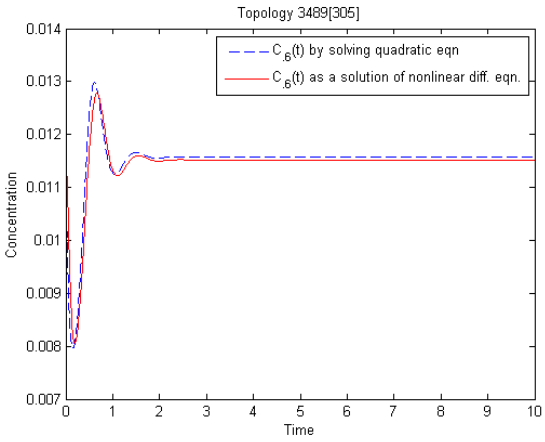



(b) AA and BB ratios

Figure 6.56: Topology 3489




(a) Quadratic approximation and Output of Nonlinear System



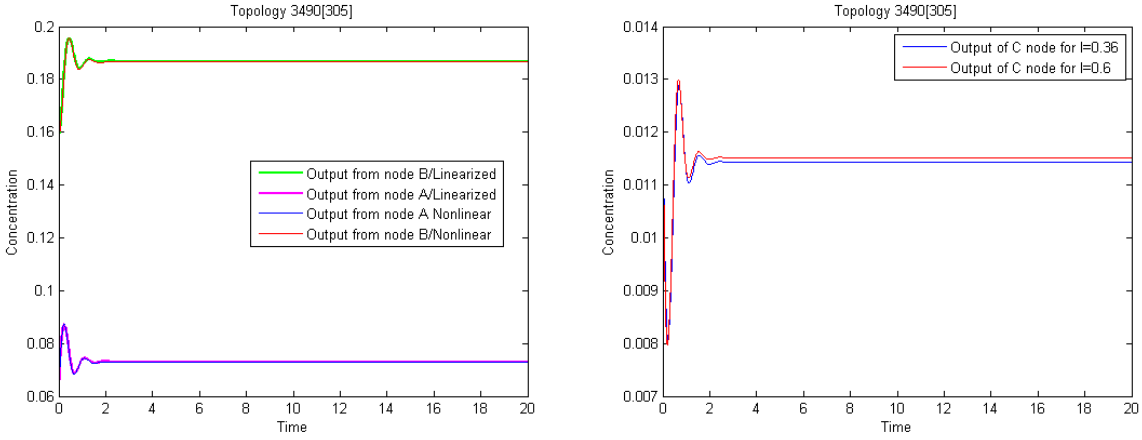

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.57: Topology 3489

Circuit 20.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} - k_{F_B B} F_B \frac{B}{B+K_{F_B B}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{BC} B \frac{1-C}{1-C+K_{BC}} - k_{AC} A \frac{C}{C+K_{AC}} - k_{CC} C \frac{C}{C+K_{CC}}
\end{aligned}
\tag{6.18}
$$

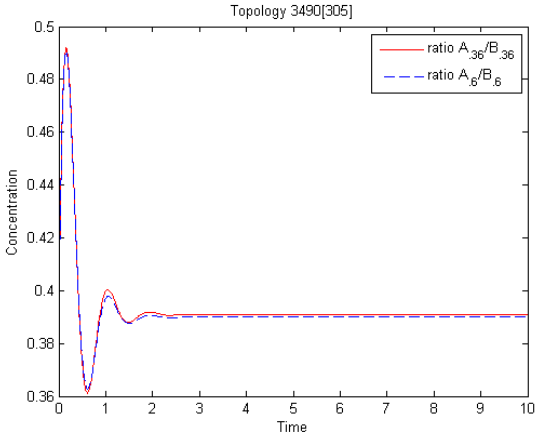
Where: $K_{IA} = 4.387832$; $k_{IA} = 19.638124$; $K_{BA} = 0.005461$; $k_{BA} = 7.103952$; $K_{AB} = 0.444375$; $k_{AB} = 12.053134$; $K_{F_B} = 1.716920$; $k_{F_B} = 11.601122$; $K_{BC} = 51.850148$; $k_{BC} = 80.408137$; $K_{AC} = 0.013988$; $k_{AC} = 8.521185$; $K_{CC} = 1.962230$; $k_{CC} = 17.382010$;



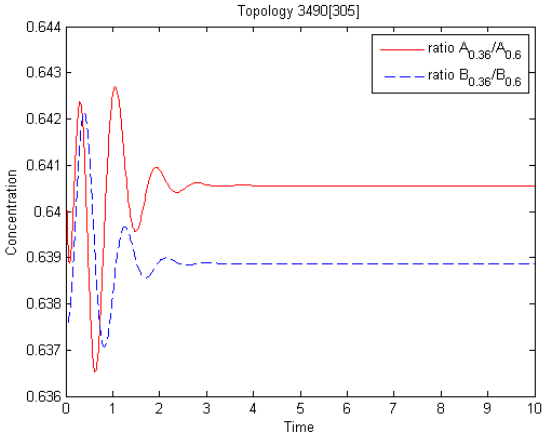

(a) Dynamics of A and B in Linearized model

(b) Ouput from C Nonlinear Model

Figure 6.58: Topology 3490

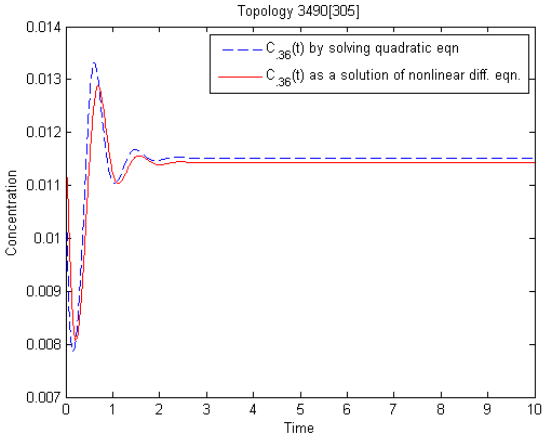



(a) A/B ratio

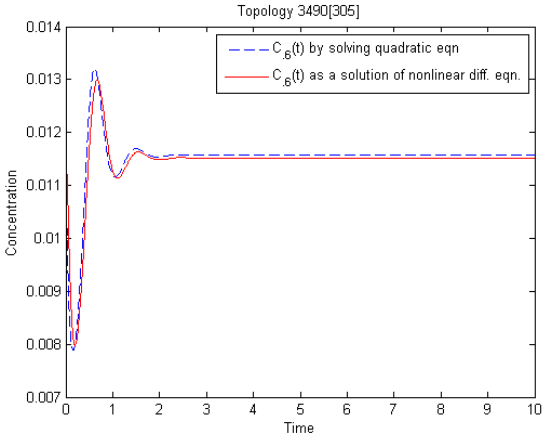



(b) AA and BB ratios

Figure 6.59: Topology 3490




(a) Quadratic approximation and Output of Nonlinear System



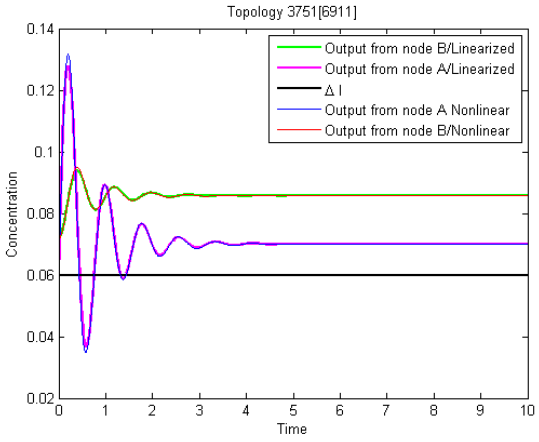

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.60: Topology 3490

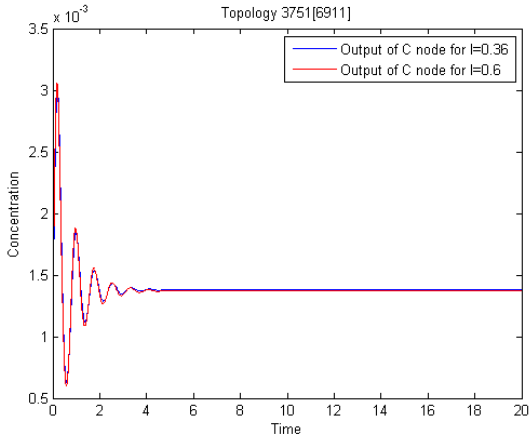
Circuit 21.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} + k_{CA} C \frac{1-A}{1-A+K_{CA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} - k_{F_B B} F_B \frac{B}{B+K_{F_B B}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{AC} A \frac{1-C}{1-C+K_{AC}} - k_{BC} B \frac{C}{C+K_{BC}} - k_{CC} C \frac{C}{C+K_{CC}}
\end{aligned}
\tag{6.19}
$$

Where: $K_{IA} = 0.093918$; $k_{IA} = 11.447219$; $K_{BA} = 0.001688$; $k_{BA} = 44.802268$; $K_{CA} = 5.026318$; $k_{CA} = 45.803641$; $K_{AB} = 0.001191$; $k_{AB} = 1.466561$; $K_{F_B} = 9.424319$; $k_{F_B} = 22.745736$; $K_{AC} = 0.113697$; $k_{AC} = 1.211993$; $K_{BC} = 0.009891$; $k_{BC} = 7.239357$; $K_{CC} = 0.189125$; $k_{CC} = 17.910182$;

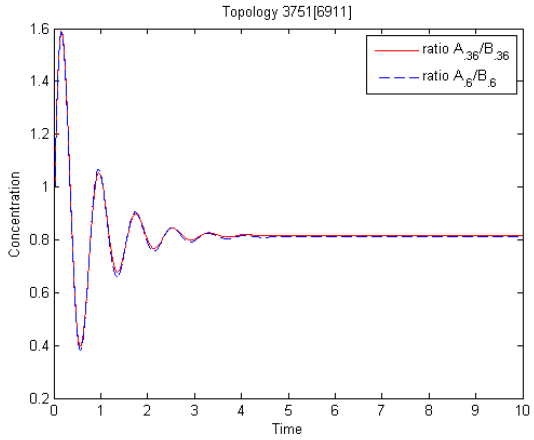



(a) Dynamics of A and B in Linearized model

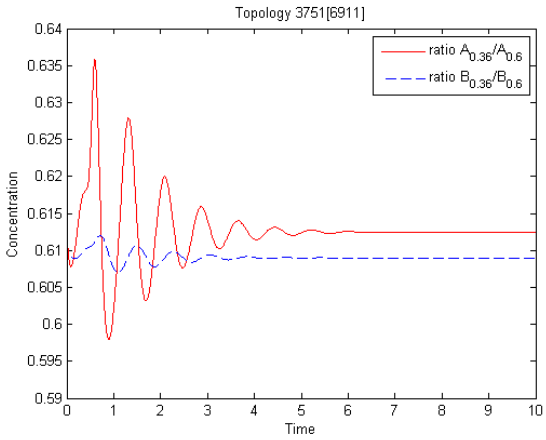



(b) Ouput from C Nonlinear Model

Figure 6.61: Topology 3751

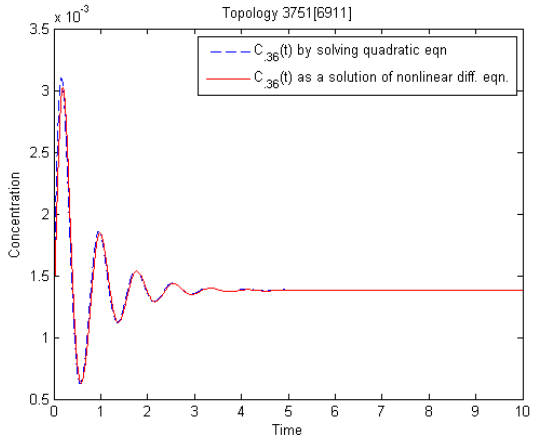



(a) A/B ratio

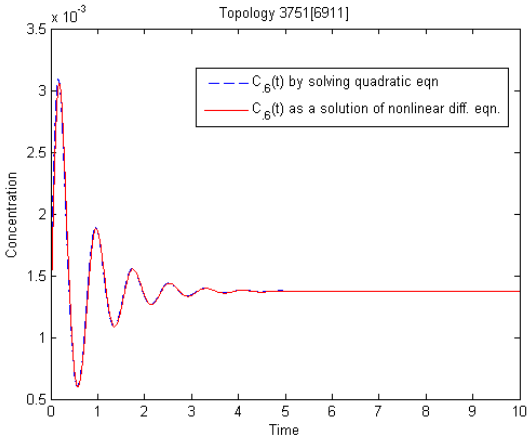



(b) AA and BB ratios

Figure 6.62: Topology 3751




(a) Quadratic approximation and Output of Nonlinear System

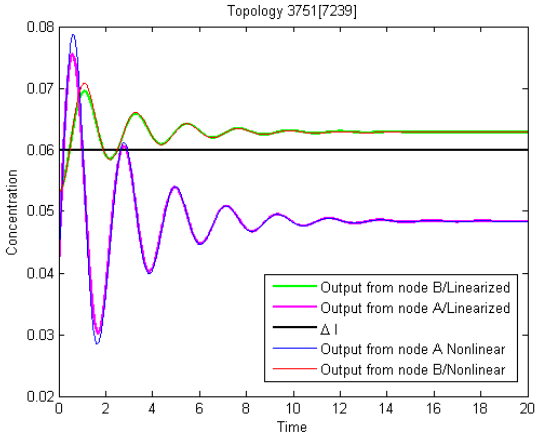



(b) Quadratic approximation and Output of Nonlinear System

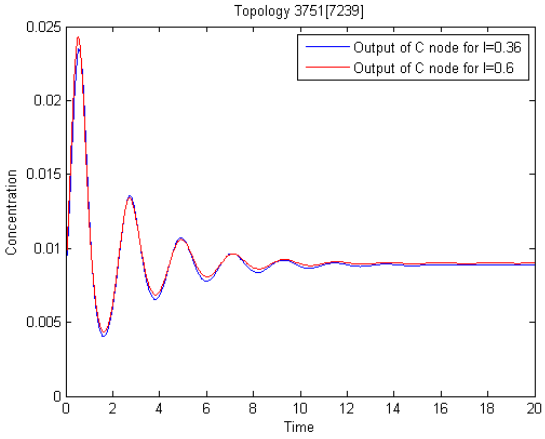
Figure 6.63: Topology 3715

Circuit 22.

This is the same topology as in previous case, only different parameter set was used: $K_{AB} = 1.620877$; $k_{AB} = 2.306216$; $K_{F_B} = 2.012565$; $k_{F_B} = 2.700847$; $K_{AC} = 0.010933$; $k_{AC} = 8.968091$; $K_{BA} = 0.001812$; $k_{BA} = 10.039221$; $K_{BC} = 0.014199$; $k_{BC} = 17.762333$; $K_{CA} = 0.002690$; $k_{CA} = 1.506954$; $K_{CC} = 2.686891$; $k_{CC} = 4.139044$; $K_{IA} = 0.161715$; $k_{IA} = 1.933303$;

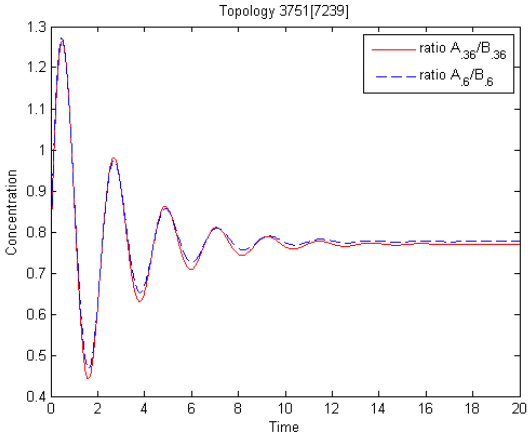



(a) Dynamics of A and B in Linearized model

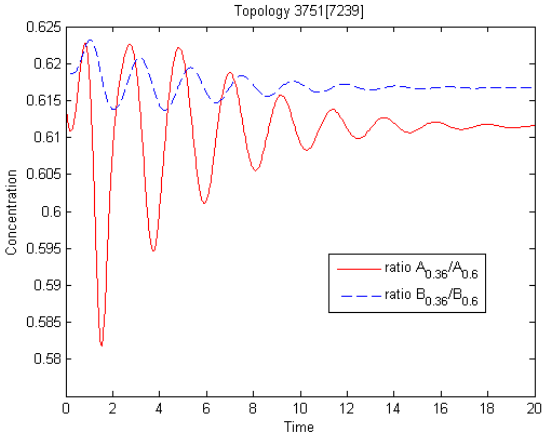



(b) Ouput from C Nonlinear Model

Figure 6.64: Topology 3751

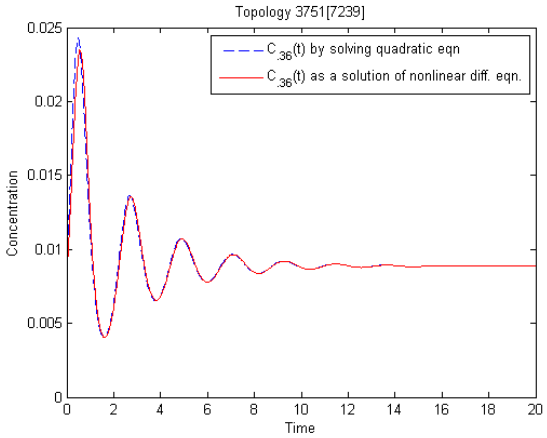



(a) A/B ratio

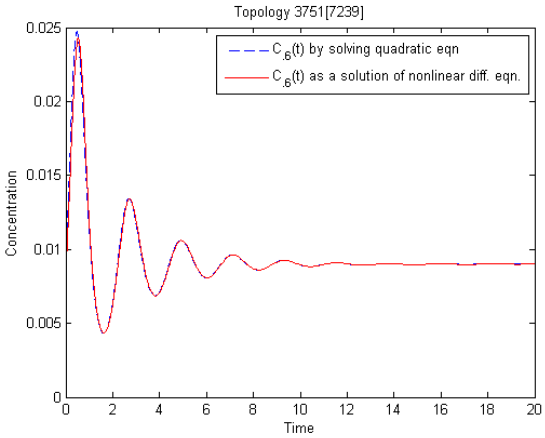



(b) AA and BB ratios

Figure 6.65: Topology 3751




(a) Quadratic approximation and Output of Nonlinear System



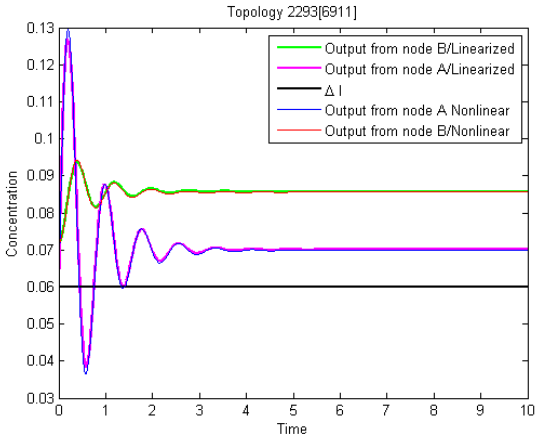

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.66: Topology 3751

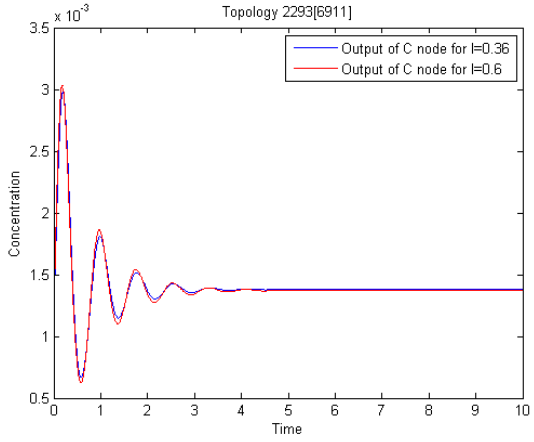
Circuit 23.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} - k_{CA} C \frac{A}{A+K_{CA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} - k_{F_B B} F_B \frac{B}{B+K_{F_B B}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{AC} A \frac{1-C}{1-C+K_{AC}} - k_{BC} B \frac{C}{C+K_{BC}} - k_{CC} C \frac{C}{C+K_{CC}}
\end{aligned}
\tag{6.20}
$$

Where: $K_{IA} = 0.093918$; $k_{IA} = 11.447219$; $K_{BA} = 0.001688$; $k_{BA} = 44.802268$; $K_{CA} = 90.209027$; $k_{CA} = 96.671843$; $K_{AB} = 0.001191$; $k_{AB} = 1.466561$; $K_{F_B} = 9.424319$; $k_{F_B} = 22.745736$; $K_{AC} = 0.113697$; $k_{AC} = 1.211993$; $K_{BC} = 0.009891$; $k_{BC} = 7.239357$; $K_{CC} = 0.189125$; $k_{CC} = 17.910182$;

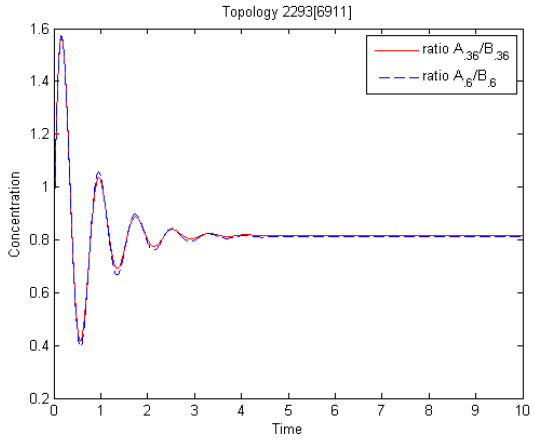



(a) Dynamics of A and B in Linearized model

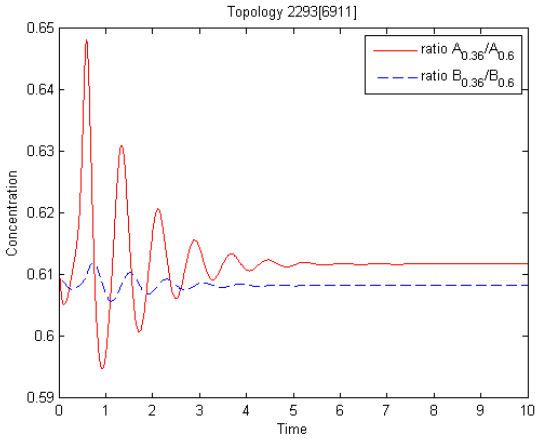



(b) Ouput from C Nonlinear Model

Figure 6.67: Topology 2293

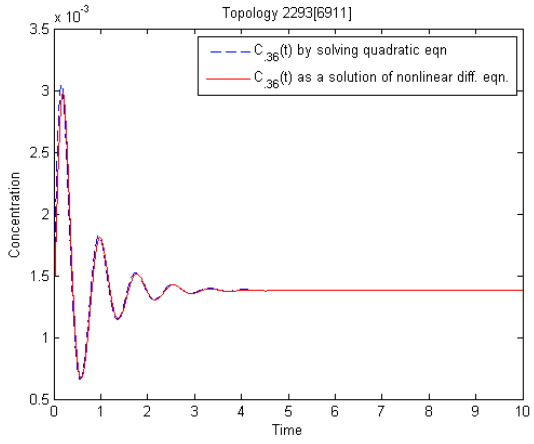



(a) A/B ratio

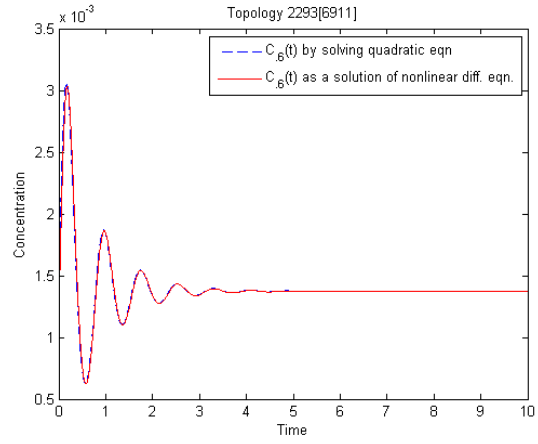



(b) AA and BB ratios

Figure 6.68: Topology 2293




(a) Quadratic approximation and Output of Nonlinear System



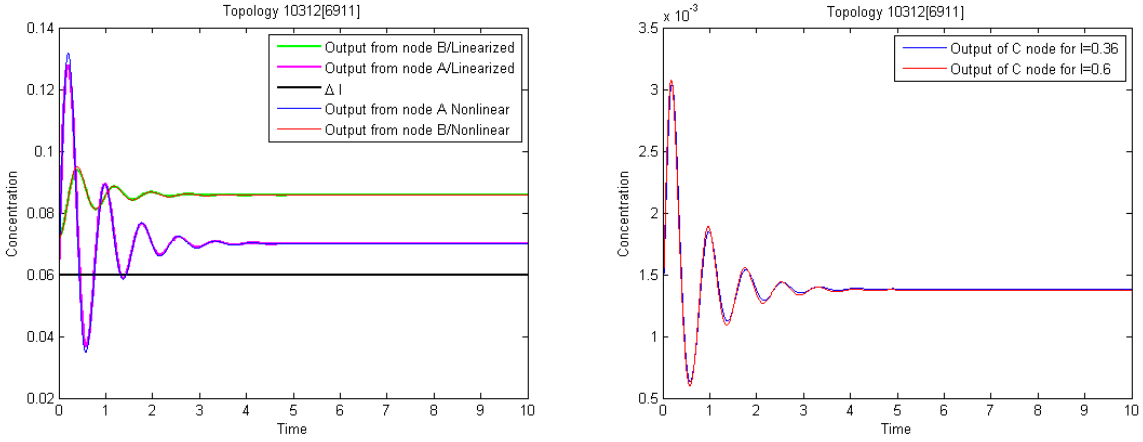

(b) Quadratic approximation and Output of Nonlinear System

Figure 6.69: Topology 2293

Circuit 24.

$$
\begin{aligned}
f_A = \quad & \frac{dA(t)}{dt} = k_{IA} I \frac{1-A}{1-A+K_{IA}} - k_{BA} B \frac{A}{A+K_{BA}} + k_{CA} C \frac{1-A}{1-A+K_{CA}} \\
f_B = \quad & \frac{dB(t)}{dt} = k_{AB} A \frac{1-B}{1-B+K_{AB}} - k_{F_B B} F_B \frac{B}{B+K_{F_B B}} \\
f_C = \quad & \frac{dC(t)}{dt} = k_{AC} A \frac{1-C}{1-C+K_{AC}} - k_{BC} B \frac{C}{C+K_{BC}}
\end{aligned}
\tag{6.21}
$$

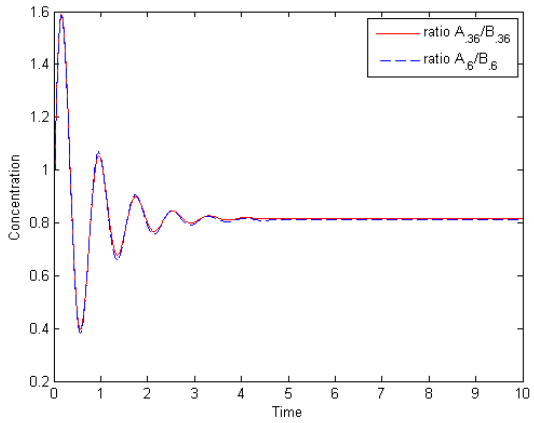
Where: $K_{IA} = 0.093918$; $k_{IA} = 11.447219$; $K_{BA} = 0.001688$; $k_{BA} = 44.802268$; $K_{CA} = 5.026318$; $k_{CA} = 45.803641$; $K_{Ab} = 0.001191$; $k_{AB} = 1.466561$; $K_{F_B} = 9.424319$; $k_{F_B} = 22.745736$; $K_{AC} = 0.113697$; $k_{AC} = 1.211993$; $K_{BC} = 0.009891$; $k_{BC} = 7.239357$;



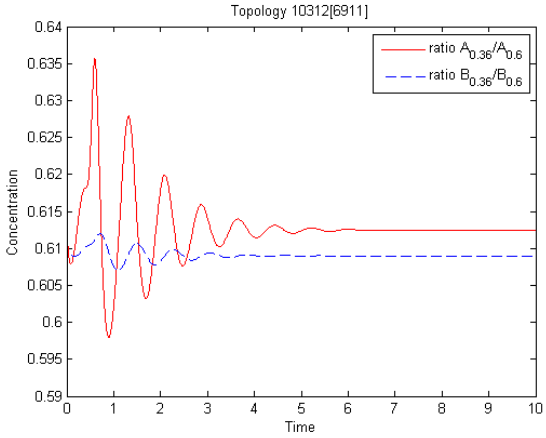

(a) Dynamics of A and B in Linearized model

(b) Ouput from C Nonlinear Model

Figure 6.70: Topology 10312

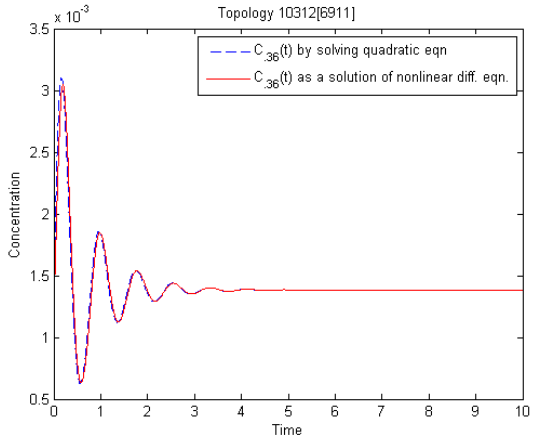



(a) A/B ratio

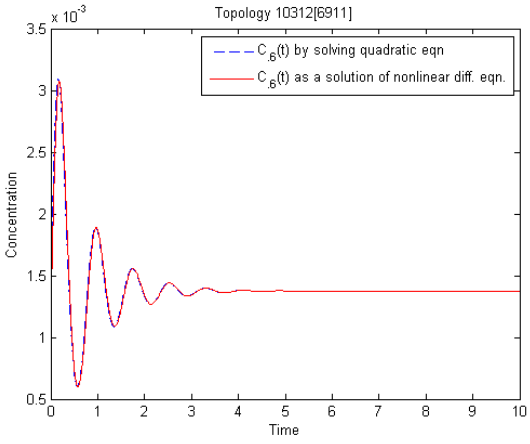



(b) AA and BB ratios

Figure 6.71: Topology 10312




(a) Quadratic approximation and Output of Nonlinear System

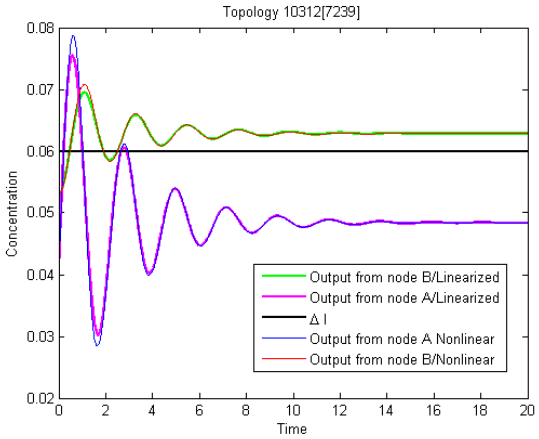



(b) Quadratic approximation and Output of Nonlinear System

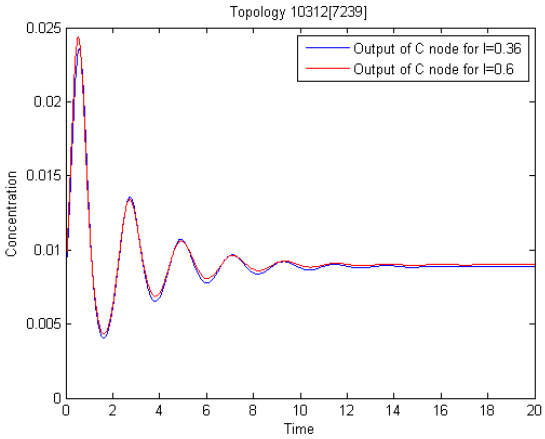
Figure 6.72: Topology 10312

Circuit 25.

This is the same topology as in previous case, only different parameter set was used: $K_{AB} = 1.620877$; $k_{AB} = 2.306216$; $K_{F_B} = 2.012565$; $k_{F_B} = 2.700847$; $K_{AC} = 0.010933$; $k_{AC} = 8.968091$; $K_{BA} = 0.001812$; $k_{BA} = 10.039221$; $K_{BC} = 0.014199$; $k_{BC} = 17.762333$; $K_{CA} = 0.002690$; $k_{CA} = 1.506954$; $K_{IA} = 0.161715$; $k_{IA} = 1.933303$;

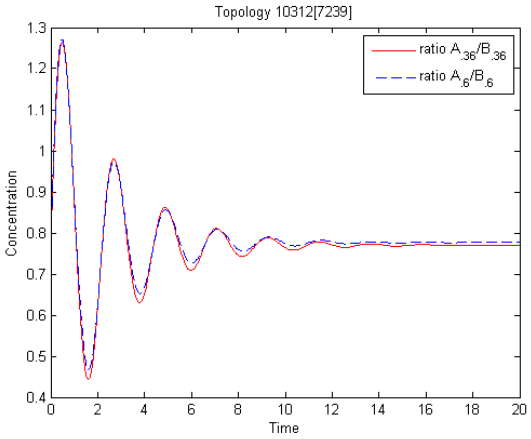



(a) Dynamics of A and B in Linearized model

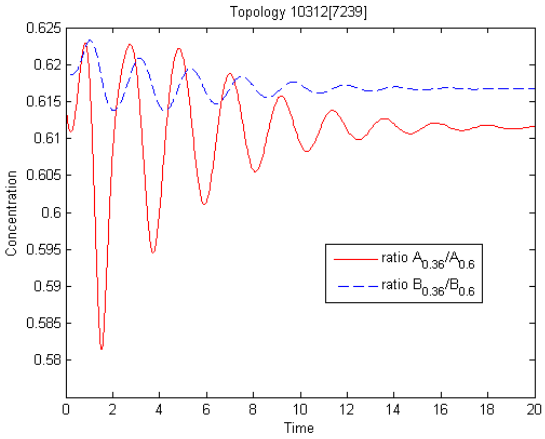



(b) Ouput from C Nonlinear Model

Figure 6.73: Topology 10312

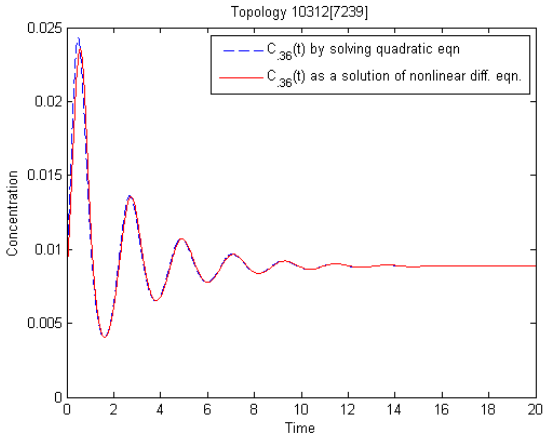



(a) A/B ratio

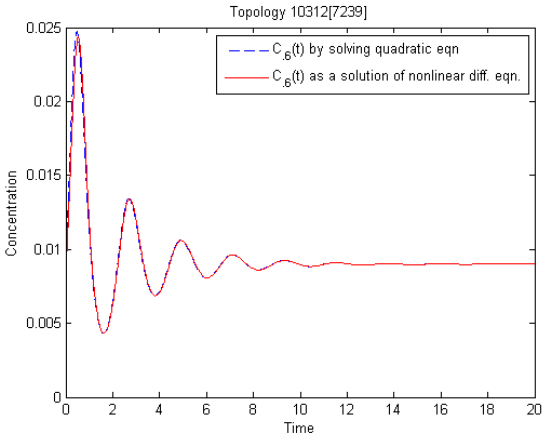



(b) AA and BB ratios

Figure 6.74: Topology 10312




(a) Quadratic approximation and Output of Nonlinear System




(b) Quadratic approximation and Output of Nonlinear System

Figure 6.75: Topology 10312

## 6.2 Topologies that exhibit Time Shifting Fold Change Detection

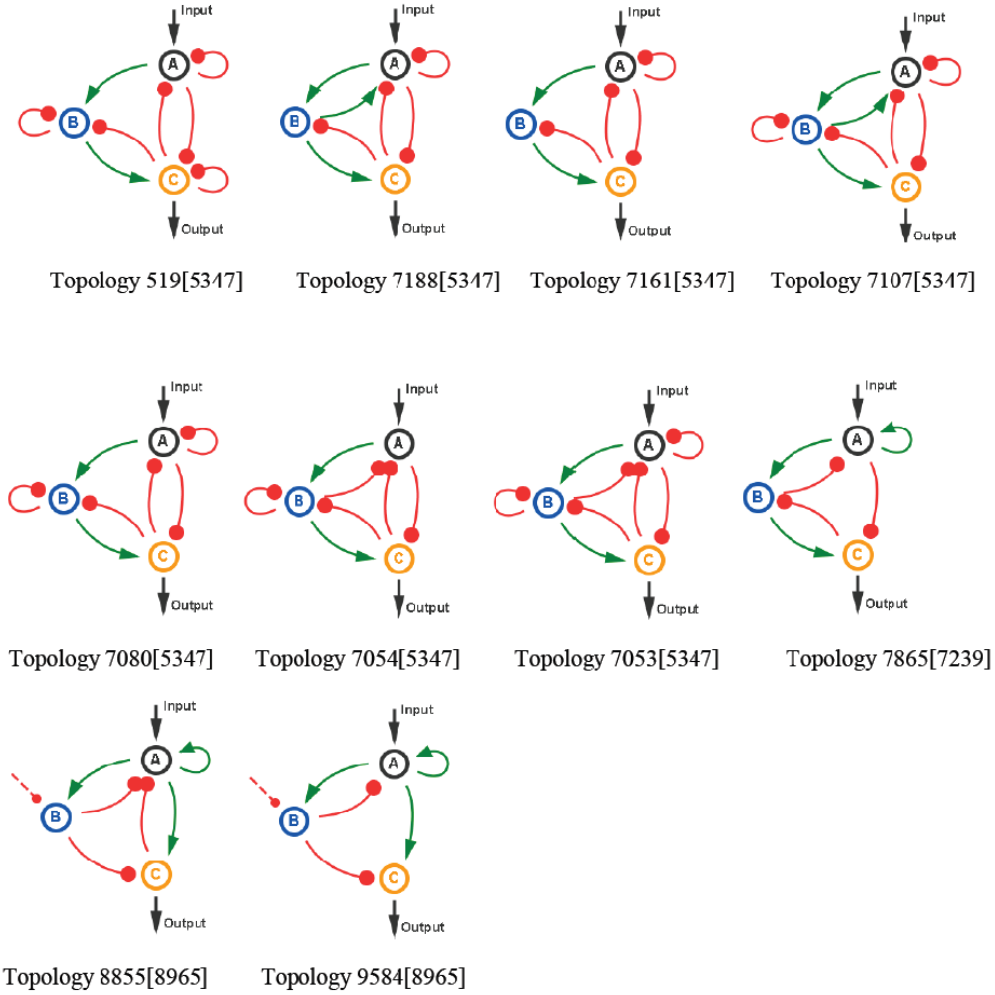



Figure 6.76: Time Shifting Fold Change Detection

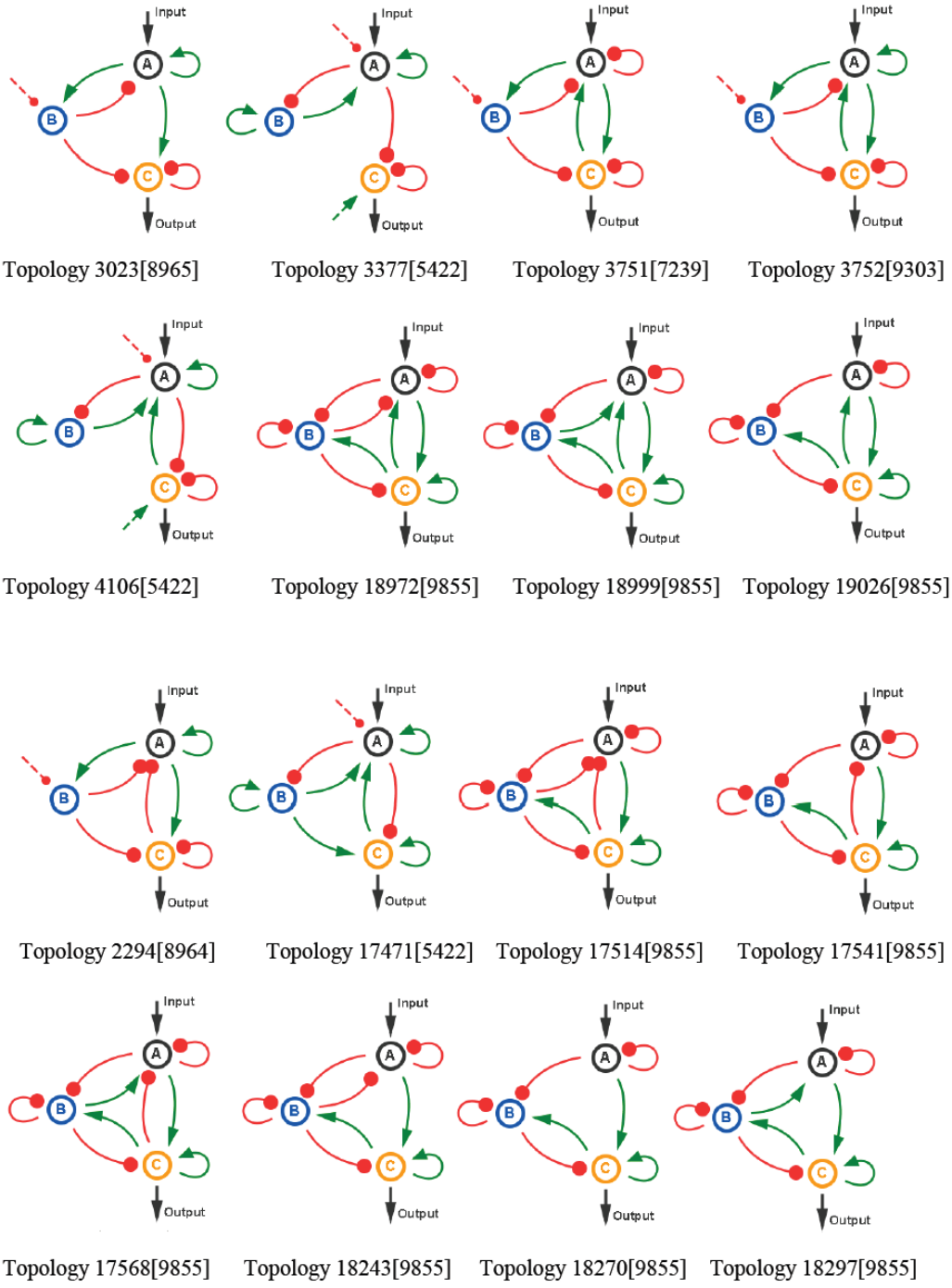



Figure 6.77: Time Shifting Fold Change Detection

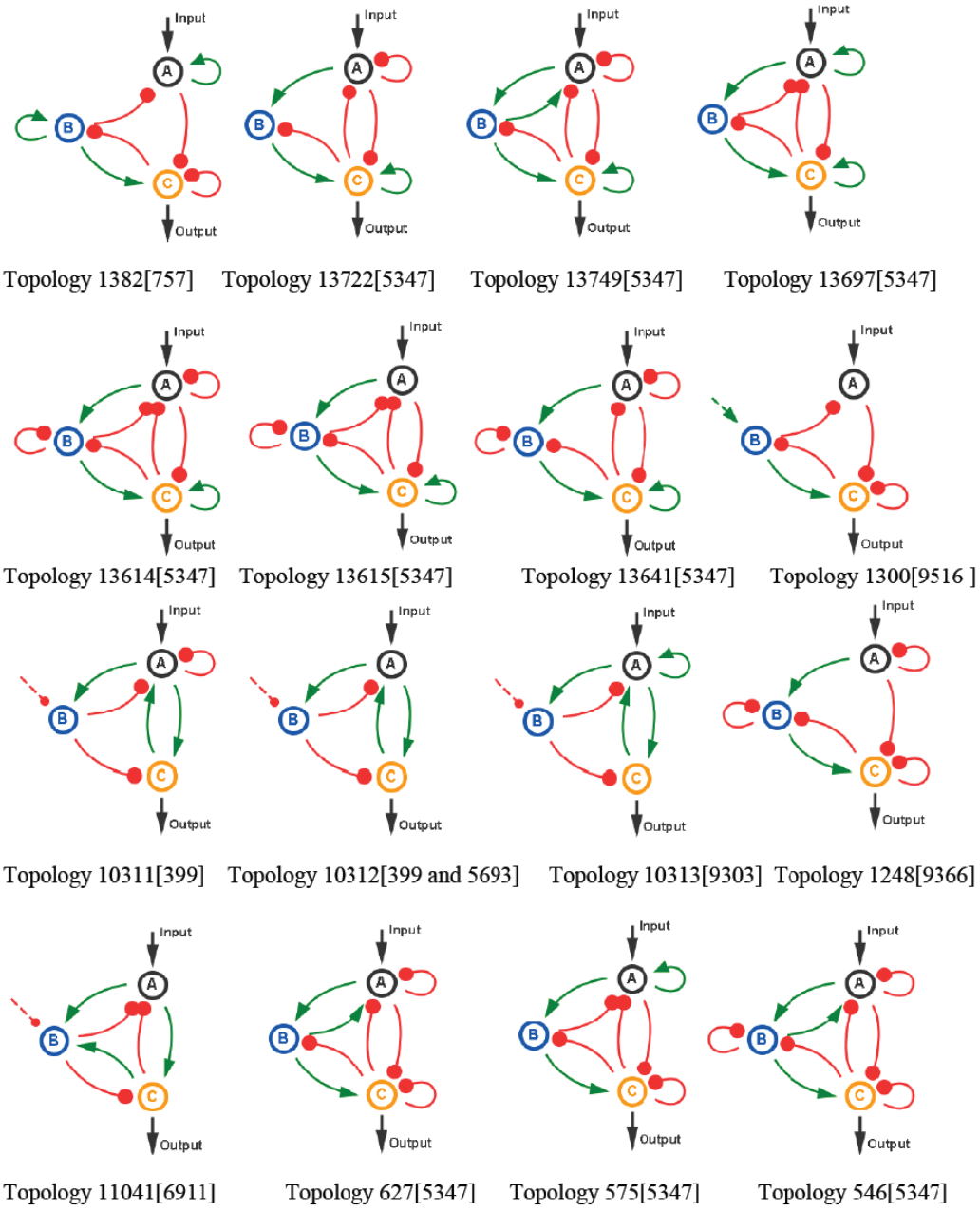



Figure 6.78: Time Shifting Fold Change Detection

## 6.3 Dynamics of C node for Time Shifting Fold Change Detection Circuits

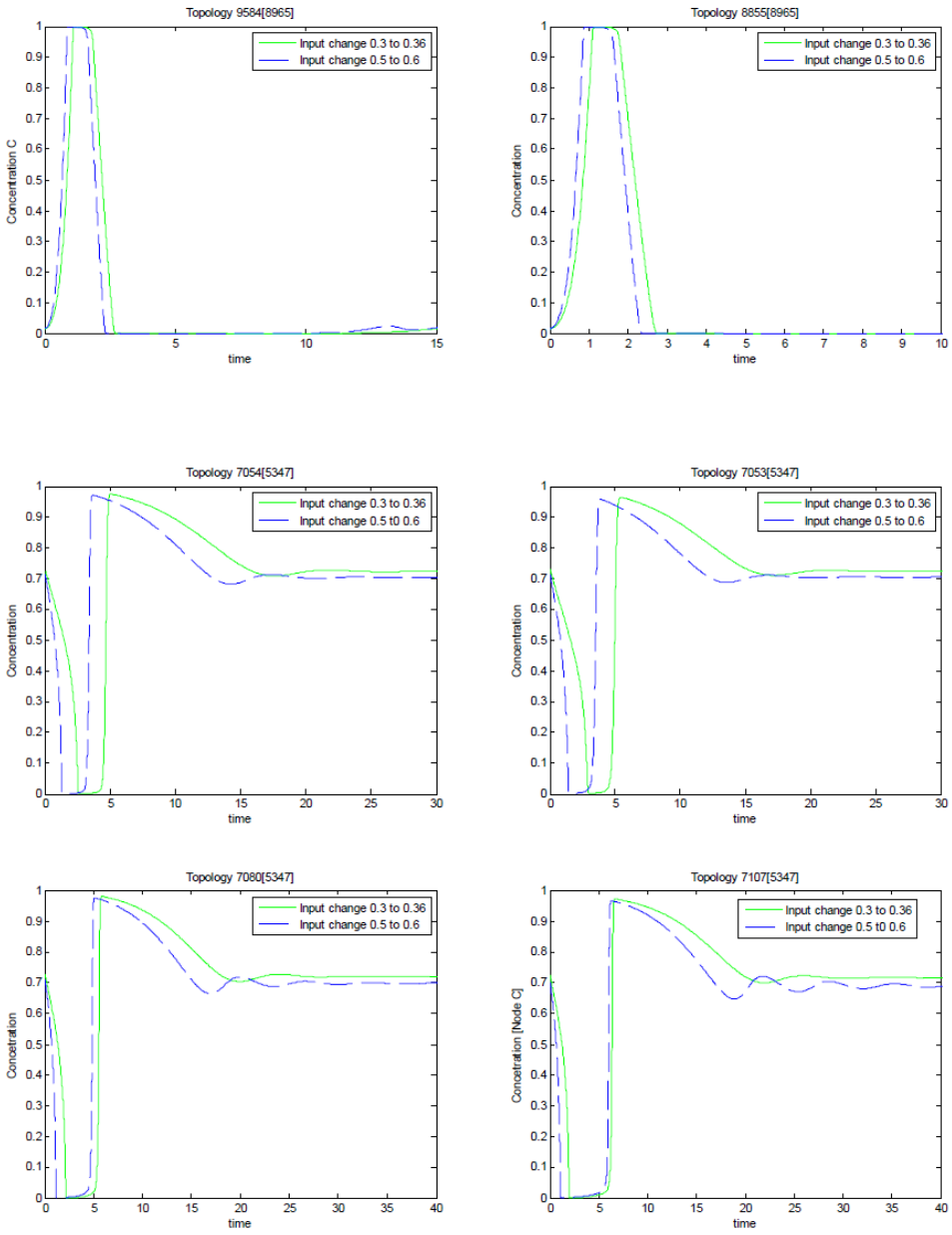



Figure 6.79: Time Shifting Fold Change Detection: Dynamics of Node C

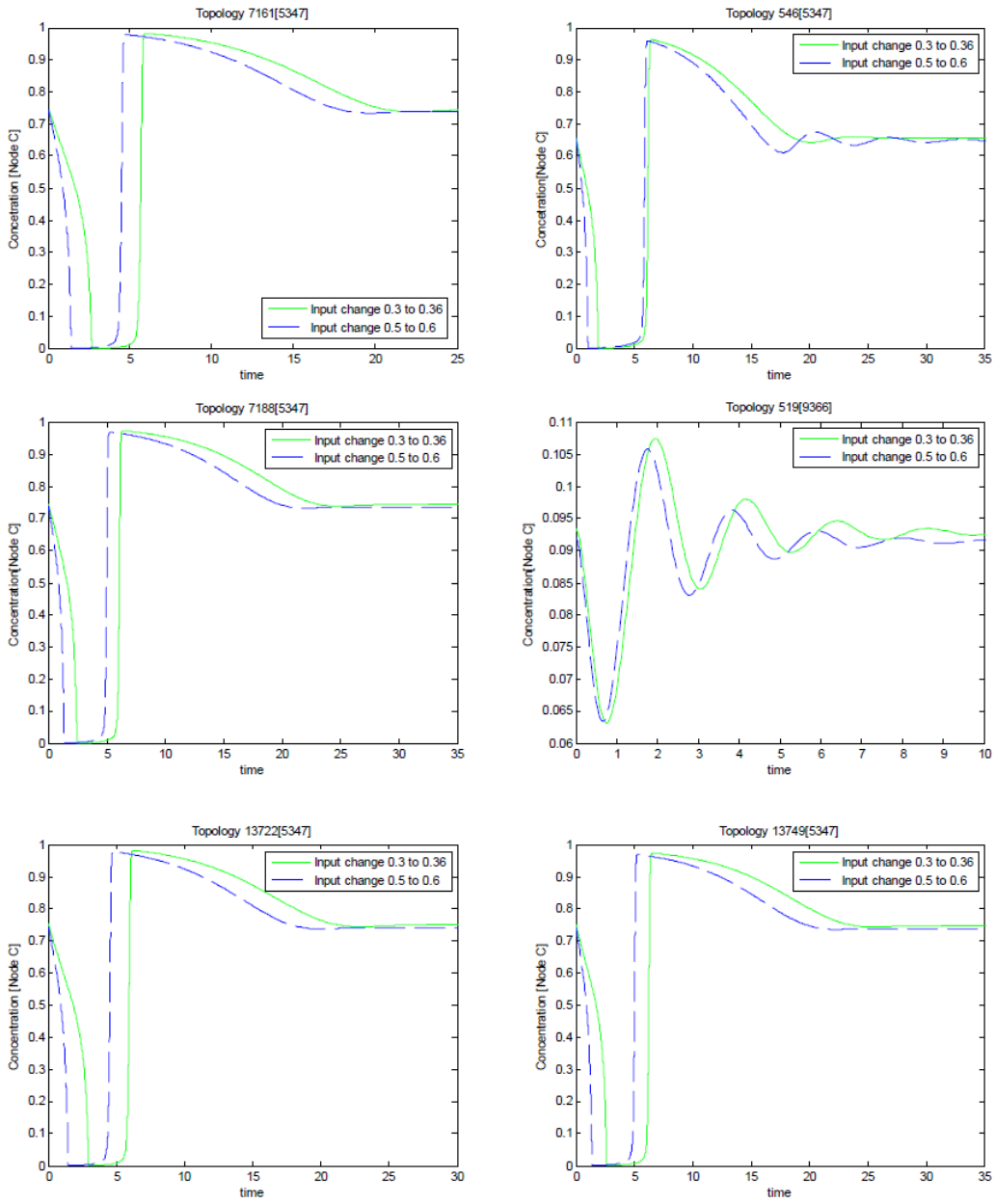



Figure 6.80: Time Shifting Fold Change Detection: Dynamics of Node C

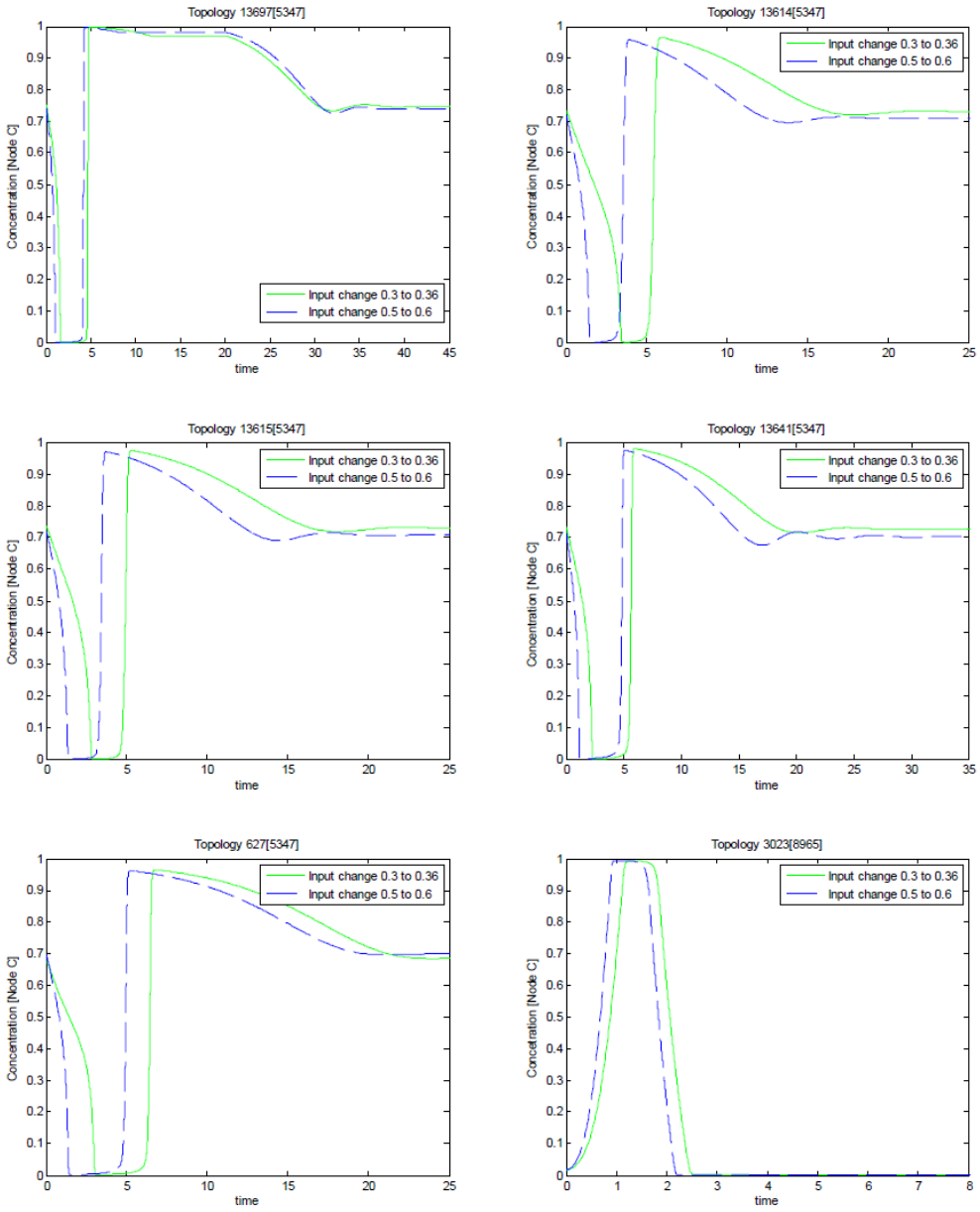



Figure 6.81: Time Shifting Fold Change Detection: Dynamics of Node C

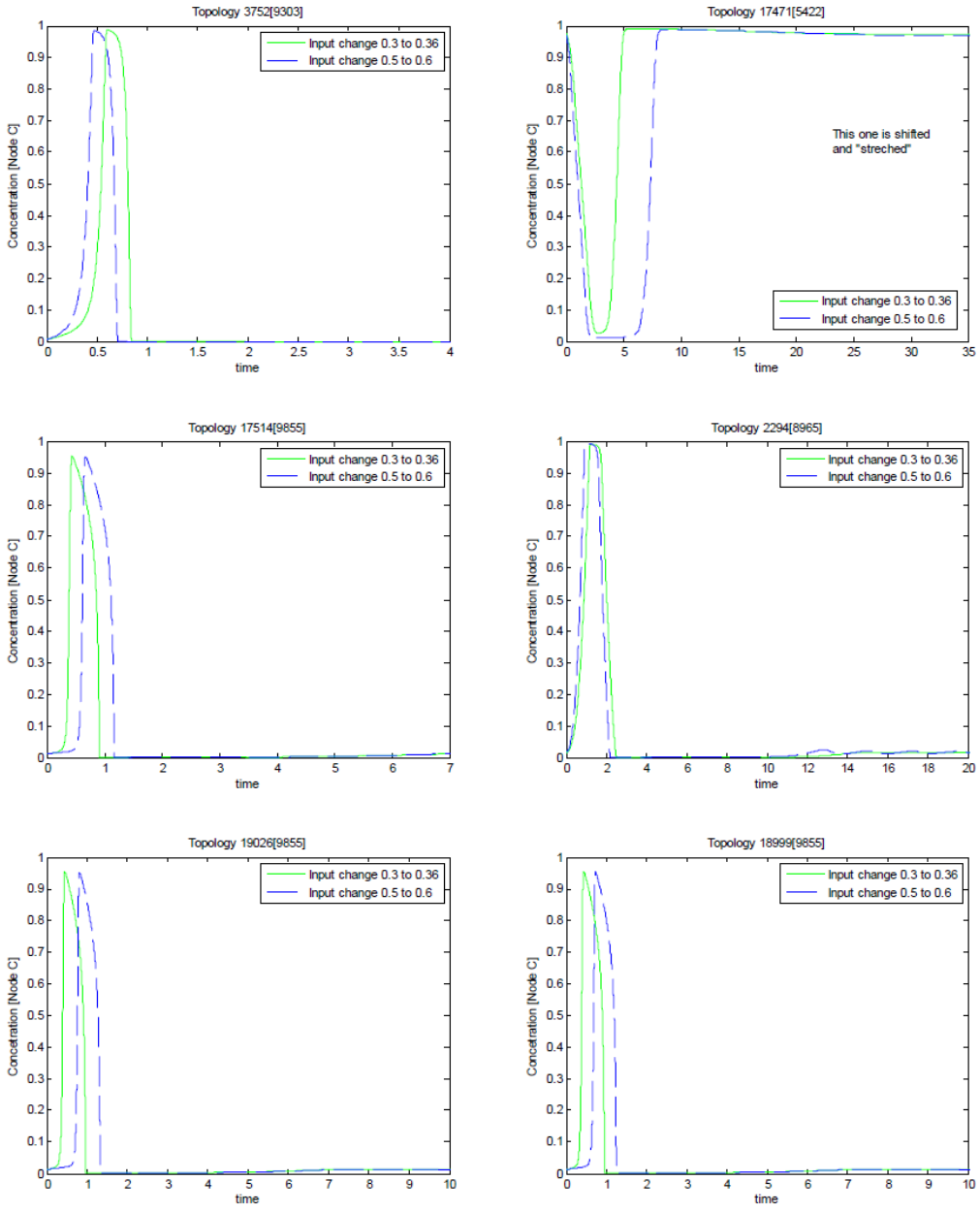



Figure 6.82: Time Shifting Fold Change Detection: Dynamics of Node C

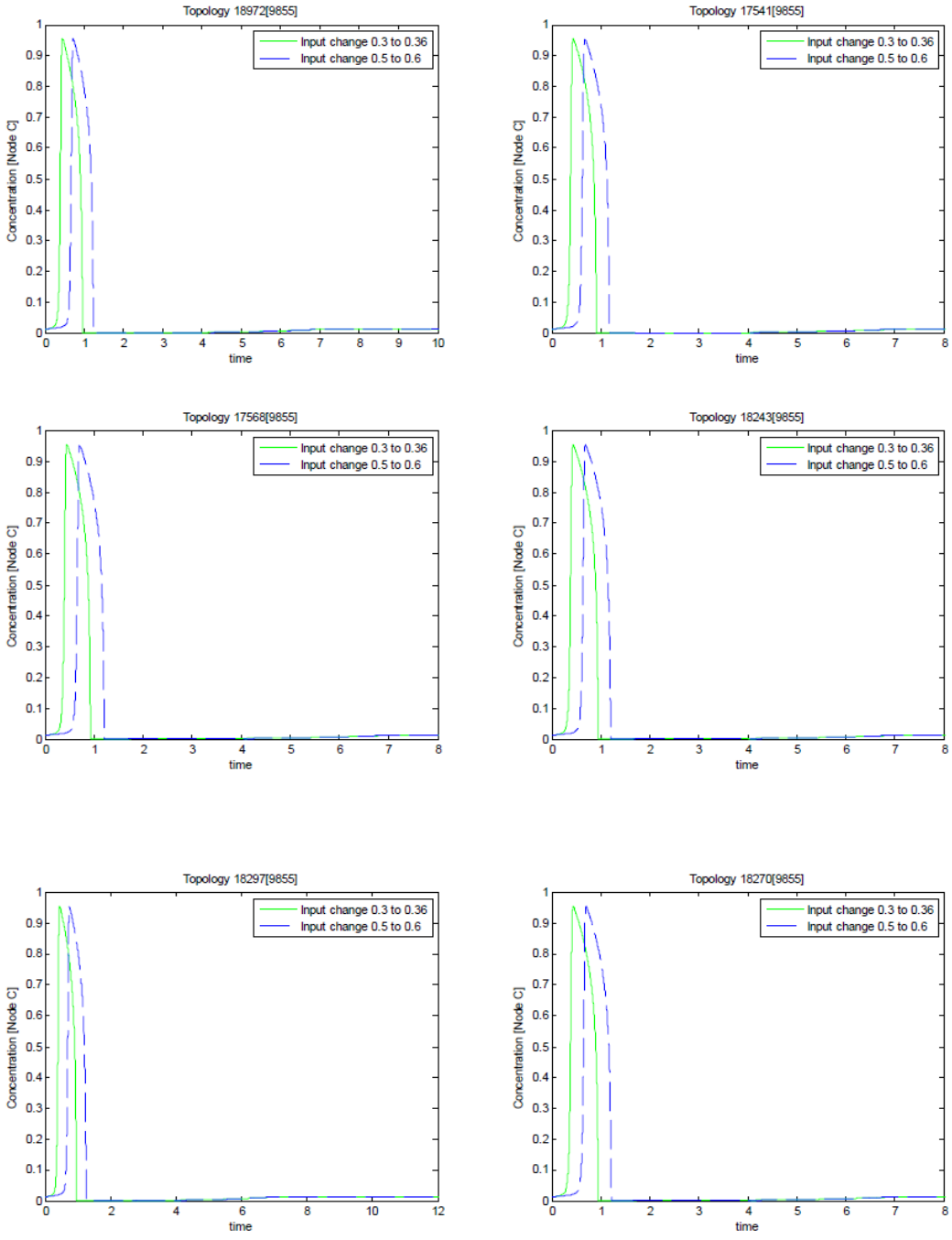



Figure 6.83: Time Shifting Fold Change Detection: Dynamics of Node C

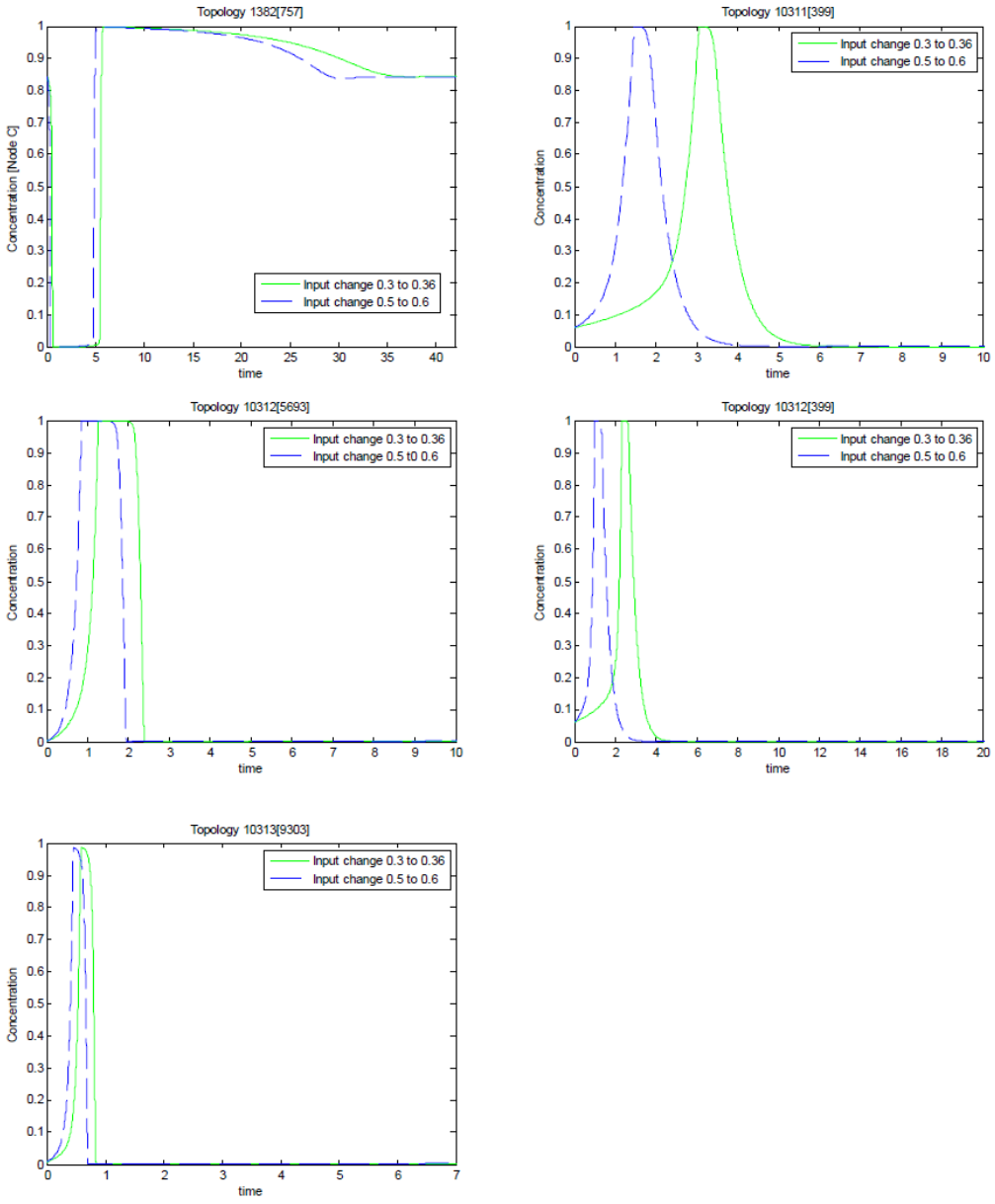



Figure 6.84: Time Shifting Fold Change Detection: Dynamics of Node C

# References

Alon, U., Surette, M.G., Barkai, N., and Leibler, S. (1999). Robustness in bacterial chemotaxis. *Nature*, 397(6715), 168–171.

Alon, U. (2007a). *An Introduction to Systems Biology: Design Principles of Biological Circuits*. Chapman & Hall/CRC Mathematical & Computational Biology, Boca Raton, FL, USA.

Alon, U. (2007b). Network motifs: theory and experimental approaches. *Nature Reviews Genetics*, 8(6), 450–461.

Barkai, N. and Leibler, S. (1997). Robustness in simple biochemical networks. *Nature*, 387(6636), 913–917.

Ferrell, J.E. (2009). Signaling motifs and weber's law. *Molecular cell*, 36(5), 724–727.

Garrett, R.H. and Grisham, C.M. (1999). *Biochemistry*. Philadelphia: Saunders College Pub.

Goentoro, L., Shoval, O., Kirschner, M.W., and Alon, U. (2009). The incoherent feedforward loop can provide fold-change detection in gene regulation. *Molecular Cell*, 36(5), 894–899.

Goldbeter, A. and Jr, D.E.K. (1984). Ultrasensitivity in biochemical systems controlled by covalent modification. interplay between zero-order and multistep effects. *The Journal of biological chemistry*, 259(23), 14441–14447.

Iman, R.L. (2001). Appendix A : Latin Hypercube Sampling 1. *Encyclopedia of Statistical Sciences,Update*, 3(September), 408–411.

Kim, D., Kwon, Y.K., and Cho, K.H. (2008). The biphasic behavior of incoherent feedforward loops in biomolecular regulatory networks. *BioEssays*, 30(11-12), 1204–1211.

Ma, W., Trusina, A., El-Samad, H., Lim, W.A., and Tang, C. (2009). Defining network topologies that can achieve biochemical adaptation. *Cell*, 138(4), 760–773.

Mangan, S. and Alon, U. (2003). Structure and function of the feed-forward loop network motif. *Proceedings of the National Academy of Sciences*, 100(21), 11980–11985.

Milo, R., Itzkovitz, S., Kashtan, N., Levitt, R., and Alon, U. (2004). Network motifs: Simple building blocks of complex networks. *Science*, 305(5687), 1107–1107.

Savageau, M.A. (1971). Concepts relating the behavior of biochemical systems to their underlying molecular properties. *Archives of Biochemistry and Biophysics*, 145(2), 612 – 621.

Shimizu, T.S., Tu, Y., and Berg, H.C. (2010). A modular gradient-sensing network for chemotaxis in escherichia coli revealed by responses to time-varying stimuli. *Mol Syst Biol*, 6, 382.

Shoval, O., Alon, U., and Sontag, E. (2010a). Symmetry invariance for adapting biological systems. *CoRR*, abs/1012.2782.

Shoval, O., Goentoro, L., Hart, Y., Mayo, A., Sontag, E., and Alon, U. (2010b). Fold-change detection and scalar symmetry of sensory input fields. *Proceedings of the National Academy of Sciences*, 107(36), 15995–16000.

Sontag, E.D. (2008). Remarks on Feedforward Circuits, Adaptation, and Pulse Memory. *ArXiv e-prints*.

Tyson, J.J., Chen, K.C., and Novak, B. (2003). Sniffers, buzzers, toggles and blinkers: dynamics of regulatory and signaling pathways in the cell. *Current Opinion in Cell Biology*, 15(2), 221 – 231.

Tyson, J.J. and Novak, B. (2010). Functional motifs in biochemical reaction networks. *Annual Review of Physical Chemistry*, 61(1), 219–240. doi:10.1146/annurev.physchem.012809.103457.

Wyss, G.D. and Jorgensen, K.H. (1998). A users guide to LHS: Sandias Latin hypercube sampling software. *SAND980210 Sandia National Laboratories Albuquerque NM*, (February).

Yi, T.M., Huang, Y., Simon, M.I., and Doyle, J. (2000). Robust perfect adaptation in bacterial chemotaxis through integral feedback control. *Proceedings of the National Academy of Sciences of the United States of America*, 97(9), 4649–4653.

# Vita

## Maja Skataric

**2009-2011** M. Sc. in Electrical and Computer Engineering, Rutgers University

**2005-2009** B. Sc. in Electrical Engineering, University of Belgrade, Serbia