

# **Inference of Gene Regulatory Network with S-system and Artificial Bee Colony Algorithm**

by

Obayed Bin Mahfuz

12101103

A thesis submitted in partial fulfillment of the requirements for the degree of  
Bachelor of Science in Computer Science and Engineering



Inspiring Excellence

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING  
BRAC UNIVERSITY

August 2015

“I hereby declare that the work in this thesis is my own work except for quotations and summaries which have been duly acknowledged.”

Signature: \_\_\_\_\_

Date : \_\_\_\_\_

Name : Obayed Bin Mahfuz

ID : 12101103

Supervised by: \_\_\_\_\_

: Dilruba Showkat

## ***Abstract***

Gene Regulatory Networks are the basic functional unit in living organisms. Gene Regulatory Network mainly refers to the behavior of thousands of genes (inside the chromosome of cell) with other genes. Each gene has expression levels and interaction with other genes. These interactions and expression levels can be calculated from their amount and time duration of protein production. Due to the invention of DNA microarray in biotechnology, we are able to find gene expression levels from real genetic regulatory networks. Now it's the time to find a reverse process to reach a satisfactory result that matches with those data derived from DNA microarray. If we can find those values that satisfy the result, then we can predict our gene behavior much early. Watching abnormal gene behavior, diseases can be found. Thus it can be a revolutionary step towards medicine and diagnosis sector. If we reach better accuracy then it will also help us to develop tissue and organs. That means, for chronic disease or any other problem if one's heart cannot pump blood, then he can repair his heart by making a new heart developed from the muscle cells from any other organ of his body. Biological systems are very much complex in nature. And S-system model is a recent and popular class to model biological systems. Hence, I am using S-system class for modeling. Gene Regulatory Networks contain a large number of genes. And artificial bee colony is best suited for population based problems. Hence, I am proposing an inference algorithm of gene regulatory network on the framework of S-system class of ordinary differential equations (ODEs) and artificial bee colony algorithm.

**Keywords:** Gene Regulatory Network, Inference, S-system, DNA microarray, Artificial Bee Colony

## **Acknowledgement**

I would like to express my sincere gratitude to my supervisor, Dilruba Showkat for her support, guidance and invaluable instructions which helped me as well as encouraged me to complete the thesis.

# Contents

<b>Declaration of Authorship.....</b>	<b>i</b>
<b>Abstract.....</b>	<b>ii</b>
<b>Acknowledgements.....</b>	<b>iii</b>
<b>List of Figures.....</b>	<b>iv</b>
<b>1 Introduction .....</b>	<b>1-4</b>
1.1 Motivation.....	2
1.2 Aims.....	3
1.3 Scope.....	3
1.4 Organization of the thesis.....	4
<b>2 Related Biological Study.....</b>	<b>5-10</b>
2.1 Gene.....	5
2.2 Functional Structure of Gene.....	6
2.3 Gene Regulation.....	6
2.4 DNA.....	7
2.5 RNA.....	7
2.6 Protein.....	8
2.7 Gene regulatory Network.....	8
2.8 Structure of Gene Regulatory network .....	8
2.9 DNA Microarray.....	9
2.10 Gene Expression Profiling.....	10
<b>3 Literature Review.....</b>	<b>11-15</b>
3.1 Modeling techniques.....	11
3.1.1 S-system Model.....	12
3.1.2 Boolean Network.....	12
3.1.3 Bayesian Network.....	12
3.1.4 Linear Time Variant.....	13
3.2 Searching Algorithm.....	14
3.2.1 Differential Algorithm.....	14

3.2.2	Self Adaptive Differential Evolution.....	14
3.2.3	Particle Swarm Optimization.....	14
3.3	Summary.....	15
<b>4</b>	<b>Proposed Approach.....</b>	<b>16-22</b>
4.1	S-system model .....	16
4.2	Mutation.....	17
4.3	Crossover.....	17
4.4	Artificial Bee Colony Algorithm.....	19
4.5	Proposed Algorithm.....	21
4.6	Fitness Evaluation Criteria.....	22
<b>5</b>	<b>Experimental Results.....</b>	<b>23-32</b>
5.1	Experimental Setup.....	23
5.2	Noise free data.....	24
5.3	5% Noisy data.....	25
5.4	10% Noisy data.....	26
5.5	15% Noisy data.....	28
5.6	Real data.....	29
5.7	Network structure of real data .....	30
5.8	Summary.....	32
<b>6</b>	<b>Application.....</b>	<b>33-34</b>
6.1	Identifying Diseases early.....	33
6.2	Organ development.....	33
6.3	Side effect less medicine production.....	34
6.4	Creating nature friendly organisms .....	34
<b>7</b>	<b>Conclusion and Future Work.....</b>	<b>35-36</b>
6.1	Conclusion.....	35
6.2	Future Scope.....	35

## References

## List of Figures

Figure 2.1: Chromosome inside a cell .....	6
Figure 2.2: Gene regulation in living cell.....	7
Figure 2.3: Gene Regulatory Network added with all information.....	9
Figure 2.4: An artificial Genetic Regulatory Network.....	9
Figure 2.5: DNA microarray image with an automated system background.....	9
Figure 4.1: Reconstruction procedure of a network using S-system.....	17
Figure 4.2: Mutation inside DNA.....	18
Figure 4.3: Crossover between parents.....	19
Figure 5.1: Estimated vs target noiseless data graph.....	25
Figure 5.2: Estimated vs target 5% noisy data graph.....	26
Figure 5.3: Estimated vs target 10% noisy data graph.....	27
Figure 5.4: Estimated vs target 15% noisy data graph.....	28
Figure 5.5: Estimated vs target real data graph.....	30
Figure 5.6 Network Structure of real data (Production).....	31
Figure 5.7 Network Structure of real data (Regression).....	32

## Chapter 1

### Introduction

A gene is a hereditary unit in all living organisms. It controls how living organisms inherit features from their ancestors. A gene is encoded in long strands of DNA. Genes provide necessary information to make proteins in a cell. Cells are the smallest individual parts of an organism. A cell is like a factory that is continuously working inside a living organism. The labor division inside a cell is as following - gene gives instruction and protein executes them. So, genes actually tell cells, what to do by giving instructions on protein creation. Gene expression is a process of producing a biologically functional molecule or gene product. Gene product can either be a protein or RNA. A gene regulatory network is a collection of genes inside a cell which interacts with each other. The interactions between these genes inside a cell can be obtained from a gene regulatory network. DNA microarray technology helps us to read expression levels of genes. So for the betterment of medicine, diagnosis and other few sectors, these microarray data needs to be analyzed.

Thus a reverse engineering process is required to determine the exact gene expression based on different factors. This process of reverse engineering is called inference. Inferring gene regulatory network is currently a topic under heavy research in the field of systems and computational biology. DNA microarray technology provides the way to measure the expression level of thousand of genes simultaneously. These expression levels are the outcome regulatory realtions among gene. So inferring gene regulatory network from these microarray data is known as a reverse engineering problem. Though its like a traditional inverse problem ,so solution is not like a trivial. The problem presents a large number of unknowns within a small sample size. Moreover DNA microarray data is heavily affected by noise. As a result, a powerfull mathematical technique with reliable inference algorithm is needed to solve the problem. Thus, on the framework of Artificial Bee Colony algorithm, I am using S-system model which is very efficient for solving complex biological problems for inferring Gene Regulatory Network.



## 1.1 Motivation

With the help of DNA microarray technology it is possible to measure expression of thousands of genes simultaneously. Gene regulatory network provides information about function inside a cell. Information on gene regulatory network can be used on biological applications. Prediction on these networks can also help in bio technical projects as these computational techniques are faster and cheaper than experimental labs. Hence, studying gene regulatory network we can have the knowledge of internal side of organism and observing their behaviour we can use those data.

There was a time when diseases were identified after watching symptoms of patients. Then with the development of medical science diseases were treated after testing blood, cough, stool etc. But these processes are not enough efficient to identify a disease too early. Many people do not get enough time for treatment even after the disease is identified, because we could not identify the disease early. At present, many diseases are characterized after watching abnormal gene expressions. If we can understand and can manipulate gene expressions of gene regulatory networks at the cellular level, and then we can easily identify a disease before it can spread over the whole body [11]. Unveiling gene regulatory network will allow us in tissue development. And more accurate knowledge of GRN shall lead us towards replacement tissues even whole organ of a body. For example if one's heart is having a chronic disease or injury and unable to pump blood then he may have the heart repaired from the new muscle cells developed from other cells from his own body [11].

Different modeling techniques have been developed so far to reconstruct gene regulatory network. Some of them use Boolean networks, some use Bayesian networks, some use dynamic Bayesian network, system of ordinary differential equation (ODEs), neural networks etc. All of these framework face the complexities associated with the problem. The problem involves a large number of unknown with a relatively small sample size. Also the gene expression data is affected by noise. To address those problems S-system model is used. It deals with the complex non-linear behaviour with of gene regulatory networks. Also it provides a smaller size of parameters. An algorithm based on the framework of artificial bee colony algorithm is used as the inference algorithm.

## 1.2 Aims

Ensuring accuracy is the main important thing that needs to consider when reconstructing gene regulatory network. In this research work the S-system has been used for modeling technique. S-system can capture the complex non-linear behavior of gene regulatory networks well. This model also provides a small amount of parameters or unknowns to find out. As inference algorithm artificial bee colony is used which is an optimization algorithm based on the intelligent foraging behavior of honey bees. Further two genetic operators have been integrated within the framework of the artificial bee colony algorithm. These operators are mutation and crossover. The objective of this study is to verify the validity of S-system with artificial bee colony algorithm. To the best of my knowledge and investigation, this is the first time artificial bee colony algorithm has been incorporated with the S-system model to address the problem of gene regulatory network. Using the proposed framework gene regulatory network has been inferred from artificial noise less data, noisy data on a noise scale of 5%, 10 %, 15% and on real data.

## 1.3 Scope

In this research work a reverse engineering approach based on S-system and artificial bee colony algorithm has been presented for inferring gene regulatory network from gene expression data. Genetic algorithm operators mutation and crossover are integrated within the framework of the artificial bee colony algorithm. To the best knowledge and investigation this is the first time artificial bee colony algorithm has been incorporated with S-system model to infer gene regulatory network. This is the main contribution of the presented study. Specifically, the proposed framework has been tested with the following cases :

- Reconstruction of gene regulatory network from artificial noise free data of a network of 5 genes.
- Reconstruction of gene regulatory network from artificial noisy data of a network of 5 genes, 5%, 10%, 15% noisy data were used.
- On a real network of 5 gene

The proposed approach has successfully reverse engineered underlying networks of those artificial noise free and noisy data and real data..

## 1.4 Ogranization of the thesis

This thesis contains six chapters. Outlines of these chapters are given below:

- **Chapter 1** gives an introduction to the problem and an overview of this work. This chapter also describes the motivation aims and objective of this research.
- **Chapter 2** describes related biological topics.
- **Chapter 3** describes some existing gene regulatory network inference techniques. This chapter also describes their strengths and weak points.
- **Chapter 4** describes the proposed framework. S-system and artificial bee colony algorithm has been discussed in detail.
- **Chapter 5** presents the experimental results. Experiment is done using artificial networks.
- **Chapter 6** presents the application of gene regulatory network.
- **Chapter 7** summarizes the whole book. Also identifies future scope of this study.

## Chapter 2

### Related Biological Study

Cell is the smallest unit in living organisms. It is the basic structural and functional unit in all life forms. Cell is often called as the building blocks of life. Cell contains the genetic materials DNA and RNA. Biological information of an organism is encoded in DNA. Cell contains gene, protein, DNA and RNA. Thus gene and protein are two very important things inside a cell. Studying gene regulatory network gives information about the complex biological systems and how they will respond to different situations. By studying gene regulatory network it is possible to gather information about the effect of different diseases on living organisms. Thus gene regulatory network can give insights on drug design. DNA microarray technology analyzes the expressions of thousand of genes simultaneously. This huge amount of data opens a way to study and analyze gene regulatory networks. Different biological terms related to this problem have been described in this chapter in the following sections.

### 2.1 Gene

Gene is the unit of heredity in living organism. Genes are encoded inside the strands of DNA. Genes hold and pass hereditary information across generations. Living organisms depend on genes heavily as they create protein and functional RNA chains. Genetic codes are stored in genes which later get translated into functional proteins. Genes, which are made up of DNA, act as instructions to make molecules called protein.

Human genome contains genes from 30,000 to 35,000 encoded by total nucleotide base pairs (A, C, T, and G) [11]. Inside the human body each gene has two copies inherited from each parent. All the people have the same gene mostly, but a small number of genes (less than 1 percent of total) are slightly different between people. These small differences contribute to each person's unique physical feature.

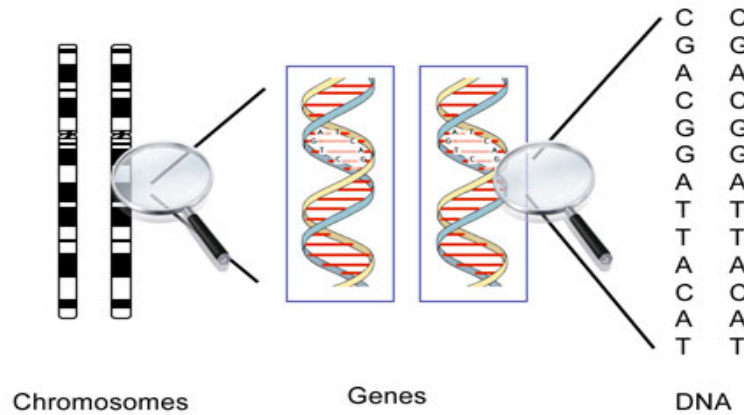


Figure -2.1: Chromosome inside a cell is made up of genes and genes are made up of DNA.

## 2.2 Functional structure of a gene

All genes have regulatory regions in addition to regions that explicitly code for a protein or RNA product. A regulatory regions shared by almost all genes is known as promoter, which provides a position that is recognized by the transcription machinery when a gene is about to be transcribed and expressed. A gene can contain multiple promoters. Other regulatory regions include enhancers, which can compensate for a weak promoter. Eukaryotic promoter regions are much more complex and difficult to identify than prokaryotic promoters.

## 2.3 Gene Regulation

The sequential processes that go on inside genes are termed as “Gene Regulation” and among those processes; we are successful to understand only two processes. Those are transcription and translation (Fig-4). These two processes control the level of gene’s expression and fix the amount of protein [9]. mRNA means “messenger RNA”, a large family of RNA molecules that convey genetic information from DNA to the ribosome, where they specify the amino acid sequence of the protein products of gene expression. And the amount of mRNA produced during transcription is a measure how active or functional a gene is [4]. A gene regulation system is made up of genes, cis-elements and regulators. Regulators are mainly protein. The genes, regulators and regulatory connection between them make a genetic regulatory network [4]. Genetic regulatory network can be said as an abstract mapping of gene regulation in living cells so that it can help us to predict the system behavior of living organism [12].

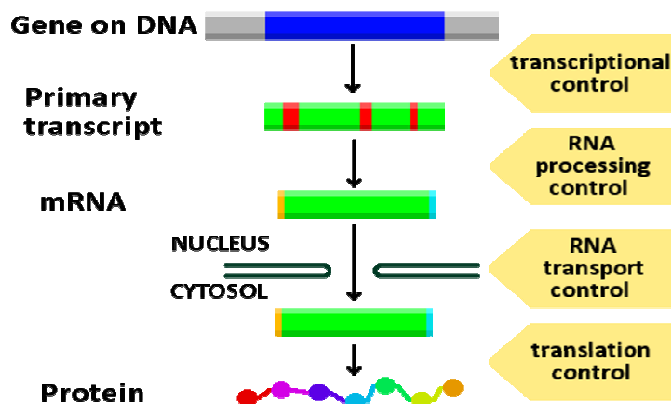


Figure-2.2: Gene regulation in living cell. Protein production of a gene present in a DNA from transcription and translation process).

## 2.4 DNA

Deoxyribonucleic acid or DNA stores the genetic instruction used in the development and functioning of all living organisms. Many viruses also contain DNA. DNA is a nucleic acid. Protein, carbohydrates and nucleic acids are the three major macro molecules essential for all life forms. DNA has a double helix structure. DNA molecule consists of two biopolymer strands coiled around each other to form a double helix. DNA is information storage. Both the strands of double helix structure have the same biological information.

## 2.5 RNA

Ribonucleic acid or RNA is one of the three major macro molecules inside a cell. It takes part in different biological roles. Like coding, decoding, regulation and gene expression. DNA, RNA, protein are the three major macromolecules essential for all forms of life. Like DNA, RNA is assembled as a chain of nucleotide. Structurally it is different from DNA. DNA has a double helix structure with two strands while RNA mostly contains one single strand. Messenger RNAs (mRNA) carry genetic information that controls the protein synthesis process. There are also different types of RNAs like – transfer RNA (tRNA), ribosomal RNA, regulatory RNAs etc.

## 2.6 Protein

Protein participates in virtually every process inside a cell. Genetic codes encoded in genes help assembling proteins from amino acid. This process is known as protein synthesis. Proteins are required for the structure, function and regulation of body's tissues and organs. Proteins are made up of hundreds or thousands of smaller units called amino acids, which are attached to one another with long chains. There are 20 different types of amino acids that can be combined to make a protein. The sequence of amino acid determines each protein 3-dimensional structure and its specific function. Messenger proteins, such as some types of hormones, transmit signals to coordinate biological processes between different cells, tissues and organs.

## 2.7 Gene Regulatory Network

Genetic Regulatory Networks are the basic functional units in living organism. Living body cells are composed of millions of cells. If we look closely inside a cell then we find its basic characteristics are vested inside chromosomes. Genes carry the necessary information for making proteins in a living body. Few factors related to gene expression (i.e. protein production) decide the behavior, structure and function of a cell. Type of protein produced, the quantity of protein produced and duration of production are main factors [11].

## 2.8 Structure of Gene Regulatory Network

An actual Genetic regulatory Network (actual means, with all necessary information that are present in a Genetic Regulatory Network) is very difficult to understand (Fig-2.3). But an artificial Genetic Regulatory Network is shown in fig-2.4, which can be a good example to understand the real complexity of the structure of a gene regulatory network.

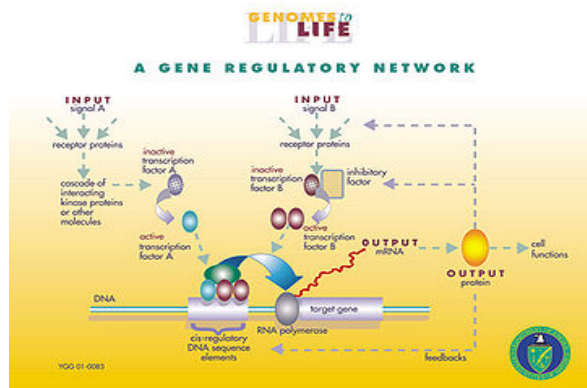


Figure -2.3:

Figure 2.3: Gene Regulatory Network added with all information, which is very difficult to understand.



Figure: 2.4

Figure 2.4: An artificial Genetic Regulatory Network. Here, it is a simulated sparse Genetic Regulatory Network with  $n=100$  genes and 100 connections.

## 2.9 DNA microarray

Human genome is the complete set of genetic information for humans (*Homo sapiens*). This information is encoded as DNA sequences within the 23 chromosome pairs in cell nuclei. And a human genome contains between 30,000 and 35,000 genes encoded 3.1 billion total nucleotide base pairs (A,C,T,G) [11]. Now if we select genes one by one and try to evaluate its interaction with other genes, then it will be very expensive and time consuming. But due to the advancement of biotechnology several methods has been developed to measure gene expression levels. Among those methods DNA microarray is one, which can measure gene expression efficiently. Using DNA microarray we can measure expression level of thousands of genes simultaneously [12].



Fig-2.5: DNA microarray image with an automated system background.



Expression profiles are made using the data from DNA microarray. Expression levels from different experiments and time interval are put together for making these profiles [23]. Ultimately, the study of Genetic Regulatory Network has become much easier due to the introduction of DNA microarray technology for measuring gene expression levels.

## **2.10 Gene Expression profiling**

Gene expression profiling is the measurement of the activity (the expression) of thousands of genes at once, to create a global picture of cellular function. These profiles can, for example, distinguish between cells that are actively dividing, or show the cells react to particular treatment. Many experiments of this sort measure an entire genome simultaneously, that is every gene present in particular cell. With the advancement in biotechnological sector, DNA microarray technology was invented. This technology measures the relative activity of previously identified target genes. Sequence based techniques, like serial analysis of gene expression are also used for gene expression profiling. Expression profiling provides new information about what genes do under new conditions. Overall, microarray technology produces reliable expression profiles. Many times more efforts are given to analyze these data than performing initial expression.

## Chapter 3

### Literature Review

Proteins are the main actors inside a cell. So information on gene regulatory network provides information about cellular process and behavior. Gene expression data is the outcome of regulatory relations among genes. Normal flow of problem is that knowing the regulatory relations expression data can be generated. But practically we got the expression data from experiments but not any information about regulatory relationships. So gene expression data is analyzed to predict regulations from them. Thus the problem becomes a reverse engineering problem. This reverse engineering has complexities like large number of unknowns within small sample size, defective expression data etc. Behavior study of different diseases can gain help from gene regulatory network study. So far different techniques have been developed to solve the gene regulatory network inference problem. Two main entities are involved with this problem:

- A mathematical model that describes relations among genes
- A search algorithm to find some parameters within the framework of the mathematical model.

Mathematical model gives abstract structure of the network. It defines a structure through a mathematical formula or relation. It provides a set of parameters within the model that describes the relationship among genes. Among these parameters some can be regulatory and some can be different kinetic parameters that express the behavior of the network.

Different approaches have been used to model gene regulatory network. Boolean networks, Bayesian networks, neural networks, linear time variant etc. are some of the techniques to model gene regulatory network. As the searching algorithm genetic algorithms, swarm intelligent based techniques are some of the ways that has been used so far.

### 3.1 Modeling techniques

A model describes the regulatory relations among genes in a gene regulatory network. It provides an abstract structure of network. A model mainly describes the structure through a formula. This formula contains different parameters to describe the regulatory relations among genes and also other properties of the network. Some of them are describes in the following sections.

### 3.1.1 Boolean Networks

Among the existing models, Boolean networks are the simplest. In Boolean network modeling each gene, each input and each output is considered as a node in directed graph. An edge is there from one node to another if there is a link between two nodes. Each node can have two states “on” and “off”. “On” corresponds to the gene being expressed. “Off” means the substance is being present. Time is considered as proceeding in discrete steps. At each new state of a node is Boolean function of the prior states of nodes having edge with current nodes. [11]

A Boolean network is a directed graph  $G(X, E)$  where the nodes  $x_i \in X$ , are Boolean variables. To each node,  $x_i$  is associated a Boolean function,  $b_i(x_{i1}, x_{i2} \dots, x_{il})$ ,  $1 \leq n$ ;  $x_{ij} \in X$ , where the arguments are all and only the parent nodes of  $x_i$  in  $G$ . Together, at any given time, the states (values) of all nodes represent the state of the network, given by the vector  $S(t) = (x_1(t), x_2(t) \dots, x_n(t))$ , [1]. For gene networks the node variables correspond to level of gene expression to either up or down. The Boolean function at the nodes, model the aggregated regulation effect of all their parent nodes. Boolean networks are simple. When number of entities is small and only qualitative knowledge is available Boolean networks can provide meaningful insights. But these networks do not correctly model the dynamics of transcription factor that down regulates itself. Also this model is computationally expensive to analyze larger networks.

### 3.1.2 Bayesian networks

Bayesian networks are probabilistic models. They combine two area probability and graph.  $G(X, E)$ . Where the nodes,  $x_i \in X$ , are random variables that describes gene expressions and the edges represent the dependencies between nodes. The random variables are drawn from a conditional

probability distribution. A Bayesian network implicitly encodes the Markov assumption that gives its parents; each variable is dependent of its –descendents. With this assumption each Bayesian network uniquely specifies a decomposition of joint distribution over all variables down to the conditional distributions of the nodes. Besides the set of dependencies a Bayesian network implies a set of independencies too. This probabilistic framework is very appealing for modeling casual relationships because one can query the joint probability distribution for the probabilities of events (represented by the nodes) given other events.[][]

### 3.1.3 Linear Time variant

Linear time variant is a linear formalism. Gene regulatory networks have complex non linear relations. Linear models typically are not a good match for them. But linear time variant model can capture the non linear relations among genes. But it is simpler than non linear models. Linear time variant has the following equation:

$$Z_i(t) = \sum_{k=1}^n W_{i,j}(t)X_i(t)$$

Where  $W_{i,j}$  indicates the regulatory values. A positive value indicates activation; negative value indicates repression where zero indicates regression.  $W_{i,j}$  is a time varying function.

A linear time variant model based approach has been proposed in [1]. With this model it used a self adaptive differential evolution approach as reverse engineering algorithm. Self adaptive differential evolution converges faster with more certainly. As fitness evaluation function they used the widely used mean squared error (MSE) function.

## 3.2 Searching Algorithm

Different algorithms have been used so far to address the problem of finding the gene regulatory network. Most of them are evolutionary approaches or swarm based approaches. Differential evolution, self adaptive differential evolution, particle swarm optimization are some of the

widely used techniques. These algorithms are population based search algorithms. Most of them starts with random solutions and try to improve the solution using multiple iterations.

### **3.2.1 Differential Evolution**

Differential Evolution (DE) is a method that optimizes a problem by iteratively trying to improve a candidate solution with measure to improve a candidate solution with regard to a given measure of quality. Differential evolution is a meta-heuristic approach. Differential evolution maintains a population of candidate solutions. In each iteration and generation it creates a set of new solution from existing population. The new solution generation procedure follows a simple formula. Typically the genetic algorithm operator mutation and crossover are used to generate new solutions from existing ones. After generating new solutions a greedy selection approach is used on some fitness evaluation criteria. [][]

### **3.2.2 Self Adaptive Differential Evolution**

Self Adaptive Differential Evolution is a variant of the classic differential evolution approach. In differential evolution approach users set the parameters of the algorithm at the beginning of the procedure. This parameter remains same throughout the whole process. Control parameters are the main factors of the exploration process in a differential evolution algorithm. Self adaptive differential evolution works with a variable or adaptive set of parameters. It uses the mutation crossover and selection operators as differential evolution but the values of the parameter scaling factor and CP (Crossover probability) are calculated adaptively. In each iteration of the algorithm these parameters are changed according to some specific formula or conditions. [][]

### **3.2.3 Particle Swarm Optimization**

Particle swarm optimization (PSO) maintains a set of particles or solutions. Each particle has a velocity and position associated with it. Particles search through the entire space by updating their positions and velocity. Particle swarm optimization maintains two best result local best and

global best. Local best corresponds to each particle's best solution so far and global best solution corresponds to overall best particles among all solutions. Each particle movement is influenced by its local best known position but it is also guided towards the best the best known position in the search space, which are updated as better positions are found by other particles. PSO is a meta-heuristics as it makes over few or no assumptions about the problem being optimized and can search very large spaces of candidate solutions.

PSO starts with random candidate solution or particles. After that it iterates and each iteration changes the velocity of each particle to a better solution. PSO maintains global and local aspects of the search process separately. For each particle it maintains a local best. PSO also maintains a global best which is ultimate solution of the problem. Global best is selected from local best.

### **3.3 Summary**

So far many different techniques have been developed to infer gene regulatory networks. Among them Bayesian network, Boolean network, Recurrent Neural network, Linear Time variant , s-system approaches are widely used. All of these deal with complex nonlinear behavior of biological systems. They all have their pros and cons. With these models different searching techniques have been used. Some of the widely used algorithms are genetic algorithms, evolutionary algorithms, differential evolution algorithm, swarm intelligent based techniques like particle swarm optimization etc.

## Chapter 4

### Proposed Approach

#### 4.1 S-system model

Biological systems are very much complex in nature. So it is not very easy to model a biological system. But we can model using systems of ordinary differential equations (ODEs). S-system is a recent as well as popular class of ODEs for modeling biological systems. The parameters of an S-system are generally estimated from time course profiles [3]. Interacting components of organisms together make a biological system. The process of expression and the interaction of these components are nonlinear [2]. A non-linear system means its output is not directly proportional to its input. For example we cannot predict the exact weather forecast because a simple change in one part of the system creates complex effects throughout. The nonlinear behavior and gene expressions can be modeled using S-system class. An S-system that represents the dynamics of n gene is described as:

$$\frac{dX_i}{dt} = \alpha_i \prod_{j=1}^n X_j^{g_{i,j}} - \beta_i \prod_{j=1}^n X_j^{h_{i,j}}, i = 1, 2, \dots, n$$

Where,

- $X_i$  means the expression level of i-th gene
- n means the total number of genes in the network.
- $\alpha$  and  $\beta$  represent constant scale parameter, indicating the intensity of the relationship between  $X_i$  and other genes in the network.
- $g_{i,j}$  and  $h_{i,j}$  are exponential parameters, indicating the action of whether gene i activates or inhibits gene j respectively[2].

For example in figure-4.1 a network is developed from the values of respective parameters of the equation of S-system.

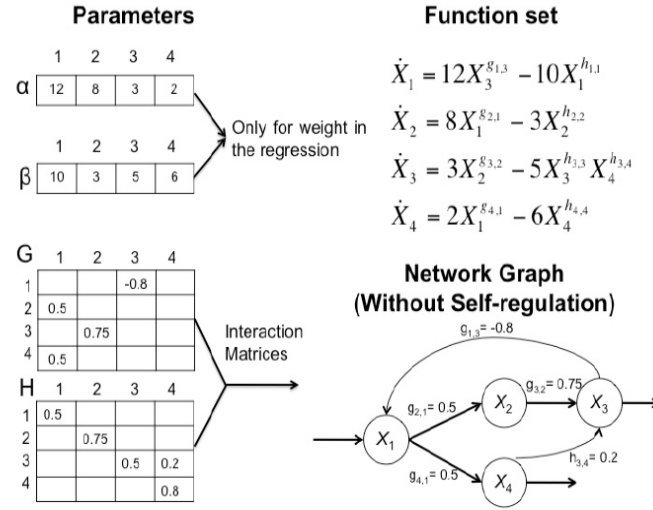


Figure 4.1: Reconstruction procedure of a network using S-system.

- The left hand side of the figure shows predicted  $\alpha$ , G,  $\beta$  and H parameters.
- The top right gives the function format of the system with the predicted parameters.
- The bottom right gives graphical representation of the network, according to the predicted elements. Each non zero  $g_{ij}$ ,  $h_{ij}$  gives an activation or inhibition from gene j to gene i, respectively. Each elements that regulate self regulation (arrows with no joining properties i.e.  $g_{ii}$  and  $h_{ii}$ ) are not shown in the graph.

S-system model can be used to infer genetic regulatory network. Mainly based on the equation and putting the parameters value in the equation, we can reconstruct a regulatory network and match with the profiles of gene regulatory network found from DNA microarray data.

## 4.2 Mutation:

Mutation is a genetic operator used to maintain genetic diversity from one generation of population of a genetic algorithm to the next. It is analogous to biological mutation. Mutation alters one or more gene value in a chromosome from initial state. In mutation the solution may



change entirely from the previous solution. Hence GA can come to better solution by using mutation. Mutation occurs during evolution according to a user definable mutation probability. The classic example of a mutation operator involves a probability that an arbitrary bit in a genetic sequence will be changed from its original state. A common method for implementing the mutation operator involves generating a random variable for each bit in a sequence. This random variable tells whether or not a particular bit will be modified. When the gene encoding is restrictive as in permutations problems, mutations are swaps, inversion and scrambles. The purpose of mutation operator is preserving and introducing diversity.

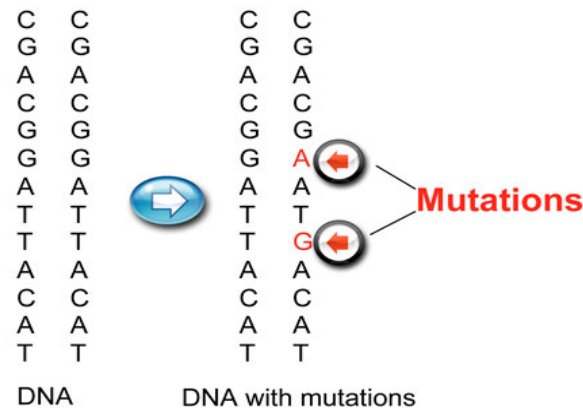


Figure 4.2: Mutation inside DNA.

A common way of doing mutation operation is for each target vector  $X_i^G$  with  $i = 1, 2, \dots, N$ , a mutant vector  $V_i^{G+1}$  is generated based on the following equation:

$$V_i^{G+1} = X_{r1}^G + F (X_{r2}^G - X_{r3}^G)$$

Here, the integer  $r1$ ,  $r2$  and  $r3$  represent random and mutually different indexes and also different from  $i$ .  $F$  is known as the scaling factor and controls the amplification of the differential variation.

### 4.3 Crossover

Crossover is the operation which takes more than one parent solution and produces a new child solution from them. Different types of crossover include single point crossover, two point

crossover, uniform crossover, arithmetic crossover etc. Crossover often involves the property crossover probability based on which value is inherited in child solution from the parent.

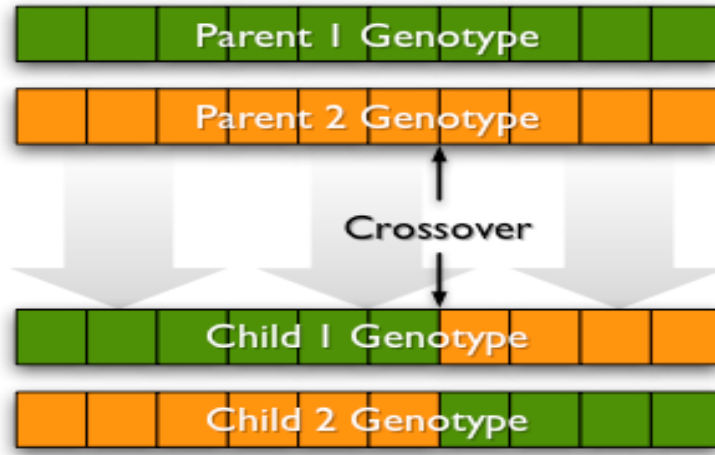


Figure 4.3: Crossover between parents

Setting up this parameter properly impacts the performance of solutions in later generations. One common approach to perform crossover is fitness proportionate selection. In this selection procedure each solution is assigned with a fitness value and the probability of that solution depends on the fitness level. A higher fitness valued solution will have the higher possibility of selection. The following equation is used for crossover:

$$p_i = \frac{f_i}{\sum_{k=1}^n f_i}$$

Here,  $p_i$  means probability,  $n$  means number of genes and  $f_i$  means fitness of  $x_i$ .

#### 4.4 Artificial Bee Colony Algorithm

Within last 15 years several algorithms has been developed depending on different intelligent behaviors of honey bee swarms. Among those, artificial bee colony (ABC) is one which has been most widely studied on and applied to solve real world problems. This algorithm was developed by Karaboga in 2005. And it has also been researched and developed more efficiently by

Karaboga himself and several other people [8]. This algorithm has been used in several promising sectors. Like Neural Network, image processing, wireless sensor network, different engineering problems purposes etc [8]. In this algorithm, there are three types of bees in a colony.

- Employed bees
- Onlooker bees
- Scout bees

A bee is going to a food source visited by itself previously is called an employed bee. A bee waiting in the dancing area of the employed bees to choose a food source is called an onlooker bee. A bee searching randomly for new food source is called scout bee. So according to ABC the colony of bees can be divided into two major parts. The first part is called employed bees and the second part by onlookers. It is assumed that one bee is employed for each food source. So number of food source is equal to the number of employed bees. Employed bees go to their food source and dance. When they abandon a food source it becomes a scout. Onlooker bees observe the dance of employed bees and choose food sources based on dance.

The general algorithmic structure of ABC optimization approach is given below:

1. Initial Phase
2. **REPEAT**
3.       Employed Bees Phase
4.       Onlooker Bees Phase
5.       Scout Bees Phase
6.       Memorize the best solution achieved so far
7. **UNTIL** (Cycle = maximum Cycle Number or a maximum CPU time)

In robust search process, exploration and exploitation process must be carried out together. In Artificial Bee Colony algorithm we see the scout bees control exploration process, while the employed bees and onlookers' carryout the exploitation process in search space. The number of employed bees and onlookers are equal to the number of population. The employed bee whose food source has been exhausted becomes a scout bee. The position of an enhanced nectar amount of food represents a possible solution to the optimization problem. When using artificial bee

colony algorithm, to optimize a function, potential food source represents a potential solution to the problem. Nectar amount of a food source denotes the quality of the solution.

#### 4.5 Proposed Algorithm

To infer gene regulatory network, based on S-system model an algorithm is designed in this research work. To the best of my knowledge and investigation this is the first time artificial bee colony algorithm has been incorporated with S-system model to infer gene regulatory network. Two genetic algorithm (GA) operators namely mutation and crossover has been used. Here food sources are the candidate solutions. Nectar amount is the fitness of a solution. Employed and onlooker bees discover neighbor food sources or new solutions from mutation and crossover operators. The scout bee phase occurs after the specific interval of the overall process.

The algorithm works as follows:

1.	Initialize N random employed bees using random variables
2.	$G=1$
3.	Repeat
4.	for $i \leftarrow N$ do
5.	Produce new solution $y_i$ from $X_i$ using Mutation and Crossover
	Procedure
6.	Evaluate fitness of the new solution $y_i$
7.	Replace $x_i$ with $y_i$ if $X_i$ has better fitness
8.	End
9.	for $i \leftarrow N$ do
10.	Choose a solution $x_i$ from a higher fitness
11.	Produce new solution $y_i$ from $X_i$ using Mutation and Crossover
12.	Evaluate fitness of the new solution $y_i$
13.	Replace $x_i$ with $y_i$ if $y_i$ has better fitness
14.	End
15.	Choose a solution with higher fitness from all solutions N

16.	Perform the scout bee phase of generating value
17.	$G = G+1$
18.	UNTILL $G = G(\max)$

#### 4.6 Fitness Evaluation criteria

A fitness evaluation criterion is needed to find the goodness of a solution. The most commonly used fitness evaluation criteria is the qualitative difference between the estimated and target data. This criterion uses a function called squared error. And the function is as follows:

$$\text{Fit}(x_i) = (e_i - d_i)^2$$

**Here**  $x_i$  means i-th gene,  $e_i$  means expected value and  $d_i$  means target value.

## Chapter 5

### Experimental Results

In this research paper a reverse engineering algorithm has been developed on the framework of artificial bee colony algorithm for the reconstruction of gene regulatory network. Based on the equation of S-system model we are generating sample networks and selecting through artificial bee colony algorithm. So we can say that our algorithm worked if the expression levels found by the best solution is equal or approximately equal to the expression levels in the data for respective genes. And thus we can construct our network on the parameters inside the best solution. For the verification of the proposed algorithm, the algorithm was tested in noiseless data and noisy data. Then it was also tested with real data. And algorithm successfully found all the regulations in noiseless, noisy as well as real data. So, on a network of 5 genes the detail outline of the testing procedure of proposed algorithm is as follows:

- Noiseless data
- 5% noisy data
- 10% noisy data
- 15% noisy data
- Real data

#### 5.1 Experimental Setup

The code of the proposed algorithm is written in Java programming language on eclipse platform. While tested on a machine of 4GB ram and core i3 processor it takes an average time of 5.45 minutes to give the result of one dataset. This time increases with the increase of noise level.

The ranges of the parameters, variables and other necessary values are shown below:

Name	Minimum	Maximum
$\alpha$ and $\beta$	1	10
g and h	-0.99	0.99
G (Total iteration of algorithm)		10000
N (Employed bee and Onlooker bee number)		100

F(Scaling factor)		.5
-------------------	--	----

## 5.2 Noiseless data

First the proposed algorithm was tested on a noise less data. Inferred parameters of the data are as follows:

	Gene 1	Gene 2	Gene 3	Gene 4	Gene 5
$\alpha$	7	5	5	10	9
$\beta$	7	10	1	7	1

**Inferred g values:**

i	j→	Gene 1	Gene 2	Gene 3	Gene 4	Gene 5
Gene 1		-.04	.21	.14	-.33	.46
Gene 2		.79	.24	-.1	-.86	.67
Gene 3		.63	.28	-.71	-.37	.22
Gene 4		.27	-.13	-.48	.86	-.19
Gene 5		.8	-.16	.95	-.39	.98

**Inferred g values:**

i	j→	Gene 1	Gene 2	Gene 3	Gene 4	Gene 5
Gene 1		-.18	.18	.56	-.02	-.62
Gene 2		.77	-.95	.8	-.01	-.66
Gene 3		-.3	-.38	-.56	.82	-.31
Gene 4		-.8	.42	.55	-.84	.64
Gene 5		.4	.28	.74	.94	-.81

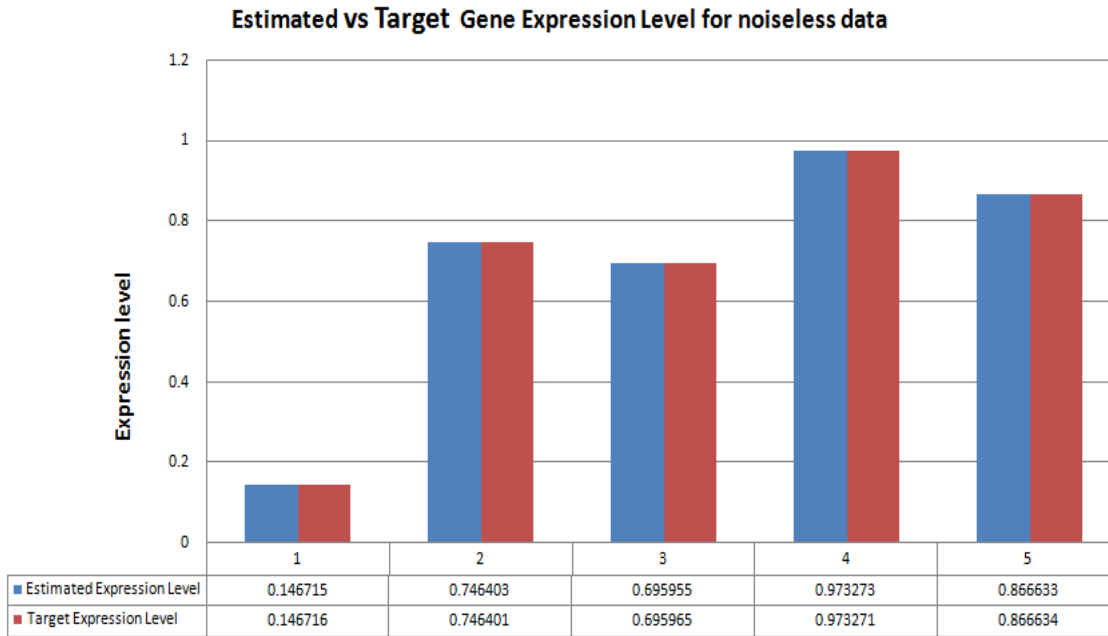


Figure 5.1: We can see the accuracy level of the algorithm. Estimated and target graph are exactly same. Thus the parameters identified are valid.

(Here, x-axis indicate the gene number and y-axis indicate expression level)

### 5.3 5% noisy data

After noiseless data the proposed algorithm was tested with 5% noisy data. And the Inferred parameters of the data are as follows:

	Gene 1	Gene 2	Gene 3	Gene 4	Gene 5
$\alpha$	10	5	1	4	1
$B$	9	7	3	2	3

#### Inferred g values:

i	j→	Gene 1	Gene 2	Gene 3	Gene 4	Gene 5
Gene 1		-.85	.76	-.57	-.86	-.59
Gene 2		-.05	-.82	.93	-.44	-.08
Gene 3		.23	.59	-.72	.52	-.48
Gene 4		-.37	.81	.58	.53	.71
Gene 5		0	.53	-.68	-.87	.93

#### Inferred h values:



i	J→	Gene 1	Gene 2	Gene 3	Gene 4	Gene 5
Gene 1		-.48	-.67	.86	-.31	.77
Gene 2		.43	-.45	.31	-.21	-.6
Gene 3		-.01	-.38	.63	.53	.08
Gene 4		-.56	-.22	.74	-.03	-.65
Gene 5		-.02	00	-.96	-.18	.95

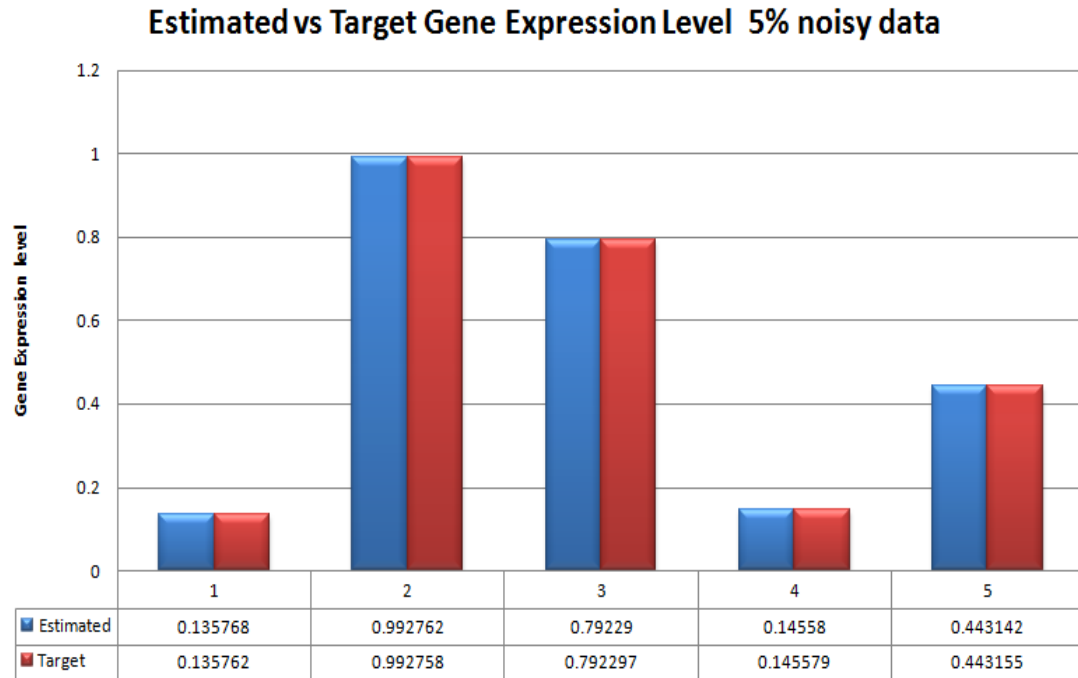


Figure 5.2: We can see the accuracy level of the algorithm. Estimated and target graph are exactly same. Thus the parameters identified are valid.

(Here, x-axis indicate the gene number and y-axis indicate expression level)

#### 5.4 10% noisy data

The proposed algorithm was tested with 10% noisy data. And the Inferred parameters of the data are as follows:

	Gene 1	Gene 2	Gene 3	Gene 4	Gene 5
$\alpha$	10	5	1	4	1
$\beta$	9	7	3	2	3

**Inferred g values:**

i	j→	Gene 1	Gene 2	Gene 3	Gene 4	Gene 5
Gene 1		-.85	.76	-.57	-.86	-.59
Gene 2		-.05	-.82	.93	-.44	-.08
Gene 3		.23	.59	-.72	.52	-.48
Gene 4		-.37	.81	.58	.53	.71
Gene 5		.00	.53	-.68	-.87	.93

**Inferred h values:**

i	J→	Gene 1	Gene 2	Gene 3	Gene 4	Gene 5
Gene 1		-.48	.67	-.86	-.31	.77
Gene 2		.43	-.45	.31	-.21	-.6
Gene 3		-.01	.38	.63	.53	.08
Gene 4		-.56	-.22	-.74	-.03	-.65
Gene 5		-.02	.00	-.96	-.18	.95



Figure 5.3: We can see the accuracy level of the algorithm. Estimated and target graph are approximately same. Thus the parameters identified are valid.

(Here, x-axis indicate the gene number and y-axis indicate expression level)

### 5.5 15% noisy data

The proposed algorithm was tested with 15% noisy data. And the Inferred parameters of the data are as follows:

	Gene 1	Gene 2	Gene 3	Gene 4	Gene 5
$\alpha$	1	4	3	5	6
$B$	3	8	8	7	6

**Inferred g values:**

i	j→	Gene 1	Gene 2	Gene 3	Gene 4	Gene 5
Gene 1		.1	-.04	.18	.91	-.28
Gene 2		-.64	-.6	-.04	.33	.69
Gene 3		-.78	.75	.35	.18	.1
Gene 4		-.31	-.31	.4	-.43	.92
Gene 5		.91	.88	.13	-.99	.1

**Inferred h values:**

i	J→	Gene 1	Gene 2	Gene 3	Gene 4	Gene 5
Gene 1		.87	-.06	.07	.12	-.86
Gene 2		.3	-.05	.74	.18	-.39
Gene 3		-.73	-.11	-.22	-.2	.92
Gene 4		-.85	-.17	.66	.33	.02
Gene 5		-.57	-.12	-.45	.85	.8

**Estimated vs target Expression level for 15% noisy data**

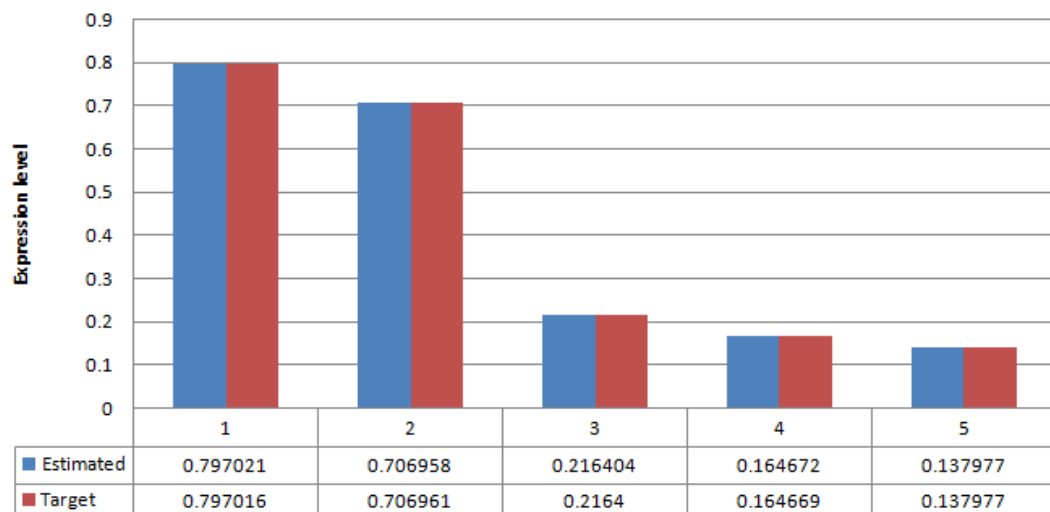


Figure 5.4: We can see the accuracy level of the algorithm. Estimated and target graph are approximately same. Thus the parameters identified are valid.

(Here, x-axis indicate the gene number and y-axis indicate expression level)

## 5.6 Real data

The proposed algorithm was tested with real data. And the Inferred parameters of the data are as follows:

	Gene 1	Gene 2	Gene 3	Gene 4	Gene 5
$\alpha$	1	3	3	1	8
$\beta$	10	6	8	3	5

### Inferred g values:

i	j→	Gene 1	Gene 2	Gene 3	Gene 4	Gene 5
Gene 1		-.99	-.31	.1	-.94	-.49
Gene 2		.53	.55	.59	.58	-.8
Gene 3		-.21	.23	-.9	.6	-.82
Gene 4		-.32	.04	.67	-.25	.85
Gene 5		-.7	.46	.79	-.65	-.49

### Inferred h values:

i	J→	Gene 1	Gene 2	Gene 3	Gene 4	Gene 5
Gene 1		-.63	.84	.34	.48	-.85
Gene 2		-.36	.41	.99	-.35	.22
Gene 3		-.97	.46	-.61	-.01	-.18
Gene 4		-.02	.57	.52	.42	-.65
Gene 5		.53	.96	.65	.04	-.29



Figure 5.5: We can see the accuracy level of the algorithm. Estimated and target graph are approximately same. Thus the parameters identified are valid.

(Here, x-axis indicate the gene number and y-axis indicate expression level)

## 5.7 Network Structure of the real data

### 5.7.1 Network Structure of real data (Production).

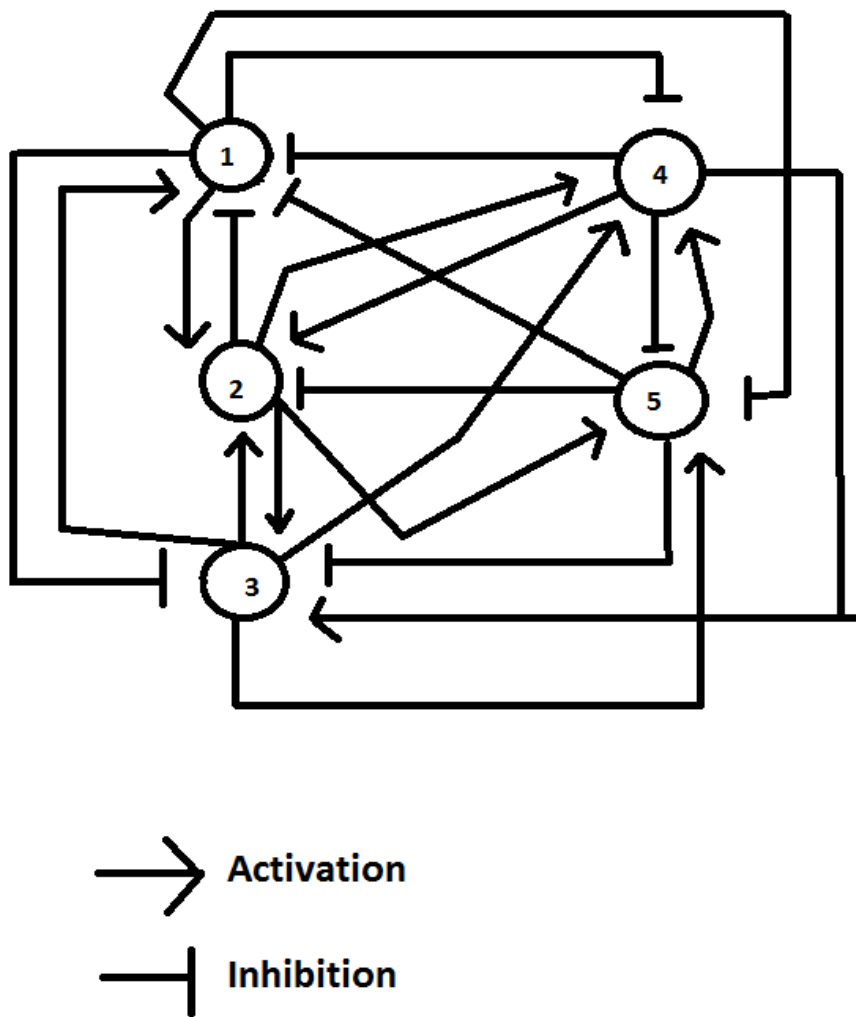


Figure 5.6 Network Structure of real data (Production). (Activation inhibition corresponds to g values of real data. Self regulations are not shown).

### 5.7.1 Network Structure of real data (Regression)

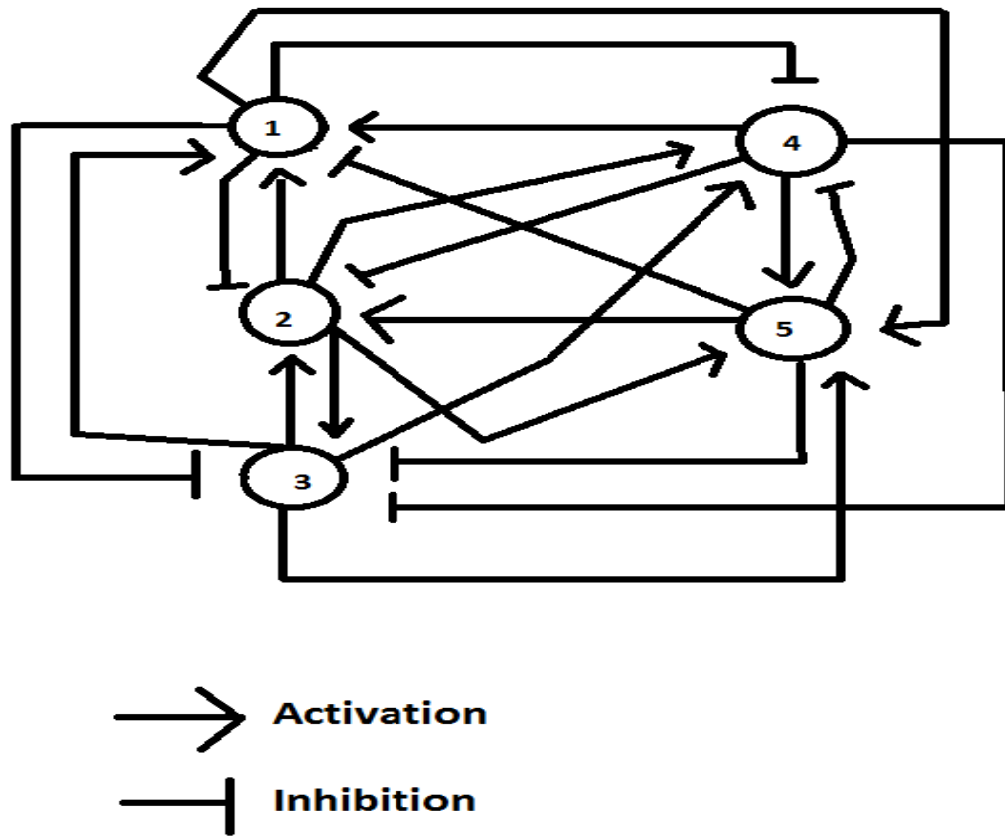


Figure 5.7 Network Structure of real data (Regression). (Activation inhibition corresponds to h values of real data. Self regulations are not shown)

## 5.8 Summary

Thus, we get the network structure of the real data used to test proposed algorithm. From all the data we found after testing with the proposed algorithm correctly infer the gene regulatory network and we justify those comparing with the target values. And we see that the algorithm shows correct regulation for noiseless data, noisy data with noise level 5%, 10%, 15% and also for real data.

## Chapter 6

### Applications

Gene regulatory network (GRN) has many real life applications. But to reveal those we need accurate calculations. Gene regulatory network can help us mainly in medical sector. Starting from early disease detection to side effect less medicine production GRN can be much efficient to design. Not only in medical sector but also we can use the inferred network to identify the characteristics of specific insects and thus we can produce organisms according to our own wish. The important applications of gene regulatory network are briefly discussed below:

#### 6.1 Identifying Diseases early

There was a time when diseases were identified after watching symptoms of patients. Then with the development of medical science diseases were treated after testing blood, cough, stool etc. But these processes are not enough efficient to identify a disease too early. Many people do not get enough time for treatment even after the disease is identified, because we could not identify the disease early. At present, many diseases are characterized after watching abnormal gene expressions. If we can understand and can manipulate gene expressions of gene regulatory networks at the cellular level, and then we can easily identify a disease before it can spread over the whole body [11].

#### 6.2 Organ development

Unveiling gene regulatory network will allow us in tissue development. And more accurate knowledge of GRN shall lead us towards replacement tissues even whole organ of a body. For example if one's heart is having a chronic disease or injury and unable to pump blood then he may have the heart repaired from the new muscle cells developed from other cells from his own body [11].



### **6.3 Side effect less medicine production**

When we are attacked by any disease, for treatment purposes we need to take more medicine so that we can reduce side effects. But if we can understand the gene regulatory network correctly then we can produce medicine which will be based on the treatment of particular type of gene. Thus the probability of having side effects reduces and also helps us not to take more medicine than necessity for reducing side effects.[11]

### **6.4 Creating nature friendly organisms**

Grabbing information about genetic networks will allow us to develop more nature friendly organisms. And we can use them in agricultural fields for killing insects. We can produce insect resistant crop. We won't have to use insecticides or chemicals which decrease the crop production and also hamper the fertility of land. This is a very innovative way of development in agriculture.[11]

These are the main important applications of genetic regulatory network. Other than this there are more applications of GRN is present but we need to make more accurate calculations on a large scale to find out more and more features that can be implemented using gene regulatory networks.

## **Chapter 7**

### **Conclusion and Future Work**

#### **7.1 Conclusion**

Gene regulatory network plays important role in the functional behavior of an organism. In a network gene pass information among them. The information is mainly carried out by protein. By the invention of DNA microarray technology, it has become possible to calculate gene expression level for several numbers of genes simultaneously. But the expression levels of these genes calculated by DNA microarray are not sufficient enough to infer the network structure. If we want to know the whole network then we need an efficient model to produce solutions and an algorithm to find the best suited solution. If the best solution found matches to target data then we come to conclusion that our algorithm is working correctly.

In this research work, to find the network structure of those genes, an algorithm is proposed on the framework of S-system model and artificial bee colony algorithm. The experiment was carried out in noiseless, noisy as well as real datasets. And the result found was promising. The proposed algorithm successfully inferred all the regulations. The result found was justified by comparing the target and the expected data, which ensures the strength of the proposed algorithm.

#### **7.2 Future work**

##### **7.2.1 Working on time series data**

In this research work I worked only on a network at a specific time, which ensures our algorithm by comparing with the target and estimated expression levels. But my future aim is to work on the network with multiple time series datasets, which will improve the inferring capability of the

algorithm. But for working on time series data we need another formula which can transform our created equation into time series. There are some useful and efficient processes to get time series data. Among them we can use Fourier Transformation procedure, squashing function; runge-kutta order method etc can be notable. Thus I need to find the best one suited for the dataset and work on the whole network. If we can do that then we will have more inferring ability of a network structure.

### **7.2.2 Reduce time complexity:**

Here, in this research work I mainly focused on the accuracy of the algorithm rather than time complexity. My future objective is to reduce the time complexity of program as much as we can. Time complexity is very important because the number of population we are working is in a small scale. But if we go for large scale network then we might need more time to get a single data. So our time complexity has be reduced for calculating larger network.

## REFERENCES

- [1] Akay, B., and Karaboga, D., 2015, “A Survey on the Applications of Artificial Bee Colony,” *Signal, Image, and Video Processing*. *Signal, Image and Video Processing*, **9**(4), pp. 967-990.
- [2] Almeida, J. S., & Voit, E. O., 2003, “Neural-Network-Based Parameter Estimation in S-System Models of Biological Networks,” *Genome Informatics*, **14**, pp. 1-19.
- [3] Almeida, J. S., and Voit, E. O., 2003, “Neural-Network-Based Parameter Estimation in S-System Models of Biological Networks,” *Genome Informatics*, **14**, pp. 114-123.
- [4] Aluru, S., 2005, *Handbook of Computational Molecular Biology*, CRC press. Florida. Chap.27.
- [5] Badallo, M. R., 2013, “Implementation and Testing of Two Bee-Based Algorithms in Finite Element Model Updating,” Master Thesis, KTH Royal Institute of Technology.
- [6] Borelli, F. F., de Camargo, R. Y., Martins, D. C., and Rozante, L. C., 2013, “Gene Regulatory Networks Inference Using a Multi-GPU Exhaustive Search Algorithm. *BMC Bioinformatics*, **14**(Suppl 18), S5.
- [7] Chang, Y. H., Gray, J. W., and Tomlin, C. J., 2014, “Exact Reconstruction of Gene Regulatory Networks Using Compressive Sensing,” *BMC Bioinformatics*, **15**(1), pp. 400-422.
- [8] Chowdhury, A. R., Chetty, M., and Vinh, N. X., 2013, “Incorporating Time-Delays in S-System Model for Reverse Engineering Genetic Networks. *Bioinformatics*”, **14**(1), pp. 1-22.

- [9] Crombach, A., Wotton, K. R., Cicin-Sain, D., Ashyraliyev, M., and Jaeger, J., 2012, “Efficient reverse-engineering of a developmental gene regulatory network,” *PLoS Comput Biol*, **8**(7), pp. e1002589- e1002589.
- [10] Gonzalez, O. R., Küper, C., Jung, K., Naval, P. C., and Mendoza, E., 2007, “Parameter Estimation Using Simulated Annealing for S-system Models of Biochemical Networks,” *Bioinformatics*, **23**(4), pp. 480-486.
- [11] Handley, D., 2002, “Evaluating Machine Learning Algorithms Used to Infer Gene Regulatory Network Structure,” Doctoral Dissertation, Carnegie Mellon University.
- [12] Kabir, M., Noman, N. and Iba, H., 2010, “Reverse Engineering gene regulatory network from microarray data using linear time-variant model,” *BMC Bioinformatics*, **11**(Suppl 1), S56.
- [13] Karaboga, D., and Basturk, B., 2008, “On the Performance of Artificial Bee Colony (ABC) Algorithm,” *Applied Soft Computing*, **8**(1), pp.687-697.
- [14] Karaboga, D., Gorkemli, B., Ozturk, C., & Karaboga, N., 2003, “A Comprehensive Survey: Artificial Bee Colony (ABC) Algorithm and Applications.” *Artificial Intelligence Review*, **42**(1), pp.21-57.
- [15] Kikuchi, S., Tominaga, D., Arita, M., Takahashi, K., & Tomita, M., 2003, “Dynamic Modeling of Genetic Networks Using Genetic Algorithm and S-System,” *Bioinformatics*, **19**(5), pp. 643-650.
- [16] Kimura, S., Ide, K., Kashiwara, A., Kano, M., Hatakeyama, M., Masui, R., Nakagawa, N., Yokoyama, S., Kuramitsu, S. and Konagaya, A., 2005, “Inference

- of S-system Models of Genetic Networks Using a Cooperative Coevolutionary Algorithm,” *Bioinformatics*, **21**(7), pp. 1154-1163.
- [17] Konak, A., Coit, D. W., & Smith, A. E., 2006, “Multi-objective optimization using genetic algorithms: A tutorial.” *Reliability Engineering & System Safety*, **91**(9), pp. 992-1007.
- [18] Kumar, S., Sharma, V. K., and Kumari, R., 2014, “Randomized Memetic Artificial Bee Colony Algorithm,” *International Journal of Emerging Trends & Technology in Computer Science*, **3**(1), pp. 52-62.
- [19] Kutalik, Z., Tucker, W. and Moulton, V., 2007, “S-system Parameter Estimation for Noisy Metabolic Profiles Using Newton-Flow Analysis,” *IET Systems Biology*, **1**(3), pp. 174-180.
- [20] Liang, S., Fuhrman, S., and Somogyi, R., 1998, “Reveal, a General Reverse Engineering Algorithm for Inference of Genetic Network Architectures,” *Pacific Symposium on Biocomputing*, **3**(3), pp. 18-29.
- [21] Liu, L. Z., Wu, F. X., and Zhang, W. J., 2014, “Estimating Parameters of S-systems by an Auxiliary Function Guided Coordinate Descent Method,” *Systems Science & Control Engineering: An Open Access Journal*, **2**(1), pp. 125-134.
- [22] Liu, P.-K. and Wang, F. S., 2008, “Inference of Biochemical Network Models in S-System Using Multi-Objective Optimization Approach,” *Bioinformatics*, **24**(8), pp. 1085-1092.
- [23] Lobovac, Z. and Olsson, B. 2003, “Towards Reverse Engineering of Genetic Regulatory Network,” *Master Thesis*, University of Skovde.
- [24] Margolin, A. A., Nemenman, I., Basso, K., Wiggins, C., Stolovitzky, G., Favera, R. D., and Califano, A., 2006, “ARACNE: an Algorithm for the Reconstruction of Gene

- Regulatory Networks in a Mammalian Cellular Context,” *BMC Bioinformatics*, **7**(Suppl 1), pp. 1-7.
- [25] Ng, K. M., 2006, “A Graphical Method for Automatic Code Generation from Extended S-System Petri Net Models,” Master Dissertation, Kolej Universiti Teknologi Tun Hussein Onn.
- [26] Noman, N., Palafox, L., and Iba, H., 2013, “Reconstruction of gene regulatory networks from gene expression data using decoupled recurrent neural network model,” *Natural Computing and Beyond*, pp. 93-103.
- [27] Omkar, S. N., Senthilnath, J., Khandelwal, R., Naik, G. N., & Gopalakrishnan, S., 2011, “Artificial Bee Colony (ABC) for Multi-Objective Design Optimization of Composite Structures,” *Applied Soft Computing*, **11**(1), pp. 489-499.
- [28] Savageau, M. A., 1988, “Introduction to S-systems and the underlying power-law formalism,” *Mathematical and Computer Modelling*, **11**, pp. 546-551.
- [29] Savageau, M. A., and Voit, E. O., 1987, “Recasting Nonlinear Differential Equations as S-Systems: a Canonical Nonlinear Form,” *Mathematical Biosciences*, **87**(1), pp. 83-115.
- [30] Sorribas, A., and González, A., 1999, “The Power-Law Formalism as a Tool for Modeling Hormonal Systems,” *Computational and Mathematical Methods in Medicine*, **2**(1), pp.19-38.
- [31] Tegner, J., Yeung, M. S., Hasty, J., and Collins, J. J., 2003, “Reverse Engineering Gene Networks: Integrating Genetic Perturbations with Dynamical Modeling,” *Proceedings of the National Academy of Sciences*, **100**(10), pp. 5944-5949.
- [32] Vilela, M. Chou, I.C., Vinga, S., Vasconcelos, A.T.R., Voit, E.O. and Almeida, J.S. “Parameter Optimization in S-System Models,” *BMC Systems biology*. **2**(1), p.35.

- [33] Vohradsky, J., 2001, "Neural Network Model of Gene Expression," The FASEB Journal, **15**(3), pp. 846-854.
- [34] Vrugt, J. A., Gupta, H. V., Bastidas, L. A., Bouten, W., & Sorooshian, S. (2003). Effective and Efficient Algorithm for Multi-objective Optimization of Hydrologic Models. Water Resources Research, **39**(8), pp.1-19.
- [35] Werhli, A. V., Grzegorzcyk, M., & Husmeier, D., 2006, "Comparative Evaluation of Reverse Engineering Gene Regulatory Networks with Relevance Networks, Graphical Gaussian Models and Bayesian Networks," BMC Bioinformatics, **22**(20), pp. 2523-2531.
- [36] Xinyi, Y., Dent, J. E., & Nardini, C., SPEM (S-system Parameter Estimation Method) Vignette.
- [37] Yoon, M. U., 2010, "Differential Equation Models and Numerical Methods for Reverse Engineering Genetic Regulatory Networks," Ph.D. Dissertation, University of Tennessee.
- [38] Zitzler, E., Deb, K., and Thiele, L., 2000, "Comparison of Multi-Objective Evolutionary Algorithms: Empirical Results," Evolutionary Computation, **8**(2), pp. 173-195.



## REFERENCES

- [1] Akay, B., and Karaboga, D., 2015, “A Survey on the Applications of Artificial Bee Colony,” *Signal, Image, and Video Processing*. *Signal, Image and Video Processing*, **9**(4), pp. 967-990.
- [2] Almeida, J. S., & Voit, E. O., 2003, “Neural-Network-Based Parameter Estimation in S-System Models of Biological Networks,” *Genome Informatics*, **14**, pp. 1-19.
- [3] Almeida, J. S., and Voit, E. O., 2003, “Neural-Network-Based Parameter Estimation in S-System Models of Biological Networks,” *Genome Informatics*, **14**, pp. 114-123.
- [4] Aluru, S., 2005, *Handbook of Computational Molecular Biology*, CRC press. Florida. Chap.27.
- [5] Badallo, M. R., 2013, “Implementation and Testing of Two Bee-Based Algorithms in Finite Element Model Updating,” Master Thesis, KTH Royal Institute of Technology.
- [6] Borelli, F. F., de Camargo, R. Y., Martins, D. C., and Rozante, L. C., 2013, “Gene Regulatory Networks Inference Using a Multi-GPU Exhaustive Search Algorithm. *BMC Bioinformatics*, **14**(Suppl 18), S5.
- [7] Chang, Y. H., Gray, J. W., and Tomlin, C. J., 2014, “Exact Reconstruction of Gene Regulatory Networks Using Compressive Sensing,” *BMC Bioinformatics*, **15**(1), pp. 400-422.
- [8] Chowdhury, A. R., Chetty, M., and Vinh, N. X., 2013, “Incorporating Time-Delays in S-System Model for Reverse Engineering Genetic Networks. *Bioinformatics*”, **14**(1), pp. 1-22.

## REFERENCES

- [9] Crombach, A., Wotton, K. R., Cicin-Sain, D., Ashyraliyev, M., and Jaeger, J., 2012, “Efficient reverse-engineering of a developmental gene regulatory network,” *PLoS Comput Biol*, **8**(7), pp. e1002589- e1002589.
- [10] Gonzalez, O. R., Küper, C., Jung, K., Naval, P. C., and Mendoza, E., 2007, “Parameter Estimation Using Simulated Annealing for S-system Models of Biochemical Networks,” *Bioinformatics*, **23**(4), pp. 480-486.
- [11] Handley, D., 2002, “Evaluating Machine Learning Algorithms Used to Infer Gene Regulatory Network Structure,” Doctoral Dissertation, Carnegie Mellon University.
- [12] Kabir, M., Noman, N. and Iba, H., 2010, “Reverse Engineering gene regulatory network from microarray data using linear time-variant model,” *BMC Bioinformatics*, **11**(Suppl 1), S56.
- [13] Karaboga, D., and Basturk, B., 2008, “On the Performance of Artificial Bee Colony (ABC) Algorithm,” *Applied Soft Computing*, **8**(1), pp.687-697.
- [14] Karaboga, D., Gorkemli, B., Ozturk, C., & Karaboga, N., 2003, “A Comprehensive Survey: Artificial Bee Colony (ABC) Algorithm and Applications.” *Artificial Intelligence Review*, **42**(1), pp.21-57.
- [15] Kikuchi, S., Tominaga, D., Arita, M., Takahashi, K., & Tomita, M., 2003, “Dynamic Modeling of Genetic Networks Using Genetic Algorithm and S-System,” *Bioinformatics*, **19**(5), pp. 643-650.
- [16] Kimura, S., Ide, K., Kashihara, A., Kano, M., Hatakeyama, M., Masui, R., Nakagawa, N., Yokoyama, S., Kuramitsu, S. and Konagaya, A., 2005, “Inference

## REFERENCES

- of S-system Models of Genetic Networks Using a Cooperative Coevolutionary Algorithm,” *Bioinformatics*, **21**(7), pp. 1154-1163.
- [17] Konak, A., Coit, D. W., & Smith, A. E., 2006, “Multi-objective optimization using genetic algorithms: A tutorial.” *Reliability Engineering & System Safety*, **91**(9), pp. 992-1007.
- [18] Kumar,S., Sharma.V.K., and Kumari,R., 2014, “Randomized Memetic Artificial Bee Colony Algorithm,” *International Journal of Emerging Trends & Technology in Computer Science*, **3**(1), pp. 52-62.
- [19] Kutalik,Z.,Tucker,W. and Moulton ,V., 2007, “S-system Parameter Astimation for Noisy Metabolic Profiles Using Newton-Flow Analysis,” *IET Systems Biology*, **1**(3), pp. 174-180.
- [20] Liang, S., Fuhrman, S., and Somogyi, R., 1998, “Reveal, a General Reverse Engineering Algorithm for Inference of Genetic Network Architectures,” *Pacific Symposium on Biocomputing*, **3**(3), pp. 18-29.
- [21] Liu, L. Z., Wu, F. X., and Zhang, W. J., 2014, “Estimating Parameters of S-systems by an Auxiliary Function Guided Coordinate Descent Method,” *Systems Science & Control Engineering: An Open Access Journal*, **2**(1), pp. 125-134.
- [22] Liu,P.-K. and Wang, F.S., 2008, “Inference of Biochemical Network Models in S-System Using Multi-Objective Optimization Approach,” *Bioinformatics*, **24**(8), pp.1085-1092.
- [23] Lobovac,Z. and Olsson,B. 2003, “Towards Reverse Engineering of Genetic Regulatory Network,” *Mater Thesis*, University of Skovde.
- [24] Margolin, A. A., Nemenman, I., Basso, K., Wiggins, C., Stolovitzky, G., Favera, R. D., and Califano, A., 2006, “ARACNE: an Algorithm for the Reconstruction of Gene

## REFERENCES

- Regulatory Networks in a Mammalian Cellular Context,” *BMC Bioinformatics*, **7**(Suppl 1), pp. 1-7.
- [25] Ng, K. M., 2006, “A Graphical Method for Automatic Code Generation from Extended S-System Petri Net Models,” Master Dissertation, Kolej Universiti Teknologi Tun Hussein Onn.
- [26] Noman, N., Palafox, L., and Iba, H., 2013, “Reconstruction of gene regulatory networks from gene expression data using decoupled recurrent neural network model,” *Natural Computing and Beyond*, pp. 93-103.
- [27] Omkar, S. N., Senthilnath, J., Khandelwal, R., Naik, G. N., & Gopalakrishnan, S., 2011, “Artificial Bee Colony (ABC) for Multi-Objective Design Optimization of Composite Structures,” *Applied Soft Computing*, **11**(1), pp. 489-499.
- [28] Savageau, M. A., 1988, “Introduction to S-systems and the underlying power-law formalism,” *Mathematical and Computer Modelling*, **11**, pp. 546-551.
- [29] Savageau, M. A., and Voit, E. O., 1987, “Recasting Nonlinear Differential Equations as S-Systems: a Canonical Nonlinear Form,” *Mathematical Biosciences*, **87**(1), pp. 83-115.
- [30] Sorribas, A., and González, A., 1999, “The Power-Law Formalism as a Tool for Modeling Hormonal Systems,” *Computational and Mathematical Methods in Medicine*, **2**(1), pp.19-38.
- [31] Tegner, J., Yeung, M. S., Hasty, J., and Collins, J. J., 2003, “Reverse Engineering Gene Networks: Integrating Genetic Perturbations with Dynamical Modeling,” *Proceedings of the National Academy of Sciences*, **100**(10), pp. 5944-5949.
- [32] Vilela, M. Chou, I.C., Vinga, S., Vasconcelos, A.T.R., Voit, E.O. and Almeida, J.S. “Parameter Optimization in S-System Models,” *BMC Systems biology*. **2**(1), p.35.

## REFERENCES

- [33] Vohradsky, J., 2001, "Neural Network Model of Gene Expression," The FASEB Journal, **15**(3), pp. 846-854.
- [34] Vrugt, J. A., Gupta, H. V., Bastidas, L. A., Bouten, W., & Sorooshian, S. (2003). Effective and Efficient Algorithm for Multi-objective Optimization of Hydrologic Models. Water Resources Research, **39**(8), pp.1-19.
- [35] Werhli, A. V., Grzegorzczuk, M., & Husmeier, D., 2006, "Comparative Evaluation of Reverse Engineering Gene Regulatory Networks with Relevance Networks, Graphical Gaussian Models and Bayesian Networks," BMC Bioinformatics, **22**(20), pp. 2523-2531.
- [36] Xinyi, Y., Dent, J. E., & Nardini, C., SPEM (S-system Parameter Estimation Method) Vignette.
- [37] Yoon, M. U., 2010, "Differential Equation Models and Numerical Methods for Reverse Engineering Genetic Regulatory Networks," Ph.D. Dissertation, University of Tennessee.
- [38] Zitzler, E., Deb, K., and Thiele, L., 2000, "Comparison of Multi-Objective Evolutionary Algorithms: Empirical Results," Evolutionary Computation, **8**(2), pp. 173-195.