UNIVERSITAT POLITÈCNICA DE VALÈNCIA

ESCUELA TÉCNICA SUPERIOR INGENIEROS INDUSTRIALES VALENCIA

TRABAJO FIN DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES

Design and Characterization of a Robust Incoherent Feedforward Synthetic Genetic Circuit

AUTORA: ALEJANDRA GONZÁLEZ BOSCÁ

TUTOR: JESÚS ANDRÉS PICÓ MARCO

COTUTORES: YADIRA FERNANDA BOADA ACOSTA
ALEJANDRO VIGNONI

Curso Académico: 2013-14

To my parents and my brother.

I would like to thank to:

Prof. Jesús Picó for trusting me to do this work and his enthusiastic direction with the ongoing desire to learn and teach.

Eng. Yadira Boada who has worked with me during this project development in a climate of comradeship and has become my friend.

Dr. Alejandro Vignoni for his highly appreciated help and his friendly and close spirit.

The whole research group GCSC of the Instituto Universitario ai2 *for giving me support and encouragement.*

Manuel Álvarez for his affection and care.

And my parents for their endless strength and allowing me the chance to make progress both in personal and learning life.

Contents

List of Figures

Part I

MEMORY

Table of contents

Summary

The goal of this work is to derive design principles for the implementation of robust incoherent feedforward (IFF) synthetic genetic circuits.

These class of circuits are ubiquitous in biological gene regulation networks (GRN). They allow the organisms to present adaptive behavior, or adaptation for short. This behavior is generally related to the so-called homeostasis capability in living organisms. Thus, adaptation consists of the circuit capability to respond to an input stimulus and return to its original value even when the input change persists. Notice this aception of adaptation is different from the one appearing in other branches of engineering. The biological adaptive IFF GRN is to some extent an analogous to a positive flank detector in electronics.

In synthetic biology, feedforward genetic circuits can be used as pulse generator and response accelerator. Furthermore it is theoretically demonstrated that fold-change detection can be generated by this topology, so we can obtain a response that is proportional to the fold-change in the stimulus relative to the background.

Tough the general principles behind the behavior of feedforward gene regulation circuits are already well-known, their actual implementation to achieve the desired performance is still challenging. Studies in the literature either implement a network and analyse the performance *a posteriori*, or deal with very simplified non realistic computational models.

In this thesis a realistic biochemical first principles model is first defined. Then, the model is reduced using both time-scale separation, and existence of invariant moieties. A multi-objective optimization approach is used to obtain the Pareto-optimal solutions in the circuit parameters space that make the circuit to achieve robust adaptation. Monte-Carlo sampling is also used to asses on the degradation of circuit performance outside the Pareto front.

Using all this information, design principles are tried to infer in order to be able to offer new tools for the systematic design of genetic synthetic incoherent feedforward circuits with pre-established adaptive response.

Next, these sets of optimal model parameters values are compared against the biologically achievable values to check the feasibility of implementation, and tuning rules using biological tuning knobs are proposed. Finally, in order to show the applicability of this work, a biological prototyping has been done.

Chapter 1

Introduction.

1.1 ANTECEDENTS AND MOTIVATIONS

Synthetic biology (SB) is one of the most cutting-edge fields in Biotechnology nowadays. Combining the knowledge from Genetic Engineering, Metabolic Engineering and Systems Biology, Synthetic Biology (SB) is able to see biological systems as a composition of pieces that may be altered, removed or exchanged between different systems to obtain a relevant organism (nonexistent without engineering) which reports benefits to society. SB is area that combines biology and engineering working together on designing and building biological devices and systems so as to achieve a particular purpose. It uses overlapping technologies from many fields and disciplines and shares their methodologies to come up with novel biomolecular components and networks to reprogram living organisms.

This can result in a big change in the way of living over the coming years, helping in many matters such as targeted therapies for attaching 'super-bugs' and diseases, or leading to cheaper drugs and vaccines. But even outside biomedicine, synthetic biology has its applicability in several areas. In the field of Energy, this can provide 'green' means to supply cars, custom-built microbes for generating hydrogen and other fuels, or for performing artificial photosynthesis. Also the detection of pollutants, and their breakdown or removal from the environment, are good points to highlight its importance. Another example is the production of fine or bulk chemicals in Chemical industry, including proteins to provide an alternative to natural fibres or existing synthetic ones, and in the field of Agriculture, the novel food additives.

The greater challenge of creating self-sustaining and self-replicating artificial cells and re-engineered organisms have become in a goal of SB. To achieve this goals, modelling how synthetic genetic circuits behave and developing and incorporating individual gene sequences into DNA as a genetic 'lego' blocks are key steps.

The use of mathematical models is of paramount importance in Synthetic Biology. They

enable the study of properties that emerge from the interaction and properties of individual parts. The modelling process itself results in hypothesis to be experimentally tested, thereby iteratively producing refined models and insight about cellular mechanisms. They are useful for engineering biological systems as they contribute to our understanding of how endogenous systems are put together and work in their interactions, and hence, how cells and synthetic networks operate and then predict their behavior.

To achieve the desired biological behaviours, the designer may work with synthetic Gene Regulatory Networks (GRN). There are many classes of GRN motifs that achieve different classes of behaviours [3]. These motifs can be thought of as modular components that can be used in more complex circuits. Out of the many possibilities, feedforward circuits have received lot of attention in the last years, for the appear in large quantities in nature. One of the most oftenly encountered feedforward motif is the incoherent feedforward gene regulatory network (IFF GRN).

The IFF GRN allows the organisms to present adaptive behavior. This consists of the circuit capability to respond to an input stimulus and return to its original value even when the input change persists. Notice this aception of adaptation is different from the one appearing in other branches of engineering. The biological adaptive IFF GRN is to some extent an analogous to a positive flank detector in electronics. This thesis is centred in these kind of genetic circuits.

1.2 GOAL, PURPOSE AND SCOPE

1.2.1 Goal

The goal of this work is to derive design principles, in the sense of tuning of model parameters, for the systematic design of genetic synthetic incoherent feedforward circuits (IFF GRN) with pre-established robust adaptive response. These sets of optimal model parameters values are compared against the biologically achievable values to check the feasibility of implementation, and tuning rules using biological tuning knobs are proposed. As secondary goal, a biological prototyping of the circuit has been carried out.

1.2.2 Purpose

The purpose of the design circuit is the utilization of the IFF GRN as an essential biological element for the design of more complex synthetic biologic circuits. In synthetic biology, feedforward genetic circuits can be used as pulse generator and response accelerator. In this regard, the IFF GRN will be implemented as a module by means of standardised DNA sequences so-called *biobricks*. Biobricks can be understood as Lego-like building blocks

used to design and assemble synthetic biological circuits, which would then be incorporated into living cells to construct new biological systems, as it is concerned.

1.2.3 Scope

Though the general principles behind the behavior of feedforward gene regulation circuits are already well-known, their actual implementation to achieve the desired performance is still challenging.

In this project a realistic biochemical first principles model is first defined. In the analysis of biological phenomena, mathematical models are often reduced by means of model reduction techniques based on appropriate assumptions on time-scales separation and invariant moieties, which has to preserve all its features if a successful simplifying model is intended to be obtained. Complex systems are typically too expensive to simulate in complete detail, so there is a need to minimize the execution times when computational algorithms work out solutions for multi-scale models. Here is where the importance of reduced models lies in. Thus, the initial model is reduced using both time-scale separation, and existence of invariant moieties. A multi-objective optimization approach is then used to obtain the Pareto-optimal solutions in the circuit parameters space that make the circuit to achive robust adaptation. Monte-Carlo sampling is also used to asses on the degradation of circuit performance outside the Pareto front. Standard rules to achieve this performance by selecting the suitable values of these parameters as *tuning knobs*, are then inferred. Finally, a basic prototyping of the biological biobricks required for the actual biological implementation is done.

This work could be addressed to readers with no biological background, but with some very basics in biology. For this reason a very short summary of the *Central Dogma* of Molecular Biology and how to model *gene expression* has been given. A basic knowledge on system dynamics is required to fully understand the methodologies used in this project.

1.3 OUTLINE

Chapter 2: Normative.

Scientific instruments, materials, chemical substances, and biological species and materials are needed to effectively carry out the practical work (so called *wet-lab* work) designed and prototyped in the theoretical analysis (so called *dry-lab* work). This section contains the rules and norms to follow in the wet-lab work related to the results of this thesis.

Chapter 3: State of art.

In this chapter, an overview is given of the existing theoretical and practical results required to design the IIF GRN in the remaining of the thesis. First, an overview is given of the Central Dogma of molecular biology, that states how proteins are produced (*expressed*). The basic forms of protein expression are described: constitutive gene expression, and gene transcription regulation by activators and repressors. Then, synthetic genetic circuits, network motifs and concepts such as robustness and adaptation are adressed. Finally, the multi-objective optimization (MOO) approach is introduced for the proper understanding of the methods used in this work.

Chapter 4: Modelling an Incoherent Feed-Fordward network.

A complete biochemical model of the IFF GRN is derived, and its corresponding dynamical model based on balance equations is formulated. This first model is of large order, which implies a high computational cost for the parameters estimation process. Therefore, the dynamical model is reduced using time-scale separation and detection of invariant moieties. The reduction is made to achieve a reduced model more amenable for computational analysis, but avoiding excessive reduction that would lead to lack of biological relevance.

Chapter 5: Parameters optimization

In this chapter parameters optimization is performed to derive the set of values of the parameters in the reduced model that allow the circuit to achieve adaptive behaviour. The proposed reduced model is used to simulate the synthesis of a required protein using a Matlab code. The adaptive behaviour is specified using a set of index. This leads to a multi-objective optimization problem. A multi-objective optimization algorithm implemented in Matlab is used to get the Pareto-optimal solutions. Also a Monte-Carlo analysis is done so as to compare results and deduce structural robustness, leading to design principles for the systematical construction of adaptive genetic circuits.

Chapter 6: Prototyping.

This chapter is focused on the choice of parameters in order to create an adaptive circuit that could be implemented using actual biological promoters encountered in the *biobricks* databases.

Chapter 7: Budget.

The project budget includes an estimation of both the use of lab resources for the circuit implementation, and the cost induced for the circuit design and prototyping.

Chapter 2

Normative.

This thesis is only concerned with the so called dry-lab, i.e. the design and analysis of biological synthetic circuits by means of computational methods. Yet, the goal is to both design the genetic circuit, and prototype its biological implementation. This wet-lab implementation will imply the use of biological agents and chemical substances.

The use of biological agents and chemical substances brings different risks for human health when used, depending fundamentally on the agent nature or the substance that concerns. This means an obligation from people in charge of scientific activity sites and direct users, to have a deep knowledge in the characteristics of such risk factors, with the aim of maintaining their health.

In order to identify and analyse labour risks associated to the various operations with biologic features that are habitually done in biotechnology laboratories, and so as to know how to take action and the steps to institute for prevention and control, the following manuals and decrees serve as a source of information.

Technical guidance for evaluation and prevention of risks related to biologic agents exposure. Royal Decree 667/1997 of 12 May 1997 BOE nº124 of 24 May 1997. Ministry of Labour and social welfare & National Institute of Workplace Safety and Hygiene.

Guía técnica para la evaluación y prevención de los riesgos relacionados con la exposición a agentes biológicos. Real decreto 667/1997, de 12 de mayo BOE nº124, de 24 de mayo. Ministerio de Trabajo y Asuntos Sociales & Instituto Nacional de Seguridad e Higiene en el Trabajo.

Safety manual for biologic activities in biotechnology laboratories. UPV.

Manual de seguridad para operaciones en laboratorios de biotecnología y de tipo biológico. UPV.

Laboratory Biosafety Manual 3rd Edition. 1983 World Health Organization (WHO).

Manual de Bioseguridad en el Laboratorio 3ra Edición.bOrganización Mundial de la Salud (OMS), 1983.

To summarize the essential information for procedures that probably will take place during the implementation of the desired genetic circuit, these following sections are pointed out. It is assumed the the physical implementation of the biological circuit will take place in the Comunitat Valenciana. Thus, both the *Generalitat Valenciana* normatives, and the national Spanish ones must be considered.

1. **Laboratory as a workplace. Generalities.**

   According to what is disposed in Royal Decree 486/1997 of 14 April 1997, minimal health and safety conditions in workplaces have to be respected. Attending to the activities that will be developed, in case they take place at UPV laboratories, the labour activities that must be considered are:

   a) Docent assignment for educational work.
   b) Field research, including previous preparatory operations, maintenance service, etc.

   In the Safety Manual for Biologic Activities in biotechnology laboratories (UPV) the following aspects are mentioned in more detail. Here, only the relevant details are described.

   Order and cleanliness: Do not overload shelves and storage areas nor obstruct crossing and enclosing areas. Be careful with spilling liquids on the tables and the floor and do the disposing of waste in suitable containers. Clean and keep correctly materials and equipment after use them, and put products away in the storage areas.

   Work spaces per worker: Full height from floor to roof: 3 meters. Free surface per worker: 2 square meters. Cubic capacity (volume) not used by worker: 10 cubic meters.

   Temperature, humidity and ventilation: Thermal isolation must to be proper according to climate conditions where laboratories are located. Temperature, humidity and ventilation limits according to what is established in annex III Royal Decree 486/1997. Minimum illumination conditions according to what is established in annex IV Royal Decree 486/1997.

2. **Chemical products manipulation and storage.**

   For the correct manipulation and storage it is necessary for the user to identify the different risky compounds, according to what is disposed in the Royal Decree 363/1995 of 10 March. No dangerous compounds will be used for this work.

   The Royal Decree 99/2003 of 24 January 2003 incorporates the following definitions:

   Substances: Chemical elements and their compounds in natural state, or those obtained by means of any production procedure, including needed additives to keep stability product and contaminants that result from used technique, excluding solvents that could isolate neither having influence on the stability nor modifying the composition.

   Preparations: mixtures or solvents composed by two or more chemical substances.

3. **Safely operations in laboratories where biologic agents are manipulated.**

   In order to protect worker's health against risks from **biologic agents** exposure during the developmental activities, the Royal Decree 664/1997 of 12 May 1997 was published inside the regulatory framework of Law 31/1995 of 8 November 1995 about Prevention of Occupational Risks.

   According to aforementioned Royal Decree 664/1997 of 12 May 1997, biologic agents are defined as *microorganisms, including those genetically modified, cell cultures and human endoparasites, liable to produce any kind of infection, allergy or toxicity*.

   In turn, a microorganism is considered as *any biologic entity, cellular or not, able to replicate or transfer genetic material*. There are four types of basic microorganisms: bacteria, fungus, virus and parasites (protozoans, species of helminth, etc.). A cell culture is the result of growing in vitro cells obtained from multi-cellular organisms.

   Depending on the infection risk, the royal Decree 664/1997 clasifies biologic agents in four groups. The biologic agents needed for this work are **biologic agents from group 1**, those that are unlikely to cause an illness in the human being.

   Biologic agents more likely to produce any kind of risky illness (groups from 2 to 4 in increasing dangerousness order), can be found in the link `http://www.mtas.es/insht/legislation/biologic.htm{#}anexo2`.

   Before starting any activity which implies the manipulation of biologic agents, these have to be identified through an inventory. Control methods for biologic agents are oriented according to the aforementioned groups of biologic agents. At this respect, the control methods for the biologic agents of this work are **Control methods from group 1**.

   Some preventive measures of general character and for laboratories with **1st Control Level** (control methods from group 1) are described in the Safety Manual for Biologic

Activities in biotechnology laboratories (page 84-87). Some of these recommendations of interest for biologic agents for group 1, apart from general preventive methods, are :

- Not to use pipettes with the mouth. Use properly devices.
- Use lab coat to prevent normal clothing from contamination. Not to use lab coat out of the laboratory.
- Always use ocular protection when risk of splash exists. If it is possible, plastic material instead of glass material, so as to decrease risk of cutting.
- Decontamination of working surfaces at least one time per day and always a spillage occurs.
- All the staff has to wash their hands after manipulating infectious materials and when leaving the laboratory.

With respect to *biologic material transportation*, some preventive measures have to be taken into account:

- Samples transportation between laboratories will be done such in case of falling down, will not splash.
- Samples must be tagged or identified opportunely and will not be used for other aim.
- Samples should not be carried by hand.

With respect to *biologic samples storage*:

- Biologic samples must be put in restricted access zones, minimizing the possibility of contamination in staff and environment.
- The storage in nitrogen liquid freezers implies the use glasses and protection masks preventing from nitrogen liquid splashes. Moreover, in case of breaking equipment, container must be emptied and let the nitrogen liquid to evaporate before proceeding to its cleaning.

With respect to *waste processing*: all biologic rejects have to be decontaminated before its removal, fulfilling the rules disposed in national Law 10/1998 of 21 April 1998 about Wastes, and the autonomous Royal Decree 240/1994 of 22 November 1994, by which it is approved the 'Reglamento Regulador de la Gestion de los Residuos Sanitarios'; Order of 14 July 1997 from *Conselleria de Medio Ambiente de la C.V.*, approving the Decree 240/1994; Law 10/2000 of 12 December 2000, about *Residuos de la Comunidad Valenciana*.

Wastes from laboratories that use biologic agents are normally classified in:

- Biologic solid wastes as urban wastes.
- Special biologic solid wastes.
- No pathogenic solid wastes from microbiological cultures.
- Biologic fluid wastes.

Waste processing for these different types of wastes are described in the *Safety Manual for Biologic Activities in biotechnology laboratories* (page 70-71).

For more information, consult the normative documents described at the beginning of this chapter.

Chapter 3

State of the art

3.1 THE CENTRAL DOGMA AND GENE EXPRESSION IN MOLECULAR BIOLOGY

The Central Dogma of molecular biology was first articulated by Francis Crick in 1958. The dogma is a framework for understanding the transfer of sequence information hard-wired in DNA (genes), and it is intrinsically related to the concept of *gene expression*.

Gene expression refers to the process of producing a specific and controlled amount of gene products. These products are often proteins, which may have structural or mechanical functions like acting as enzymes that catalyse specific metabolic pathways, that in turn integrate the metabolic network, understood as the numerous different set of chemical transformations or reactions that co-exist in cells and ensure that life is sustained. Also proteins can work as receptors or transmitters in cell signaling, may form complexes that carry out reactions, or serve as transporters for other molecules.

Proteins are macromolecules consisting of one or more chains of amino acids. Each chain is a linear polymer chain of amino acids bonded together by peptide bonds, a polypeptide [21]. The sequential information is carried by biopolymers, and this fact is given in biological reactions that work at different rates of product and degradation between the three principal elements: gene, mRNA and protein.

The biopolymers that comprise DNA, RNA and amino acids are linear polymers and the sequence of their monomers effectively encodes the information whose normal flow or transfer could be described in three fundamental steps: DNA is firstly copied to DNA (DNA replication), DNA information is copied into messenger RNA or mRNA by means of a protein called RNA Polymerase (transcription), and finally proteins can be synthesized using the information in mRNA as a template with the help of ribosomes (translation).

This is basically the essence of the so- called Central Dogma in molecular biology. Let's see this in more detail, just enough to better understand in a basic way the three processes

mentioned above.

1. **DNA replication**
   Inside the cell, the DNA replicates its information in a process that involves many enzymes. DNA covers four types of nucleotides each one referenced to the nitrogenous base that they contain (Guanine, Adenine, Thymine and Cytosine) recorded using the letters G, A, T and C. DNA replication begins at specific point when the two parent strands are unwound with the help of DNA helicases, this means that this enzyme unbinds and separates a portion of DNA so the DNA double helix is broken. Then, there are two incomplete DNA strands called complementary and template strands. Therefore, single stranded DNA binding proteins attach to the unwound strands, preventing them from winding back together. Only one strand serves as a genetic information template for transcription at any given time, and the other strand is referred to as the noncoding strand. The strands are held in position, binding easily to DNA polymerase, which catalyzes the following processes along with other enzymes called DNA primase and DNA ligase until the duplication is completed.

2. **Transcription**
   Transcription is the process by which the information contained in a section of DNA is transferred to a newly assembled piece of messenger RNA (mRNA). This process is triggered by the binding of RNA Polymerase to a determined region in DNA called promoters, and is activated or inhibit by a range of promoter specific proteins called transcription factors. This binding complements sequence of DNA after the replication, in other words, the template strands is complemented and generate the messenger ribonucleic acid (mRNA), that carries the information contained in the gene. When RNA Polymerase reaches a termination sequence on the DNA template strand, transcription is terminated and the mRNA transcript and RNA Polymerase are released from the complex (see Figure 2.3d). In eukaryotic cells, the site of transcription (the cell nucleus) is usually separated from the site of translation (the cytoplasm), so the mRNA must migrate from the nucleus to the cytoplasm, where it encounters cellular bodies called ribosomes so as to start Translation process. In prokaryotic cells, which have no nuclear compartment, the process of transcription and translation may be linked together in the cytoplasm.

3. **Translation**
   The mRNA, which carries the gene's instructions, dictates the production of proteins . It is read by the ribosome as triplet codons, usually beginning with an AUG (adenine-uracil-guanine), or initiator methionine codon downstream of the ribosome binding site. Ribosomal units move along the mRNA chain converting the information encoded in triplet codons into a chain of amino acids defining the desired protein. As the amino acids are linked into the growing peptide chain, they begin folding into the

correct conformation and this new polypeptide chain is released from the ribosome as a complete protein. Translation ends with stop codon, that could be UAA, UGA, or UAG.

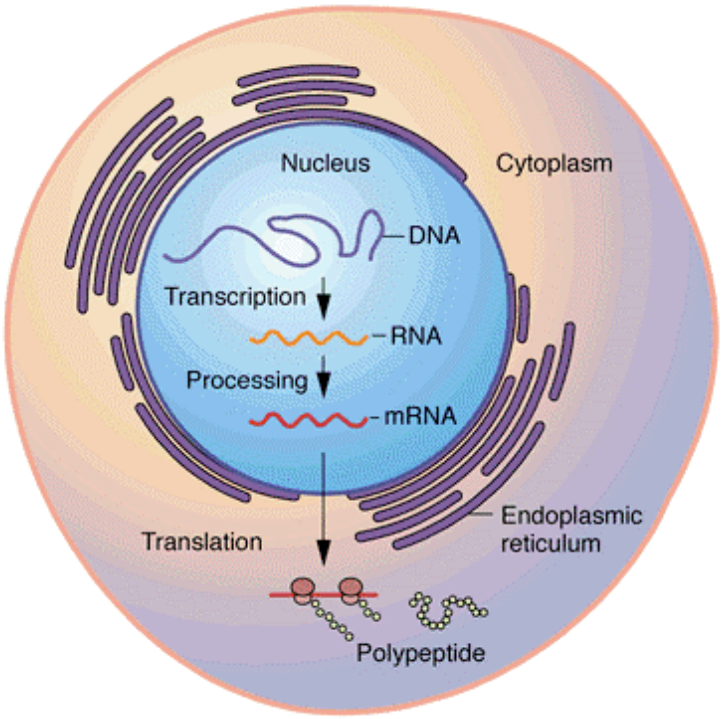

Figure 3.1: Three principal stages of Central Dogma of molecular biology: DNA replication, mRNA Transcription and protein Translation

Previously to the next section, it is interesting to appreciate the differences among the several types of *gene expression*. Attending to how it is regulated, these are some of the terms used to refer to this kind of such relevant process:

Constitutive gene expression This is non-regulated gene expression. The gene is continually transcibed.

Facultative gene expression The gene is only transcribed when needed, as opposed to the constitutive gene.

Inducible gene expression It is inherently based on regultaion of gene expression. The inducible gene is either responsive to environmental change or dependent on the position in the cell cycle. This is interesting in order to intentionally think up a synthetic circuit where the gene transcription could be regulated.

Constitutive gene expression and gene transcription regulation with inducible genes will be addressed in the next sections, as the second one deals with specifically what our work

consists on and the first one is indeed necessary to introduce how we model the most simple gene expression by mass action laws.

3.2 CHARACTERIZATION OF GENETIC COMPONENTS

3.2.1 Constitutive gene exppression

As said before, the main processes that comprise the Central Dogma can be classified as reactions that include degradation and transformation rates among the fundamental elements gen, mRNA and protein.

$$\text{Gene} \xrightarrow{Transcr} \text{mRNA} \xrightarrow{Transl} \text{Protein} \tag{3.1}$$

$$\text{mRNA} \xrightarrow{Degrad} \emptyset \tag{3.2}$$

$$\text{Protein} \xrightarrow{Degrad} \emptyset \tag{3.3}$$

It is said that the gene expression is constitutive when the gene is always 'ON'. That means that the starting point in the synthesis of the protein is not regulated by activator or repressor agents and there is not promoter to associate with in order to start the process.

This biological problem demands a model that can represent a multi-component, temporally evolving dynamic system. In these terms, differential equation models come to the fore and the regulatory networks can be represented by ordinary differential equations (ODEs). Using the law of mass action, the ODEs set for *constitutive* expression is given as:

$$\dot{m} = k_1 - d_1 m \tag{3.4}$$

$$\dot{p} = k_2 m - d_2 p \tag{3.5}$$

where: m is the mRNA concentration, p is the mRNA concentration, k_1 is the constitutive transcription rate, k_2 is the translation rate, d_1 is the mRNA degradation rate and d_2 is the protein degradation rate.

- k_1: it is considered to be constant and it represents the number of mRNA molecules produced per gene, per unit of time. In this case, k_1 is for only one copy of the gene in the cell. If there were several copies (e.g. plasmid located gene) k_1 must be multiplied by the copy number C_n to obtain the total transcription rate.
- d_1: the typical half-time for mRNA in *E. coli*, the value is between $[2,8]$ minutes (min) and average value is 5 min.

- k_2: it is considered to be constant and it represents the number of protein molecules produced per mRNA molecule, per unit of time.
- d_2: it is formed by two terms: i) the first term corresponds to the tendency of the protein to break down per unit of time, ii) the second term called the dilution term corresponds to the variation of the cell volume (through cell expansion and division) per unit of time. Typically in *E. coli*. The degradation rate is $d_2 = \frac{ln(2)}{\tau}$, where τ is the cell cycle duration between $(20, 45)$ minutes.

Few genes have constitutive expression. In most cases their expression is controlled by some external signals (DNA-binding proteins called transcriptions factors, metabolites, temperature, etc.) as discussed in the next section.

3.2.2 Gene transcription regulation

The importance of this work resides in the control of the transcription of genes, i.e. the control of the amount and timing of appearance of the functional product of a gene. This can be done through certain proteins that regulate gene expression in response to a variety of stimulus, like growth factors, stress or bacterial and viral infections, so in turn, these proteins control a number of cellular processes giving cells the flexibility to adapt to a variable environment, external signals, damage to the cell, etc.

Transcription regulation proteins are called *Transcription factors*. Each active transciption factor can bind a regulatory region of DNA that precedes the gene (promoter) to regulate the rate at which this specific target gene is read (transcribed). For this purpose, they bind determined sections of promoter called *transcription factors binding sites*. The quality of this site specifies the transcription rate of the gene. According to if these transcription factors inhibit or activate the gene expression when bound to DNA, they are given the name of repressors or activators respectively. Each edge in the network has a sign: + for activation, - for repression. Transcription networks often show comparable numbers of plus and minus edges, with more positive (activation) interactions that negative interactions.

The strength of the effect of a transcription factor on the transcription rate of its target gene is described by an **input function**. Let us consider the production of protein Y controlled by a single transcription factor X. When X regulates Y, the number of molecules of protein Y produced per unit time is a function of the concentration of X in its active form, X^*:

rate of production of $\mathbf{Y} = \mathbf{f}(\mathbf{X}^*)$

Tipically, the input function $\mathbf{f}(\mathbf{X}^*)$ is a monotonic, S-shaped function. It is an increasing function when X is an activator and a decreasing function when X is a repressor. A useful function that describes many real gene input function is the so-called *Hill function*. In

that sense, *Inducible gene expression* is often modeled through the Hill function. The Hill nonlinear equation was first introduced by A.V. Hill to describe the equilibrium relationship between oxygen tension and the saturation of hemoglobin. The Hill coefficient is commonly used to estimate the number of ligand molecules that are required to bind to a receptor to produce a functional effect. Sometimes several substrates (i.e., transcriptional factors) are needed to bind the enzyme (i.e., DNA) for the reaction to take place. In this case, this reaction is said to be *cooperative* and its model is:

$$\mathrm{E} + \mathrm{nS} \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} \mathrm{ES} \overset{k_2}{\rightarrow} \mathrm{E} + \mathrm{P}$$

where E is the free enzyme, S is the substrate, P is the reaction product, ES is the enzyme substrate complex and n is the cooperativity coefficient (see Figure 3.2). This mechanism illustrates the binding of substrate S and release of product P.

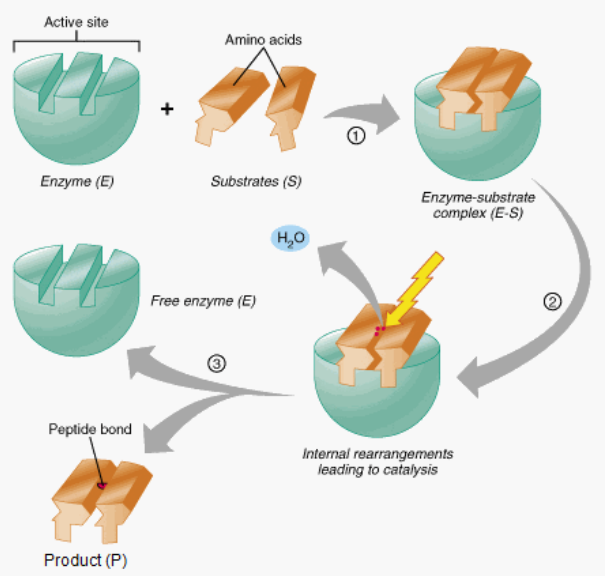

Figure 3.2: Enzyme cycle consists of (1) enzyme and substrates are free, (2) substrate binds to enzyme and form the ES complex, (3) ES places stress on the bond and (4) products are released and the enzyme is free to bind other substrates.

The Hill function can be derived from considering the equilibrium binding of the transcription factor to its site on the promoter, and it is defined as:

$$h(x) = \beta \frac{X^{*n}}{K_d + X^{*n}} \tag{3.6}$$

where β is the maximal transcription rate and x is the substrate.

Gene transcription regulation by activators.

Activation or **positive control**, occurs when the transcription level is **activated** by the cooperative binding of activators to the transcription factor binding site. That is, activators increase the transcription rate of the gene. In this case, the Hill function (see Figure 3.3a)) is a curve that rises from zero and approaches a maximal saturated level of product concentration,that is, its maximal expression level. The Hill function slope depends on the Hill coefficient n.

$$\text{Gene} \xrightarrow[\downarrow]{\text{Activator}} \text{mRNA} \underset{\downarrow}{\longrightarrow} \text{Protein} \underset{\downarrow}{}$$
$$\emptyset \qquad\qquad \emptyset$$

The following nonlinear ODE model is commonly used to describe activator controlled gene transcription:

$$\dot{m} = k_1 \frac{A^{*n}}{K^n + A^{*n}} - d_1 m \tag{3.7}$$

$$\dot{p} = k_2 m - d_2 p \tag{3.8}$$

where m, p and A^* are the mRNA, protein and activator concentrations respectively. K is the activation coefficient (it defines the concentration of active A to significantly activate expression). From the equation it is easy to see that half-maximal expression is reached when $\mathbf{A}^* = \mathbf{K}$. The value of K is related to the chemical affinity between A and its site on the promoter, as well as additional factors. k_1 is the maximal transcription rate or *maximal expression level* of the promoter. Maximal expression is reached at high activator concentrations, $A^* >> K$, because at high concentrations, A^* binds the promoter with high probability and stimulates RNAp to produce many mRNAs per unit time. n is the *Hill* coefficient (number of activators that need to bind the promoter to trigger the activation of gene expression). This coefficient governs the stepness of the input funcion. The larger is n, the more step-like the input function. Tipically, input functions are moderately steep, with n = 1 - 4.

As many functions in biology do, the Hill function approaches a limiting value at high levels of A^*, rather than increasing indefinitely. This saturation of the Hill function at high A^* concentration is fundamentally due to the fact that the probability that the activator binds the promoter cannot exceed 1, no matter how high the concentration of A^*.

Gene transcription regulation by repressors.

Repression, or **negative control**, occurs when the transcription is **repressed** by the cooperative binding of repressors to the transcription factor binding site. That is, repressors reduce the transcription rate of the gene. The Hill function in this case (see Figure 3.3b)) de-

creases from its maximal level of product concentration to the lowest level of concentration.

$$\text{Gene} \xrightarrow{\overset{\text{Repressor}}{\perp}} \text{mRNA} \longrightarrow \text{Protein}, \quad \text{mRNA} \downarrow \emptyset, \quad \text{Protein} \downarrow \emptyset$$

In the same way, the following ODE model is commonly used to describe repressor controlled gene transcription:

$$\dot{m} = k_1 \frac{K^n}{K^n + R^{*n}} - d_1 m \tag{3.9}$$

$$\dot{p} = k_2 m - d_2 p \tag{3.10}$$

where m, p and R^* are the mRNA, protein and repressor concentrations respectively, k_1 is the maximum transcription rate, K is the repression coefficient, n is the *Hill* coefficient (number of repressors that need to bind the promoter to trigger the inhibition of gene expression).

Since a repressor allows strong transcription of a gene only when it is not bound to the promoter, this function can be derived by considering the probability that the promoter is unbound by R^*. The maximal production rate is obtained when the repressor does not bind the promoter at all, that is, when $R^* = 0$. Half-maximal expression is reached when the repressor activity is equal to K.

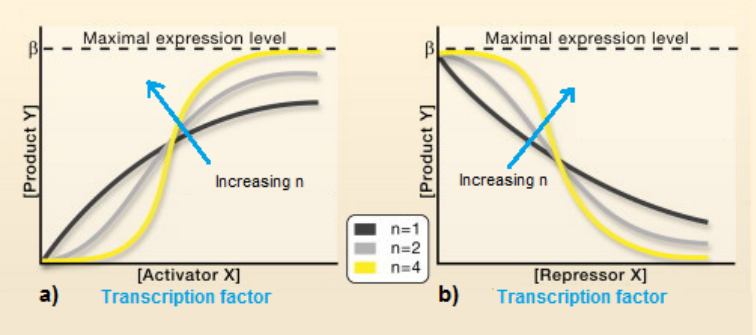

Figure 3.3: Hill function forms for transcription factors as (a) activators and (b) repressors.

Hence, each edge in the network can be thought to carry at least three numbers, β, K and n, additionally to the edge sign. These numbers can readily be tuned during evolution. K can be changed by mutations that alter the DNA sequence of the binding site of X in the promoter of gene Y, even a change of a single DNA letter. Variations in the position of the binding site or changes in sequences outside of them can strengthen or weaken the chemical bonds between X and the DNA. Similarly, the maximal activity β can be tuned by mutations in the RNAp binding site or many other factors [4].

Although the input functions described here range from a transcription rate of zero to maximal transcription rate β, many genes have nonzero minimal expression level. This is

called the gene *basal expression level*, which can be described by simply adding to the input function a term β_o.

When several transcription factors affect a gene, interacting each other in a nonlinear way, new behaviours occur. These behaviours can be expressed as combinations of the previous ones.

3.3 SYNTHETIC GENETIC CIRCUITS

3.3.1 Introduction

Cells live in a complex environment and can sense many different signals, including physical parameters such as temperature and osmotic pressure, biological signaling molecules from other cells, beneficial nutrients, and harmful chemicals. Cells respond to these signals by producing appropriate proteins that act upon the internal or external environment.

Synthetic genetic circuits can potentially result in more efficient pathways that would permit to program living cells for advanced applications. In this sense, engineers seek to harness cell's capability of initiating gene expression in response to specific signals, to program them to perform tasks or create chemicals and materials that match the complexity seen in nature and provide with tools that aid the construction of genetic circuits [13].

Genetic regulatory circuits are functional clusters of genes that could have an effect on each other's expression through inducible transcription factors and cis-regulatory elements CREs (regions of non-coding DNA which regulate the transcription of nearby genes). These circuits can be modeled and performed in silico to predict the dynamics of a genetic system. Furthermore as we have seen, circuit dynamics can be influenced by the choice of regulators and changed with expression tuning knobs. The inputs to the network are signals that carry information from the environment. Each signal is a small molecule, protein modification, or molecular partner that directly affects the activity of one of the transcription factors. The cross regulation of genes can be represented by a graph, where genes are the nodes and one node is linked to another if the former is a transcription factor for the latter. See an example in Figure 3.4.

Genetic regulatory circuits are analogous in many ways to electronic circuits. They are a network integrated by several interconnected components with tuning knobs and at least one closed trajectory. Similarly, the whole serve as a mean to develop a certain useful function, that in terms of synthetic biology, it would be novel biological functions. For most genetic circuits, a sufficient degree of cooperativity in their circuit components is required.

These networks are nonlinear and then, they require design and analysis tools more complex than those used for linear problems.

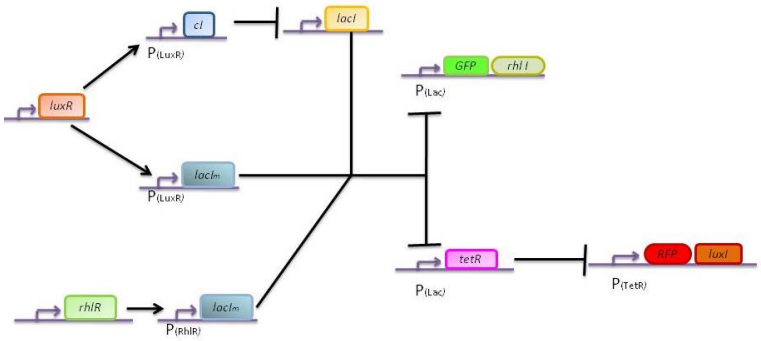

Figure 3.4: Graph example of a seven-node genetic circuit. The species luxR and rhlR are input proteins that act as transcription factors activating the gene expression of cI, lacIm, (normal arrow means activation). The production of these proteins has a repressive effect on the next genes, and in turn, a third stage of repression is derived (arrow with perpendicular termination means repression).

Moreover, genetic circuits have no signal isolation. Biological systems are constructed from very noisy devices. Circuit products may interfere with each other and the host cell, so circuits behavior is non-deterministic in nature. That means that they are inherently stochastic. In fact, we can find two kind of noise source in gene expression, intrinsic and extrinsic.

On one hand, the so-called intrinsic noise arises due to the stochastic fluctuations in the transcription and translation steps of gene expression. On the other hand, gene expression is subject to variability arising from fluctuations originating from the environment, i.e. from other cell components upstream of the system of interest. This is the so-called extrinsic noise [21].

Here is when the system robustness plays an important role. A robust control design of a system must assume that there will exist an error or uncertainty between mathematical model and reality. Robust control systems take into account this approximation or assumption so that the specified behavior is fulfilled when perturbations on the system are given. That means in biological terms, that robust designs in genetic circuits its essential function is nearly independent of biochemical parameters that tend to vary from cell to cell, even if the cells are genetically identical.

3.3.2 Network motifs

A biological transcription network is made of many interactions edges, making it very complex. Little is known about the design principles of transcriptional regulation networks that control gene expression in cells, so the question is if it is possible to define understandable patterns of interconnections that serve as building blocks so that we can understand the dynamics of the entire network based on the dynamics of these units.

Recent advances in data collection and analysis are generating unprecedented amounts of information about gene regulation networks, and they are effectively showing that do exist these building-block patterns or recurring circuit modules inside networks at frequencies much higher than those found in randomized networks. They are called network motifs [26].

For instance, the transcription networks of the bacterium *Escherichia coli* [25, 20] and the yeast Saccharomyces cerevisiae [20, 16] were found to contain the same small set of highly significant motifs. The significance of these structures raised the question of whether they have specific information-processing roles in the network, and since they do, they have been used to understand the network dynamics in terms of elementary computational building blocks [17].

Thus, simplest network motifs have been examined as they have resulted to be useful to obtain certain kind of behavior. Each network motif has a specific function in determining gene expression, such as generating temporal expression programs or velocity response, and governing the responses to fluctuating external signals (stability).

Some examples of relevant network motifs [2] are given below:

1. **Negative autoregulation**

   Negative autoregulation (NAR) occurs when a transcription factor represses the transcription of its own gene (Figure 3.5). This network motif occurs in about half of the repressors in *E. coli* and in many eukaryotic repressors.

Figure 3.5: Negative autoregulation of gene X by repression of its own promoter

   NAR has been shown to display two important functions:
   (1) NAR speeds up the response time of gene circuits.
   (2) NAR can reduce cell-cell variation in protein levels.

2. **Positive autoregulation**

   Positive autoregulation (PAR) occurs when a transcription factor enhances its own rate of production (Figure 3.6)

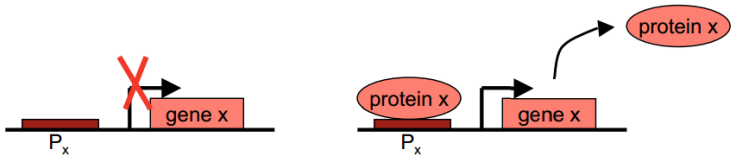

Figure 3.6: Positive autoregulation of gene X by activation of its own promoter

The properties of PAR are opposite to those of NAR:
(1) PAR slows the response time.
(2) PAR tends to increase cell-cell variability.

Slowed down response time increase fluctuation due to noise and can induce bistability.

3. **Feedforward loop**

The second family of network motifs is the feedforward loop (FFL). It predominantly appears in many known networks e.g. gene systems in *E. coli* and yeast, as well as in other organisms. It is a three-gene pattern, composed of two input transcription factors, one of which regulates the other, both jointly regulating a target gene. In other words, a motif in which a transcription factor X regulates a second transcription factor Y, such that both X and Y jointly regulate an operon Z (In genetics, an operon is a functioning unit of genomic DNA containing a cluster of genes under the control of a single promoter) . This motif has been shown to be a feed forward system, detecting non-temporary change of environment.

Feed-forward is a term describing an element or pathway within a control system, which reacts to changes in its environment, normally in order to keep any determinate state of the system. A control system which has only feed-forward behavior responds to its control signal in a pre-defined way without responding to how the load reacts; it is in contrast with a system that also has feedback, which adjusts the output to take account of how it affects the load, and how the load itself may vary unpredictably; the load is considered to belong to the external environment of the system. In a feed-forward system, the control variable adjustment is not error-based. Instead it is based on knowledge about the process in the form of a mathematical model of the process and knowledge about or measurements of the process disturbances [1].

A feedforward loop motif is ‘coherent’ if the direct effect of the general transcription factor on the effector operons has the same sign (negative or positive) as its net indirect effect through the specific transcription factor. For example, if X and Y both positively

regulate Z, and X positively regulates Y, the feedforward loop is coherent. If, on the other hand, X represses Y, then the motif is incoherent.

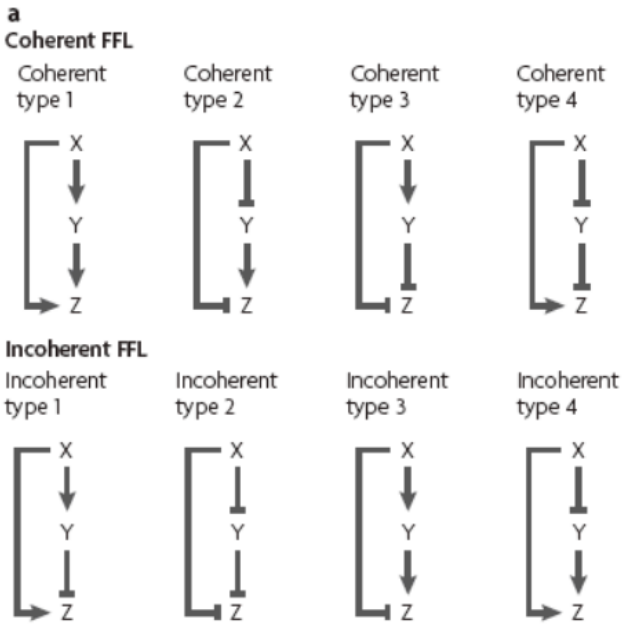

Figure 3.7: Coherent and incoherent feedfordward motifs.

The FFL has eight possible structural types, because each of the three interactions in the FFL can be activating or repressing. When theoretically analyzed the functions of these eight structural types, it was found that four of the FFL types (incoherent FFLs), act as sign-sensitive accelerators: they speed up the response time of the target gene expression following stimulus steps in one direction (e.g., off to on) but not in the other direction (on to off). The incoherent FFL mechanism can in principle apply to any gene, not only to transcription factors, because the acceleration is carried out by the two transcription factors upstream of the target gene. The other four types, coherent FFLs, act as sign-sensitive delays (create delays in response to signal changes) [17].

In the often studied transcriptional networks (*E. coli* and yeast), two of the eight FFL types occur much more frequently than the other six types. These common types are the coherent type-1 FFL (C1-FFL) and the incoherent type-1 FFL (I1-FFL). The FFL motif characterizes 40 effector operons in 22 different systems in the network database, with 10 different general transcription factors. It is found that most (85%) of the feedforward loop motifs are coherent.

There are other network motifs, such as Single-input motif (SIM) or Dense overlapping regulons (DOR).

3.3.3 Robustness and Adaptation in genetic circuits

Systems biology has revealed numerous examples of networks whose dynamic behavior is robust to system perturbations and noise. The ability of cells to extract and process information from their environment allowing them to optimize their responses, must necessarily be a robust property for it to be effective in the cell's noisy and uncertain environment. A key aim of systems biology is to identify the mechanisms through which robustness is achieved in cellular processes. Such sources of robustness can be identified through the analysis of models of biological systems [11].

Robustness is considered to be a fundamental feature of complex evolvable systems. It is attained by several underlying principles that are universal to both biological organisms and sophisticated engineering systems. Robust traits are often selected by evolution, and insights in specific architectural features observed in robust systems can provide us with a better understanding of this natural property of biological systems [12].

This capability was suggested to be an important design principle by M. Savageau in theoretical analysis of gene circuits. Several studies about this principle could be mentioned, for instance, robustness of metabolic fluxes with respect to variations of enzyme levels in yeast, that was experimentally demonstrated (Kacser and Burns, 1973). But even before robustness had been studied in a different context, the sensitivity of developmental patterning of tissues as an egg develop into an animal to various perturbations [3].

Also, the many studies about Bacterial Chemotaxis, (known as the process in which bacteria sense and move along gradients of specific chemicals), has brought about principles of robustness that can give help to rule out a large family of plausible mechanism and to home in on the correct design [3].

In bacterial chemotaxis, changes in concentration of a substrate in the surrounding media influence propulsion activity of the cell, allowing it to move to the location of the highest (attractor)/lowest (repellent) concentration. The basic features of the chemotaxis response can be described by a process that is called adaptation, a process by which the response to an extracellular stimulus returns to its pre-stimulus value even in the continued presence of the signal. In other words, the ability of a system to compensate for changes in its environment.

This is common to many biological sensory systems. We can see it in all homeostasis systems. Homeostasis is a property present in living organisms that consist of its ability to keep a stable internal condition, trading off environmental changes by means of the regulated exchange of resources and energy with the outside. This is possible because of the existence of feedback control systems network that constitute the autoregulation mechanisms in cells.

But even not going beyond it, **sensory adaptation** occurs in all body senses as well (with the possible exception of the sense of pain), when sensory receptors change their sensitivity to the stimulus. In the visual system , *dark adaptation* and *light adaptation*, understood

as adaptation to changes in light intensity, involves an immediate change in pupil size (it becomes smaller or larger, admitting less or more light), and a change in the sensitivity of the cones and rods to light (it decreases or increases). Other examples of sensory adaptation are given in the *hearing system*: as a protective mechanism, loud sound causes a small muscle attached to one of the bones of the inner ear to contract, reducing the transmission of sound vibrations to the inner ear, where the vibrations are detected. Or in the *touch system*: as we quickly adapt to hot and cold stimulation, if it is not too intense. The bath that was almost too hot to enter soon feels too cool; similarly, the cold lake we jump into for a summer swim feels freezing at first, but soon feels only refreshingly cool. Also *smell system*: we can detect amazingly low concentrations of some chemicals in the air (e.g. perfumes) but although the perfume is still in the air about us, we quickly cease to detect it.

Moreover, the proper function of many biological systems requires that external perturbations also be detected, allowing the system to adapt to these environmental changes, so *sensing* changes in the input signals may be equally important for achieving proper cell function [7].

Some ideas taken from [27] and [11] suggest that these kinds of behaviours (robustness, adaptation, signal detection...) arise from simple yet fundamental features of the system architecture. Under suitable technical assumptions, if a system adapts to a class of external input signals which belong to a predetermined class of time-functions, in the sense of regulation against disturbances or tracking signals, then the system must necessarily contain a subsystem which is itself capable of generating all the signals. In brief, adaptation and regulation with signal detection implies internal model (internal model principle IMP) [27].

Once explained through some examples what the process of adaptation means, a definition with a mathematical approach will be given: A system shows adaptation when certain quantity y(t) associated to the system, called its output (also called a regulated variable or an error) has the property that $y(t) \to 0$ as $t \to \infty$ whenever the system is subject to an input signal from a certain class. Of course, the choice of y=0 as the 'adaptation value' is merely a matter of convention, considering relative changes in response [27].

Adaptation in biology can be easily related to what is known in engineering electronic circuits as an **edge detector**. Knowing about the existence of electronic modules with detection functions for a given transition from low level to high level in the input, and then producing a pulse signal for this input change, an adaptive genetic circuit can be also considered as a biologic module to detect changes in environmental molecules concentrations, and respond with the synthesis of other ones in the same way. This can be useful, for instance, for starting other vitally important sequence of mechanisms in living organisms.

Now, introducing a mathematical description of adaptation (depicted in Figure 3.8) through the two characterizing adaptation terms defined in [29] will facilitate the analysis of this phenomenon and will be taken into account in the next descriptions in this project. These

two characteristic quantities are the circuit's sensitivity to input change and the precision of adaptation. Adaptation is understood as precision as the difference between the pre-stimulus and post-stimulus steady states, that could be defined as the inverse of the relative error.

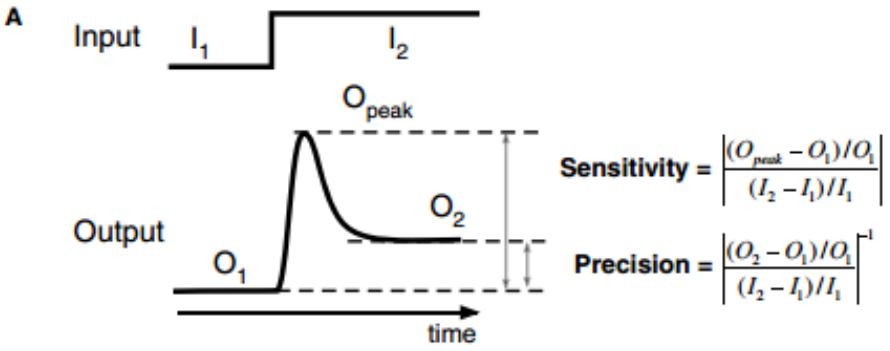

Figure 3.8: Indices for the system output for a step input [29]. Sensitivity is related to the pick height and precision is related to the error.

If the system's response returns exactly to the pre-stimulus level (infinite precision), it is called exact or perfect adaptation. Perfect adaptation range from the chemotaxis of bacteria (Bergand Brown, 1972; Macnab and Koshland, 1972; Kirsch et al., 1993; Barkai and Leibler, 1997; Yi et al., 2000; Mello and Tu, 2003; Raoet al., 2004; Kollmann et al., 005; Endres and Wingreen, 2006), amoeba (Parent and Devreotes, 1999; Yang and Iglesias, 2006), and neutrophils (Levchenko and Iglesias, 2002), osmo-response in yeast (Mettetal et al., 2008), to the sensor cells in higher organisms (Reisert and Matthews, 2001; Matthews and Reisert, 2003), and calcium homeostasis in mammals (El-Samad et al., 2002).

3.3.4 Adaptative and robust topologies

As it has been shown in 3.3.2, a motif which predominantly appears in many known networks is de IFFL and seems to play an important role. Thus, in [29], the authors computationally searched all possible three-node enzyme network topologies to identify those that could perform adaptation. Only two major core topologies emerge as robust solutions: a negative feedback loop with a buffering node and an **incoherent feed-forward loop** with a proportional node. The idea was to identify network topologies are capable of robust adaptation and then, by means of analysis techniques (circuit-function map form which it could be extracted core topological motifs essential for adaptation), revealing the existence of design principles in order to robustly engineer biological circuits that carry out a target function.

For this analysis, the authors focused on enzymatic regulatory networks, modelling network linkages by Michaelis-Menten rate equations. They started from examining the simplest networks capable of achieving adaptation. For networks composed of only two nodes (an input receiving node A and output transmitting node C, with no third regulatory node), none of the 81 possible networks with the 4 possible links, was capable of achieving adaptation for the parameter space that was scanned. Next, minimal three-node topologies with only three or fewer links between nodes were examined. The maximal complex three-node topologies contain nine links. None of the two-link, three-node networks were capable of adaptation. The minimal number of links for this to be functional seems to be three. After that, three nodes were used as a minimal framework, including one node that receives input, a second node that transmits output, and a third node that can play diverse regulatory roles. Then their adaptation properties over a range of kinetic parameters were studied. According to [29], the simplest topologies capable of adaptation are either a single class of negative feedback loop or a single class of incoherent feed-forward loop.

Attending to the definition given in this article, a negative feedback loop is a topology whose links, starting from any node in the loop, lead back to the original node with the cumulative sign of regulatory links with in the loop being negative. As said, only one class of simple negative feedback loop can robustly achieve adaptation: they are so-called Negative Feedback Loop with a Buffer Node or NFBLB. All minimal NFBLB topologies use the same integral control mechanism for perfect adaptation. 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. The importance of this buffering node is discussed in detail in [29].

By contrast, an incoherent feedforward loop is defined as a topology in which two different links starting from the input-receiving node both end at the output-transmitting node, with the cumulative sign of the two pathways having different signs (one positive and one negative).

Among feedforward loops, coherent feedforward resulted poor at adaptation, and the rest of incoherent feedforward loops also differed drastically 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. The reason this architecture is preferred is because the only way for an incoherent feedforward loop to achieve robust adaptation is for node B to serve as a proportioner for node A.

In this work, considering that incoherent feedforward loops appear to perform adaptation more robustly than negative feedback loops according to what results show, the chosen motif for modelling, simulating and discussing results was the Incoherent Feedforward Loop with a Proportioner Node or IFFLP. This topology achieves adaptation by using a different mechanism from that of the NFBLB class. Rather than monitoring the output and feeding back to adjust its level, the feedforward 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.

The modelling of this IFF motif was done in the referenced work [29]. There the authors used Michaelis-Menten kinetic rate equations to describe in an approximate way the behaviour of the three nodes IFF motif :

$$\frac{dA}{dt} = Ik_{IA}\frac{(1-A)}{(1-A)+K_{IA}} - F_A k_{F_AA}\frac{A}{A+K_{F_AA}}$$
$$\frac{dB}{dt} = Ak_{AB}\frac{(1-B)}{(1-B)+K_{AB}} - F_B k_{F_BB}\frac{B}{B+K_{F_BB}}$$
$$\frac{dC}{dt} = Ak_{AC}\frac{(1-C)}{(1-C)+K_{AC}} - Bk_{F_BC}\frac{C}{C+K_{F_BC}}$$

Nevertheless in this work a higher accuracy it is looked for. Because of that, attending to the complete set of biochemical reactions that take place in this IFF GRN, a realistic first principles 'Complete model' will be first defined in chapter 4. Then, the model is reduced using both time-scale separation and existence of invariant moieties. A 'Reduced model' is finally deduced, which even the simplifications and approximations used, is much more realistic than the set of Michaelis-Menten kinetic equations above.

3.4 MULTI-OBJECTIVE OPTIMIZATION

Multi-objective problems (MOPs) frequently appear in control engineering designing problems. These are problems in which the designer must deal with the fulfillment of multiple objectives, and the set of different techniques used for giving solution to these problems comprise what is known as multi-objective evolutionary optimization (EMO).

The term multiple-objective optimization refers to multiple criteria decision making. Thus, it is concerned with mathematical optimization problems involving more than one objective function to be optimized simultaneously. In these kind of problems optimal decisions need to be taken in the presence of trade-offs between two or more objectives that could be conflicting. In other words, it could happen that due to the impossibility of obtaining a solution that is good for all objectives, several solutions with different trade-off levels would appear. That is, there exists a possibly infinite number or Pareto optimal solutions [19]. Given that, a solution is called to be in the Pareto Front if none of the objective functions can be improved in value without degrading some of the other objective values. In this sense, without additional subjective preference information, all Pareto optimal solutions are considered equally good. However, when it exists preference information from the designer

or decision maker (DM), multi-objective optimization techniques search for a set of potentially preferable solutions so that the designer may then analyse the trade-offs among them, and select the best solution according to his/her preferences.

Control engineering problems are generally multi-objective problems where the DM also has a role, as there are several specifications and requirements that must be fulfilled. A traditional approach for calculating a solution with the desired balance among (usually conflictive) objectives is to define an optimization statement. In [10] this design procedure based on EMO is presented and significant applications on controller tuning are discussed. This optimization approach seeks for a set of Pareto optimal solutions to approximate what is known as the Pareto set [18]. Each solution in the Pareto set defines an objective vector in the Pareto front.

In order to approximate this Pareto set, classic optimization techniques [19] and evolutionary multi-objective optimization (EMO) approaches have been used. In the latter case, multi-objective evolutionary algorithms (MOEAs) have become a valuable tool to approximate the Pareto front for non-convex, non- linear and constrained optimization instances. They have been used with success in several control systems and engineering design areas [9].

3.4.1 Multi-objective optimization design (MOOD)

Any MOO design approach must follows three main steps: problem definition, MOO process and decision making stage.

The problem will be defined in chapter 5. Fort this project a multi-objective optimization design (MOOD) procedure defined by [8] was used.

This design concept is built with a family of design alternatives (Pareto optimal solutions or Pareto Front) that are specific solutions in the design concept. Thus, a set of solutions defining the Pareto set will also give a set of solutions called Pareto front, as each solution in the Pareto set determine an objective vector in the Pareto front.

In Figure 3.9, five different solutions $\diamond$ are calculated to approximate a Pareto front (bold-line). Solutions A, B, and C are non-dominated solutions, since there are no better solution vectors (in the calculated set) for all the objectives. Solutions B and C are not Pareto optimal, since some solutions (not found in this case) dominate them. Furthermore, solution A is also Pareto optimal, since it lies on the feasible Pareto front. The set of non-dominated solutions (A, B, and C) build the Pareto front approximation.

It is important to notice that most of the times the Pareto front is unknown and it shall only to be relied on approximations.

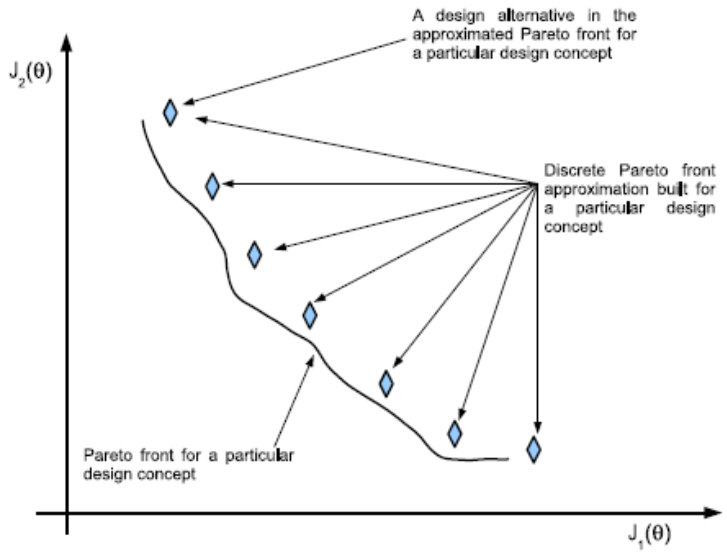

Figure 3.9: a Pareto front approximation (boldline) for particular design concept is calculated with a set of Pareto-optimal design alternatives ◇

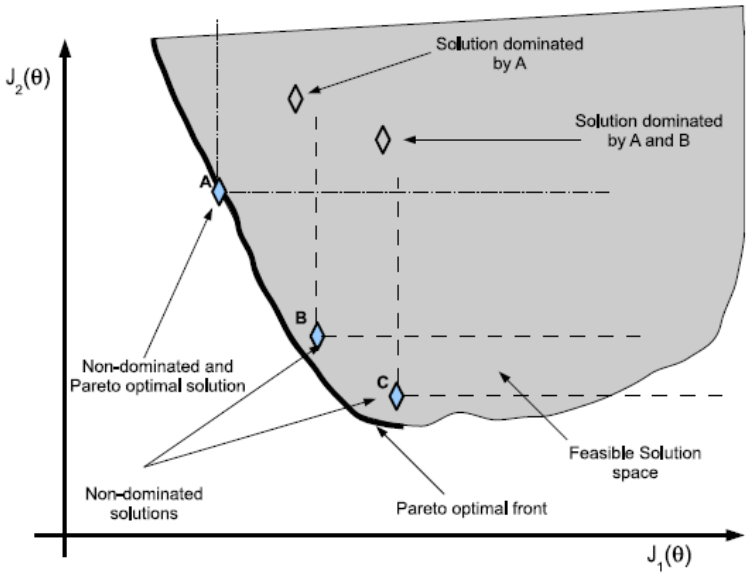

Figure 3.10: Pareto optimality and dominance concepts

Multi Objective Evolucionary Algorithm (MOEA).

When defining the multi objective problem, the selection of the optimization objectives is done for measuring the desired performance. According to the expected design alternatives, the MOEA would need to include certain mechanisms or techniques to deal with the optimization statement. Some examples are related to robust, multi-modal, dynamic and/or computationally expensive optimization.

In [8] there are some references for a comprehensive review of the early stages of MOEAs. These evolutionary/nature-inspired techniques require mechanisms to deal with EMO since they were originally used for single objective optimization.

Regarding the Pareto set sought, some desirable characteristics include (in no particular order) **convergence**, **diversity**, and **pertinency**.

Convergence refers to the algorithm's capacity to reach the real (usually unknown) Pareto front.

Diversity refers to the algorithm's capacity to obtain a set of distributed solutions that provide a useful description of objective trade-off and decision variables.

Pertinency is the capacity to obtain a set of interesting solutions from the DM point of view. Incorporating DM preferences into the MOEA has been suggested to improve the pertinency of solutions.

Regarding the optimization statement, some features could be for handling constrained, computationally expensive or large scale optimization instances. More details in [8]

Large scale optimization refers to the capabilities of a given MOEA to deal with an MOP with any number of decision variables with reasonable computational resources. Sometimes an MOEA can have remark-able convergence properties for a relatively small number of decision variables, but may be intractable (according to the computational resources available) for solving a problem with a larger number of decision variables.

Computationally expensive optimization is related to the cost function evaluation, that sometimes requires a huge amount of computational resources. Therefore, stochastic approaches could face a problem, given the complexity in evaluating the fitness (performance) of an individual (design alternative); this could affect their exploration capabilities and hence, slow down the convergence properties.

Any kind of MOO algorithm can be used in the MOO design methodology [9]. A MOEA is selected due to its flexibility to handle complex functions. It will use the performance calculated from the simulation process to evolve the population to the Pareto front. In particular, the sp-MODE algorithm is based on differential evolution technique, which is an evolutionary algorithm [8].

Multi Criteria Decision Making (MCDM).

Dealing with the multi-criteria decision making step, once the DM has been provided with a Pareto front, he or she will need to analyse the trade-off between objectives and select the best solution according to his/her preferences. It is widely accepted that visualization tools are valuable and provide the DM with a meaningful method to analyse the Pareto front and take decisions. Tools and/or methodologies are required for this final step to successfully embed the DM into the solution refinement and selection process[8].

For two-dimensional problems (and sometimes for three- dimensional problems) it is usually straight forward to make an accurate graphical analysis of the Pareto front, but the difficulty increases with the dimension of the problem. In Lotovand Miettinen (2008), visualization techniques are reviewed, including tools such as decision maps, star diagrams, value paths, GAIA, and heat map graphs. Some degree of interactivity with the visualisation tool is also desirable (during and/or before the optimization process) to successfully embed the DM into the selection process.

In [9] the LD visualization is presented and referenced. It helps to perform an analysis of the obtained Pareto front. It has been used with success in control systems up to 15 objectives, safety systems analysis, and engineering design. The LD visualization is one of the most useful methods to visualize m-dimensional Pareto fronts. The LD visualization is based on the classification of the approximation obtained. Each objective is normalized with respect to its minimum and maximum values.

To plot the LD, the LD visualization tool (LD-tool) will be used. This is *a-posteriori* visualization tool (i.e., it is used after the optimization process) that enables the DM to identify preferences zones along the Pareto front, as well as selecting and comparing solutions.

The aforementioned steps (problem definition, MOO process and the decision making stage) are important to guarantee the overall design methodology. With a poor problem definition, not matter how good our MOEA and decision making methodologies are, we will not have solutions which guarantee a good performance on the real system.

Chapter 4

Modelling an Incoherent Feed-Fordward network

4.1 INTRODUCTION TO I1-FFL

In the previous chapter 3.3.2, it was mentioned that the feedforward loop (FFL) frequently appears in well-known networks. One more specific topology of this motif that concerns this work is the incoherent feedforward loop type 1 (I1-FFL), in which an activator regulates both a gene and a repressor of the gene. See Figure 3.7

What characterizes this topology, apart from the two arms of the FFL that act in opposition, is the fact that I1-FFL is a pulse generator and response accelerator. Many sensory systems show a response that is proportional to the fold-change in the stimulus relative to the background, a feature related to Weber's Law [I]. Recent experiments suggest a response that depends on the fold-change in the input signal, and not on its absolute level. It is theoretically demonstrated that fold-change detection can be generated by the incoherent feedforward loop (I1-FFL). The fold-change detection feature of the I1-FFL applies to the entire shape of the response, including its amplitude and duration, and is valid for a wide range of biochemical parameters [15].

All these features characterizing this topology (robust adaptation and fold-change detection), make it interesting to model, simulate and analyse in order to get better tools, well-characterized parts and a comprehensive understanding of how to compose regulatory genetic circuits that can provide such abilities.

[I] Weber's Law states that the ratio of the increment threshold to the background intensity is a constant. More information in `http://www.cis.rit.edu/people/faculty/montag/vandplite/pages/chap_3/ch3p1.html`

4.2 DETERMINISTIC APPROACH OF A THREE-NODE I1-FFL

In this work, we start from scratch and make a complete model of the three-node I1-FFL and then reduce it, trying to validate the reduced model that explains the main characteristics of the genetic circuit.

The idea is to take into account just one cell (which simplifies enormously the work). Indeed cells are normally together in large populations where they can grow and divide, leading to a diffusion process and to variability in the population. These are factors also important to consider but would extend this work to the analysis of the stochastic inherent response in biologic systems.

We model the designed genetic circuit using a deterministic approach and taking into account the key regulatory interactions between the main biochemical species present in the genetic circuit: A protein, B protein, C protein and I inducer.

The proposed circuit in terms of genes and nodes can be seen in Figure 4.2. The product of gene A bound to the inducer activates gene C. Simultaneously the gene A also represses gene C by activating the repressor product of gene B. As a result, when a signal causes node A to assume its active conformation, C is produced, but after some time B accumulates, eventually attaining the repression threshold for the gene C promoter.

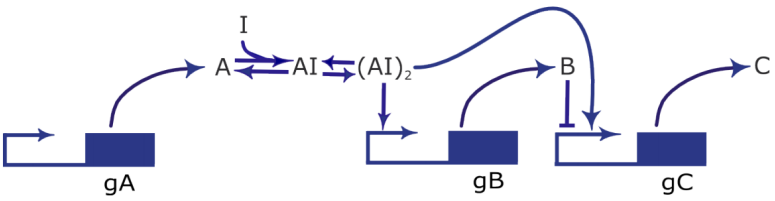

Figure 4.1: Three-node incoherent feedforward loop. gA produces the protein A, which forms a dimer with the inductor. This dimer that activates gC and gB, whose product in turn represses gC

In our gene synthetic network (see Figure 4.2), the feedforward circuit comprises a gene gC under the control of the promoter P_{gC}. The production of the protein C as a response, so that it performs robust adaptation is the aim. This expression is activated by a complex that acts as transcription factor for the promoter P_{gC}. This complex consists of a dimer that comes from the union of two monomers that in turn come from the binding of the gene product A and an inducer. The regulatory part of the circuit appears when the same complex that activated the expression of the gene C also activates B gene expression, which also acts as a transcription factor for the P_{gC} double or hybrid promoter, but in this case repressing the production of C. In other words, when A (constitutively expressed protein) is bound

to the dimer, it can prompt the activation of B and C gene expressions, but the protein B, when produced, will inhibit the transcription of the genes downstream the promoter P_{gC}. Therefore, the circuit has a feedforward loop between the concentration of the protein A and the expression of the gene gC.

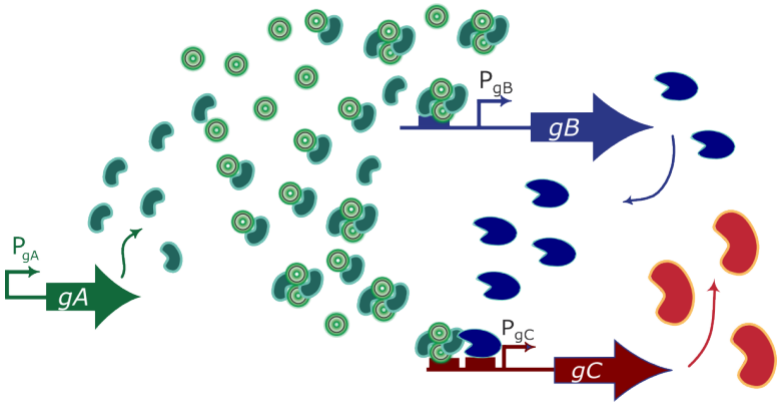

Figure 4.2: Biologic system with an incoherent feedforward loop

The modelling scheme given above can be formally written as a set of biological reactions. In the following section, a complete biochemical model of the IFF GRN is derived, and its corresponding dynamical model based on balance equations is formulated. This first model is of large order, which implies a high computational cost for the parameters estimation process that will be carried out later on. Additionally, the large differences in the time scales among the different species in the synthetic gene network (typically many orders of magnitude) create huge difficulties for simulating the temporal evolution of the network and for understanding the basic principles of its operation. Therefore, the dynamical model will be reduced using time-scale separation and detection of invariant moieties. The reduction is made to achieve a reduced model more amenable for computational analysis, but avoiding excessive reduction that would lead to lack of biological relevance.

4.3 COMPLETE MODEL

To summarize the interactions and dynamics among species involved, we can make a differentiation between the three proteins in the *gene expression block* and the *induction block*.

In the *gene expression block* the main processes considered are the binding of the RNA polymerase to the promoter, transcription, translation, mRNA degradation and protein degradation.

In the *induction block* the main processes considered are the binding between the protein A and the inducer to form monomer, the addition of external inducer, the diffusion of the inducer, its degradation, the dimer formation and its degradation, the monomer degradation, binding of the dimer to the gB promoter, binding of the dimer to the gC promoter and the binding between the activator (or repressor) and the gC hybrid promoter. The corresponding reactions.

For an individual cell, the set of biochemical reactions considered in this work are given below:

$$\mathrm{gA} + \mathrm{RNAp} \xleftrightarrow{k_1/k_{-1}} \mathrm{gA} \cdot RNAp \tag{4.1}$$

$$\mathrm{gA} \cdot RNAp \xrightarrow{k_{mA}} \mathrm{gA} + \mathrm{RNAp} + \mathrm{mA} \tag{4.2}$$

$$\mathrm{mA} \xrightarrow{k_{pA}} \mathrm{mA} + \mathrm{A} \tag{4.3}$$

$$\mathrm{mA} \xrightarrow{d_{mA}} \emptyset \tag{4.4}$$

$$\mathrm{A} \xrightarrow{d_A} \emptyset \tag{4.5}$$

$$\mathrm{A} + \mathrm{I} \xleftrightarrow{k_2/k-2} \mathrm{A} \cdot I \tag{4.6}$$

$$\emptyset \xrightarrow{k_e} \mathrm{Ie} \tag{4.7}$$

$$\mathrm{I_e} \xleftrightarrow{k_d/k_{-d}} \mathrm{I} \tag{4.8}$$

$$\mathrm{I} \xrightarrow{d_I} \emptyset \tag{4.9}$$

$$2 \cdot (\mathrm{A} \cdot I) \xleftrightarrow{k_3/k_{-3}} (\mathrm{A} \cdot I)_2 \tag{4.10}$$

$$(\mathrm{A} \cdot I) \xrightarrow{d_{AI}} \emptyset \tag{4.11}$$

$$(\mathrm{A} \cdot I)_2 \xrightarrow{d_{AI2}} \emptyset \tag{4.12}$$

$$\mathrm{gB} + (\mathrm{A} \cdot I)_2 \xleftrightarrow{k_4/k_{-4}} \mathrm{gB} \cdot (A \cdot I)_2 \tag{4.13}$$

$$\mathrm{gC} + (\mathrm{A} \cdot I)_2 \xleftrightarrow{k_5/k_{-5}} \mathrm{gC} \cdot (A \cdot I)_2 \tag{4.14}$$

$$\mathrm{gC} \cdot B + (\mathrm{A} \cdot I)_2 \xleftrightarrow{k_6/k_{-6}} \mathrm{gC} \cdot B \cdot (A \cdot I)_2 \tag{4.15}$$

$$\mathrm{gB} \cdot (A \cdot I)_2 + \mathrm{RNAp} \xleftrightarrow{k_7/k_{-7}} \mathrm{gB} \cdot (A \cdot I)_2 \cdot RNAp \tag{4.16}$$

$$\mathrm{gB} \cdot (A \cdot I)_2 \cdot RNAp \xrightarrow{k_{mB}} \mathrm{gB} \cdot (A \cdot I)_2 + \mathrm{RNAp} + \mathrm{mB} \tag{4.17}$$

$$\mathrm{mB} \xrightarrow{k_{pB}} \mathrm{mB} + \mathrm{B} \tag{4.18}$$

$$\mathrm{mB} \xrightarrow{d_{mB}} \emptyset \tag{4.19}$$

$$\mathrm{B} \xrightarrow{d_B} \emptyset \tag{4.20}$$

$$\mathrm{gC} \cdot (A \cdot I)_2 + \mathrm{B} \xleftrightarrow{k_8/k_{-8}} \mathrm{gC} \cdot B \cdot (A \cdot I)_2 \tag{4.21}$$

$$\mathrm{gC} + \mathrm{B} \xleftrightarrow{k_9/k_{-9}} \mathrm{gC} \cdot B \tag{4.22}$$

$$\mathrm{gC} \cdot (A \cdot I)_2 + \mathrm{RNAp} \xleftrightarrow{k_{10}/k_{-10}} \mathrm{gC} \cdot (A \cdot I)_2 \cdot RNAp \tag{4.23}$$

$$\mathrm{gC} \cdot (A \cdot I)_2 \cdot RNAp \xrightarrow{k_{mC}} \mathrm{gC} \cdot (A \cdot I)_2 + \mathrm{RNAp} + \mathrm{mC} \tag{4.24}$$

$$\mathrm{mC} \xrightarrow{k_{pC}} \mathrm{mC} + \mathrm{C} \tag{4.25}$$

$$\mathrm{mC} \xrightarrow{d_{mC}} \emptyset \tag{4.26}$$

$$\mathrm{C} \xrightarrow{d_C} \emptyset \tag{4.27}$$

Note: the empty set $\emptyset$ means reactant degradation.

Notice that binding between the activator (or repressor) and the gC hybrid promoter is possible, because this can happen even if the repressor B is already bound to the promoter of C and vice-versa (the activator AI_2 bound to the C promoter). For this reaction to take place it requires the previous binding between the protein B and the C promoter.

The previous reactions can be written using ODEs as was shown in section 3.2. The whole model can be written using the law of mass action kinetics:

$$\dot{x_1} = -k_1 x_1 x_2 + k_{-1} x_3 + k_{mA} x_3 \tag{4.28}$$

$$\dot{x_2} = -k_1 x_1 x_2 + k_{-1} x_3 + k_{mA} x_3 - k_7 x_{10} x_2 + k_{-7} x_{15} + k_{mB} x_{15} \tag{4.29}$$

$$\dot{x_3} = k_1 x_1 x_2 - k_{-1} x_3 - k_{mA} x_3 \tag{4.30}$$

$$\dot{x_4} = k_{mA} x_3 - d_{mA} x_4 \tag{4.31}$$

$$\dot{x_5} = k_{pA} x_4 - d_A x_5 - k_2 x_5 x_6 + k_{-2} x_7 \tag{4.32}$$

$$\dot{x_6} = -k_2 x_5 x_6 + k_{-2} x_7 + k_d I_e - k_{-d} x_6 - d_I x_6 \tag{4.33}$$

$$\dot{x_{6e}} = k_{-d} x_6 - k_d x_6 + K_e(t) - d_I I_e \tag{4.34}$$

$$\dot{x_7} = k_2 x_5 x_6 - k_{-2} x_7 - k_3 x_7^2 + 2k_{-3} x_8 - d_{AI} x_7 \tag{4.35}$$

$$\dot{x_8} = k_3 x_7^2 - 2k_{-3} x_8 - k_4 x_8 x_9 + k_{-4} x_{10} - k_5 x_8 x_{11} + k_{-5} x_{12} - k_6 x_8 x_{13} + k_{-6} x_{14} - d_{AI2} x_8 \tag{4.36}$$

$$\dot{x_9} = -k_4 x_9 x_8 + k_{-4} x_{10} \tag{4.37}$$

$$\dot{x_{10}} = k_4 x_9 x_8 - k_{-4} x_{10} - k_7 x_{10} x_2 + k_{-7} x_{15} + k_{mB} x_{15} \tag{4.38}$$

$$\dot{x_{11}} = -k_9 x_{11} x_{17} + k_{-9} x_{13} - k_5 x_{11} x_8 + k_{-5} x_{12} \tag{4.39}$$

$$\dot{x_{12}} = k_5 x_{11} x_8 - k_{-5} x_{12} - k_8 x_{12} x_{17} + k_{-8} x_{14} \tag{4.40}$$

$$\dot{x_{13}} = k_9 x_{11} x_{17} + k_{-9} x_{13} - k_6 x_{13} x_8 + k_{-6} x_{14} \tag{4.41}$$

$$\dot{x_{14}} = k_6 x_{13} x_8 - k_{-6} x_{14} + k_8 x_{12} x_{17} - k_{-8} x_{14} \tag{4.42}$$

$$\dot{x_{15}} = k_7 x_{10} x_2 - k_{-7} x_{15} - k_{mB} x_{15} \tag{4.43}$$

$$\dot{x_{16}} = k_{mB} x_{15} - d_{mB} x_{16} \tag{4.44}$$

$$\dot{x_{17}} = k_{pB} x_{16} - d_B x_{17} - k_9 x_{11} x_{17} + k_9 x_{13} - k_8 x_{12} x_{17} + k_8 x_{14} \tag{4.45}$$

$$\dot{x_{18}} = k_{mC} x_{12} - d_{mC} x_{18} \tag{4.46}$$

$$\dot{x_{19}} = k_{pC} x_{18} - d_C x_{19} \tag{4.47}$$

$$\tag{4.48}$$

where the nomenclature used is shown in Table 4.1.

Table 4.1: List of variables used in the complete model

Variable	Description	Units	Symbol
x_1	DNA promoter gene A	nM	gA
x_2	RNA polymerase	nM	RNAp
x_3	gA·RNAp complex	nM	gA·RNAp
x_4	$mRNA_{gA}$	nM	mA
x_5	A protein	nM	A
x_6	Inducer	nM	I
x_7	A·I monomer	nM	A·I
x_8	$(A{\cdot}I)_2$ dimer	nM	$(A{\cdot}I)_2$
x_9	DNA promoter gene B	nM	gB
x_{10}	$gB(A{\cdot}I)_2$ complex	nM	$gB(A{\cdot}I)_2$
x_{11}	DNA promoter gene C	nM	gC
x_{12}	$gC(A{\cdot}I)_2$ complex	nM	$gC(A{\cdot}I)_2$
x_{13}	gC·B complex	nM	gC·B
x_{14}	$gC{\cdot}B(A{\cdot}I)_2$ complex	nM	$gC{\cdot}B(A{\cdot}I)_2$
x_{15}	$gB(A{\cdot}I)_2$RNAp complex	nM	$gB(A{\cdot}I)_2$RNAp
x_{16}	$mRNA_{gB}$	nM	mB
x_{17}	B protein	nM	B
x_{18}	$mRNA_{gC}$	nM	mC
x_{19}	C protein	nM	C
x_{20}	External I_{ext}	nM	I_e

4.4 ASSUMPTIONS FOR MODEL REDUCTION

If we have a look at the dynamic nature of the interaction among molecules inside cells, it can be taken for certain through experimental evidence, that binding reactions occur very fast in comparison with those corresponding to transcription, translation or even genuine degradation. This feature of *fast binding reactions* can be translated to a mathematical characterization. Indeed, as binding reactions are assumed to be very fast as compared to the other reactions, that can be considered to be at steady state. Thus the respective differential equation of concentration can be equated to zero. Hence, all fast binding reactions that appear in the model whose product is a species resulting from two previous ones will be approximated in this way.

On the other hand, monomer formation (monomerization) is faster that dimerization [6]. Therefore, it will be assumed steady state for the differential equation corresponding to the AI complex formation.

These kind of assumptions are based on perturbation analysis. In essence, *time-scale separation* techniques consider that if some species have much faster dynamics in comparison with the rest, one can apply the *Quasi Steady-State Approximation* (QSSA) to the fast

chemical species, and reduce the number of species involved in the gene synthetic network.

More assumptions can be done through what is known as *system invariants*. In mathematics, an invariant is a property of a class of mathematical objects that remains unchanged when transformations of a certain type are applied to the objects. In the case of reaction networks, it can be observed that some reactions are a linear combination of other ones. Then, the linear combination of the concentrations of the species involved will keep constant in time. These linear combinations can be understood as a kind of quasi-species that keep invariant, i.e. keep constant concentration. These are the so called moieties.

Next we will arrive to an equivalent reduced model of complete one, applying the QSSA and invariant moieties.

First, we consider the bound species formed in fast binding reactions, as shown in table 4.2.

Table 4.2: Bound species considered coming from fast binding reactions

Variable	**Eq. reference**
$gA \cdot RNAp$	4.49
$A \cdot I$	4.50
$gB(A \cdot I)_2$	4.51
$gC(A \cdot I)_2$	4.52
$gC \cdot B$	4.53
$gC \cdot B \cdot (A \cdot I)_2$	4.54
$gB(A \cdot I)_2 \cdot RNAp$	4.55

The corresponding differential equations can be considered to be at quasi-steady state. Therefore, the corresponding species derivatives are set to zero:

$$\dot{x}_3 = k_1 x_1 x_2 - k_{-1} x_3 - k_{mA} x_3 = 0 \tag{4.49}$$

$$\dot{x}_7 = k_2 x_5 x_6 - k_{-2} x_7 - k_3 x_7^2 + 2k_{-3} x_8 - d_{AI} x_7 = 0 \tag{4.50}$$

$$\dot{x}_{10} = k_4 x_9 x_8 - k_{-4} x_{10} - k_7 x_{10} x_2 + k_{-7} x_{15} + k_{mB} x_{15} = 0 \tag{4.51}$$

$$\dot{x}_{12} = k_5 x_{11} x_8 - k_{-5} x_{12} - k_8 x_{12} x_{17} + k_{-8} x_{14} = 0 \tag{4.52}$$

$$\dot{x}_{13} = k_9 x_{11} x_{17} + k_{-9} x_{13} - k_6 x_{13} x_8 + k_{-6} x_{14} = 0 \tag{4.53}$$

$$\dot{x}_{14} = k_6 x_{13} x_8 - k_{-6} x_{14} + k_8 x_{12} x_{17} - k_{-8} x_{14} = 0 \tag{4.54}$$

$$\dot{x}_{15} = k_7 x_{10} x_2 - k_{-7} x_{15} - k_{mB} x_{15} = 0 \tag{4.55}$$

In the context of invariant moieties, we can assume that the species gA, gB, and gC are conserved during the global set of reactions. Recall these species correspond to the genes A, B, and C. The copy number of these genes can be considered constant. If look through the model equations, this hypothesis is confirmed by the presence of the following expressions:

$$\dot{x}_1 + \dot{x}_3 = 0 \tag{4.56}$$
$$\dot{x}_9 + \dot{x}_{10} + \dot{x}_{15} = 0 \tag{4.57}$$
$$\dot{x}_{11} + \dot{x}_{12} + \dot{x}_{13} + \dot{x}_{14} = 0 \tag{4.58}$$

Also RNA polymerase (RNAp) is conserved. RNAp binding to the gene C It is not considered for simplification. The amount of free RNA polymerase can be assumed to be large enough so that variations of bound RNAp can be neglected.

$$\dot{x}_2 + \dot{x}_3 + \dot{x}_{15} = 0 \tag{4.59}$$

Also, from 4.49 and 4.56 we get:

$$\begin{aligned} &\dot{x}_1 = 0 \\ &\int (\dot{x}_1 + \dot{x}_3) = \int 0 \Rightarrow x_1 + x_3 = cst \\ &x_1 = Cg_A \end{aligned}$$

From equation 4.49:

$$x_3 = \frac{k_1}{k_{-1} + k_{mA}} C_{gA} x_2$$

From equations 4.50, 4.55 and 4.59

$$\begin{aligned} &\dot{x_2} = 0 \\ &x_2 = RNApf \end{aligned}$$

where RNApf is the total free RNAp. Recall we assume that the amount of bound RNAp can be neglected. Thus, we can set $RNApf = RNAp$. Thus:

$$\begin{aligned} \dot{x}_4 &= \frac{k_{mA} k_1 RNAp}{k_{-1} + k_{mA}} C_{gA} - d_{mA} x_4 \\ \dot{x}_5 &= k_{pA} x_4 - d_A x_5 - k_2 x_5 x_6 + k_{-2} x_7 \end{aligned}$$

So the first block corresponding to gene A expression is reduced directly to the equations for the mA and A species:

$$\begin{aligned} \dot{x}_4 &= K_{mA} C_{gA} - d_{mA} x_4 \\ \dot{x}_5 &= k_{pA} x_4 - d_A x_5 - k_2 x_5 x_6 + k_{-2} x_7 \end{aligned}$$

with $K_{mA} = \frac{k_{mA} k_1 RNAp}{k_{-1}+k_{mA}}$.

For the dimerization block the analysis of the equations 4.34 and 4.35 gives that if the external supply $K_e(t) = K_e$, that means that at equilibrium

$$I_e = \frac{k_{-d}}{k_d} I_\infty C_{gA} + \frac{K_e}{k_d}$$

From equation 4.50:

$$\dot{x}_7 = k_3 x_7^2 + (d_{AI} - k_{-2}) x_7 - (k_2 x_5 x_6 + 2k_{-3} x_8) = 0$$
$$x_7 = -\frac{d_{AI} + k_{-2}}{2k_3} + \frac{1}{2k_3}\sqrt{(d_{AI} + k_{-2})^2 + 4k_3(k_2 x_5 x_6 + 2k_{-3} x_8)}$$

Notice this root only gives $x_7 \geq 0$

From equations 4.51, 4.55 and 4.57:

$$\dot{x}_9 = -k_4 x_9 x_8 + k_{-4} x_{10} = 0$$

From equation 4.52:

$$-k_5 x_{11} x_8 + k_{-5} x_{12} = -k_8 x_{12} x_{17} + k_{-8} x_{14}$$

From equation 4.54:

$$-k_6 x_{13} x_8 + k_{-6} x_{14} = k_8 x_{12} x_{17} - k_{-8} x_{14}$$

Notice that in equation 4.37 the retroactivity term is the sum of $\dot{x}_9$ (which is zero as deduced above) and the left members of the two equations that have just been written, whose equivalents are the same but with opposite sign, so they get cancelled by each other in $r_8(x)$

$$\begin{aligned}
\dot{x}_8 &= k_3 x_7^2 - 2k_{-3} x_8 \overbrace{\underbrace{-k_4 x_8 x_9 + k_{-4} x_{10}}_{\dot{x}_9 = 0} - k_5 x_8 x_{11} + k_{-5} x_{12} - k_6 x_8 x_{13} + k_{-6} x_{14}}^{r_8(x)} - d_{AI2} x_8 \\
r_8(x) &= 0 \\
\dot{x}_8 &= k_3 x_7^2 - 2k_{-3} x_8 - d_{AI2} x_8
\end{aligned}$$

The procedure for the reduction of the next two blocks (gene B and geneC expression) is as follows. From 4.58, 4.52, 4.53 and 4.54:

$$\dot{x}_{11} = 0$$

From 4.55 and replacing x_2 by RNApf:

$$x_{15} = \frac{k_7 RNAp_f}{k_{-7} + k_{mB}} x_{10}$$

From equations 4.55 and 4.51, $\mathbf{x_{10}}$ results in a $\mathbf{f} = (\mathbf{x_9}, \mathbf{x_8})$, which represents that $gB \cdot (AI)_2$ depends on the binding between gB and the dimer (activator). This can in turn be replaced in equation (from 4.57):

$$x_9 + x_{10} + x_{15} = C_{gB}$$

so, combining the three previous results, it is possible to obtain $\mathbf{x_{15}}$ as a function of $\mathbf{x_8}$:

$$x_{15} = \frac{\frac{k_4}{k_{-4}} k_7 RNAp_f C_{gB} x_8}{(k_{-7} + k_{mB}) + \frac{k_4}{k_{-4}}(k_{-7} + k_{mB}) x_8 + \frac{k_4}{k_{-4}} k_7 RNAp_f x_8}$$

$$= \frac{C_{gB} x_8}{\theta_1' + \theta_2' x_8}$$

with $\theta_1' = \frac{\mathbf{k_{-4}(k_{-7}+k_{mB})}}{\mathbf{k_4 k_7 RNAp_f}}$ and $\theta_2' = \mathbf{1} + \frac{\mathbf{k_4}}{\mathbf{k_{-4}}} \theta_1'$.

Then, from equation 4.45:

$$\dot{x}_{16} = k_{mB} \frac{C_{gB} x_8}{\theta_1' + \theta_2' x_8} - d_{mB} x_{16}$$
$$= K_{mB} \frac{C_{gB} x_8}{\theta_1 + x_8}$$

with $\frac{k_{mB}}{\theta_2'}$ and $\theta_1 = \frac{\theta_1'}{\theta_2'}$.

It turns out that the retroactivity term corresponding to the species $\mathbf{x_{17}}$ is null, as it is shown next. From equations 4.53 and 4.54, the sum of $\mathbf{\dot{x}_{13}}$ and $\mathbf{\dot{x}_{14}}$ gives:

$$k_9 x_{11} x_{17} + k_{-9} x_{13} + k_8 x_{12} x_{17} - k_{-8} x_{14} = 0$$

that is equal to $r_{17}(x)$. Therefore $r_{17}(x) = 0$ and

$$\dot{x}_{17} = k_{pB} x_{16} - d_B x_{17}$$

Finally, using 4.52, 4.53, 4.54 and 4.58, we can obtain a relationship $\mathbf{x_{12}} = \mathbf{f}(\mathbf{x_8}, \mathbf{x_{17}})$, which represents that $gC \cdot (AI)_2$ depends on the effect of the activator (dimer $(AI)_2$) and the effect of the repressor (B). With the aid of the software *Mathematica*, a reduced quasi-empirical expression is obtained:

$$\dot{x}_{18} = k_{mC} C_{gC} \frac{x_8}{\theta_2 + \theta_3 x_8 + \theta_4 x_{17} + \theta_5 x_8 x_{17}} - d_{mB} x_{18}$$
$$\dot{x}_{19} = k_{pC} x_{18} - d_C x_{19}$$

Summarising, after these assumptions and reducing methods, we reach a model with only nine differential equations characterizing the gene network. The species in the reduced model are mA, A, I, $(AI)_2$, mB, B, mC, C, and Ie respectively. The resulting reduced dynamical model is:

$$\dot{x}_4 = K_{mA} C_{gA} - d_{mA} x_4$$
$$\dot{x}_5 = k_{pA} x_4 - d_A x_5 - k_2 x_5 x_6 + k_{-2} x_7$$
$$\dot{x}_6 = -k_2 x_5 x_6 + k_{-2} x_7 + k_d I_e - k_{-d} x_6 - d_I x_6$$
$$\dot{x}_8 = k_3 x_7^2 - 2k_{-3} x_8 - d_{AI2} x_8$$
$$\dot{x}_{16} = K_{mB} C_{gC} \frac{x_8}{\theta_1 + x_8} - d_{mB} x_{16}$$
$$\dot{x}_{17} = k_{pB} x_{16} - d_B x_{17}$$
$$\dot{x}_{18} = K_{mC} C_{gC} \frac{x_8}{\theta_2 + \theta_3 x_8 + \theta_4 x_{17} + \theta_5 x_8 x_{17}} - d_{mC} x_{18}$$
$$\dot{x}_{19} = k_{pC} x_{18} - d_C x_{19}$$
$$\dot{I}_e = Ke - k_d I_e + k_{-d} x_6 - d_{Ie} I_e$$

4.5 REDUCED MODEL

In the Table 4.3, rates and constants, i.e. the set of parameters, involved in the model are listed.

Reordering the equations in previous section, we get a system of nine ordinary differential equations (each one corresponding to the dynamics of one of the species) plus an algebraic equation, and 26 decision variables which are the model parameters for the next optimization step. The resulting model being:

Table 4.3: Rates and constants from the model.

Parameter	Description	Unit
$k_{m_A}, k_{m_B}, k_{m_C}$	gA, gB, gC transcription rate	min^{-1}
$k_{p_A}, k_{p_B}, k_{p_C}$	m_A, m_B, m_C translation rate	min^{-1}
k_d, k_{-d}	I e I_e difussion rate	nM
d_{AI}	(AI) degradation rate	min^{-1}
d_{AI2}	$(AI)_2$ degradation rate	min^{-1}
k_2, k_3	(AI) y $(AI)_2$ association rate	min^{-1}
k_{-2}, k_{-3}	(AI) y $(AI)_2$ dissociation rate	min^{-1}
$C_{g_A}, C_{g_B}, C_{g_C}$	gA, gB, gC copy number	min^{-1}
θ_1	gB promoter constant	min^{-1}
$\theta_2, \theta_3, \theta_4, \theta_5$	gC promoter constants	min^{-1}
$d_{m_A}, d_{m_B}, d_{m_C}$	m_A, m_B, m_C degradation rate	min^{-1}
d_A, d_B, d_C	A, B, C degradation rate	min^{-1}
d_I, d_{Ie}	I, Ie degradation rate	min^{-1}
d_{AI}, d_{AI2}	(AI), $(AI)_2$ degradation rate	min^{-1}

$$\dot{x}_1 = k_{mA} C_{gA} - d_{mA} x_1 \tag{4.60}$$

$$\dot{x}_2 = k_{pA} x_1 - d_A x_2 - k_2 x_2 x_3 + k_{-2} M \tag{4.61}$$

$$\dot{x}_3 = -k_2 x_2 x_3 + k_{-2} M + k_d x_9 - k_{-d} x_3 - d_I x_3 \tag{4.62}$$

$$\dot{x}_4 = k_3 M^2 - 2k_{-3} x_4 - d_{AI2} x_4 \tag{4.63}$$

$$\dot{x}_5 = K_{mB} C_{gC} \frac{x_4}{\theta_1 + x_4} - d_{mB} x_5 \tag{4.64}$$

$$\dot{x}_6 = k_{pB} x_5 - d_B x_6 \tag{4.65}$$

$$\dot{x}_7 = K_{mC} C_{gC} \frac{x_4}{\theta_2 + \theta_3 x_4 + \theta_4 x_6 + \theta_5 x_4 x_6} - d_{mC} x_7 \tag{4.66}$$

$$\dot{x}_8 = k_{pC} x_7 - d_C x_8 \tag{4.67}$$

$$\dot{x}_9 = Ke - k_d x_9 + k_{-d} x_3 - d_{Ie} x_9 \tag{4.68}$$

with $M = -\frac{d_{AI}+k_{-2}}{2k_3} + \frac{1}{2k_3}\sqrt{(d_{AI}+k_{-2})^2 + 4k_3(k_2 x_5 x_6 + 2k_{-3} x_8)}$.

Notice the decrease in the number of variables to consider in this reduced model (listed in Table 4.4, as compared with those that governed the complete model. This means, in turn, a decrease in the computational cost for simulations.

The differential equation for gene B mRNA 4.64 has the form of a Hill function, with x_4 the *dimer* (activation transcription factor). Something similar is seen in the differential equation for gene C mRNA, but in this case we are dealing with an hybrid promoter, so x_6 B protein (repression transcription factor) also appears along with x_4. Doing the limit when $\lim_{x_6 \to 0}$ (no repressor action)

Table 4.4: List of variables used in the reduced model

Variable	Description	Units	Symbol
x_1	$mRNA_{gA}$	nM	mA
x_2	A protein	nM	A
x_3	Inducer	nM	I
M	A·I monomer	nM	A·I
x_4	$(A{\cdot}I)_2$ dimer	nM	$(A{\cdot}I)_2$
x_5	$mRNA_{gB}$	nM	mB
x_6	B protein	nM	B
x_7	$mRNA_{gC}$	nM	mC
x_8	C protein	nM	C
x_9	External I_{ext}	nM	I_e

$$\lim_{x_6 \to 0} \frac{x_4}{\theta_2 + \theta_3 x_4 + \theta_4 x_6 + \theta_5 x_4 x_6} = \frac{x_4}{\theta_2 + \theta_3 x_4}$$

and when $\lim_{x_4 \to 0}$ (no activator action)

$$\lim_{x_4 \to 0} \frac{x_4}{\theta_2 + \theta_3 x_4 + \theta_4 x_6 + \theta_5 x_4 x_6} = 0$$

Chapter 5

Parameters optimization

5.1 COMPUTIONAL METHODS AND TARGETS

The engineering design of the IFF genetic circuit requires a solution that results from the trade-off between different objectives. These are the sensitivity and the precision in system's response, since we are looking for system's adaptation (see Figure 5.1). Both design principles are competing alternatives as results show. This means that they are opposing or mutually exclusive.

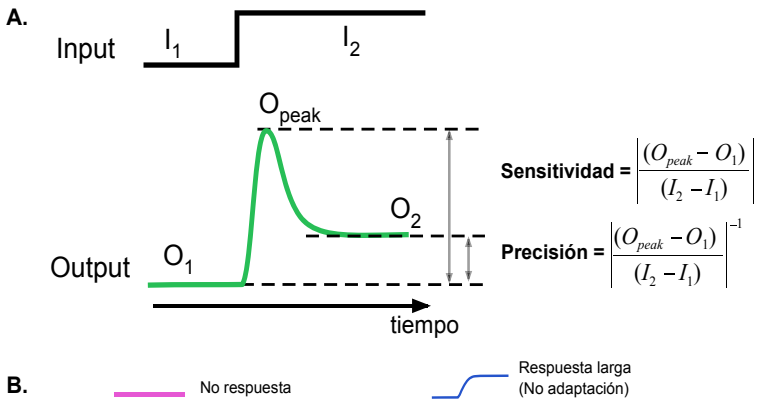

Figure 5.1: A) Adaptation mathematical expression in a genetic circuit B) Other responses which are not adaptive

An usual approach to face a multi-objective optimization problem consists of building a function able to assemble the objectives in a unique index. Normally this is done by using a weighting vector. Nevertheless the solution obtained depends too much on the correct selection of the weighting factors, and it could not reflect with enough clarity the designer preferences in relation with the desired balance of requirements.

Other option is the multi-objective optimization (MOO) [19], which is a natural alternative to face this kind of problems. This is the alternative used in this work. In MOO all

optimization goals are important to the designer, so all of them are optimized simultaneously. Thus, the solution rarely is unique, but a set of solutions called the *Pareto Front* instead (recall section 3.4). In this sense all solutions are optimal and differ from each other in the objectives balance degree. Because of that, this multi-objective problem lead us to a range of solutions that fulfil the needs. Then, the use of appropriate visualization tools will help us to deduce design principles for this genetic circuit.

The algorithm that has been used for this purposed is a multiobjective differential evolutionary algorithm with spherical pruning which has already been used for controller design with several performance objectives and robustness requirements [24].

The selection of a balanced solution under the designer criteria takes place in an *a-posteriori* analysis in the Pareto Front.

In this work the tool used to visualize the resulting front is the Level Diagram [I] (LD) [5, 23].

5.1.1 Problem approach

The MOO problem we face consists of looking for the set of values for the 26 decision variables θ –which are the model parameters in equations 4.60 to 4.68– that minimize the following objectives

$$\min_{\theta \in \Re^{26}} \mathbf{J}(\theta) \quad = \quad [J_1(\theta), J_2(\theta), J_3(\theta)] \in \Re^3$$

with:

$$J_1(\theta) \quad = \quad \frac{2(x_9(t_f) - x_9(t_0))}{\int_{t_0}^{t_f} |\frac{dx_8}{dt}| dt} \tag{5.1}$$

$$J_2(\theta) \quad = \quad \frac{x_8(t_f) - x_8(t_0)}{x_9(t_f) - x_9(t_0)} \tag{5.2}$$

$$J_3(\theta) \quad = \quad \frac{1}{\int_{t_0}^{t_f} |\frac{dx_6}{dt}| dt} \tag{5.3}$$

The index J_1 represents the inverse of absolute total variation of the concentration of the protein C (x_8) normalized with respect to changes in the external inducer x_9 . In other words, the target is to minimize the inverse of the sensitivity, i.e. to maximize the pick height in the signal response. The reason for including the number 2 in this expression is because the integer $\int_{t_0}^{t_f} |\frac{dx_8}{dt}| dt$ takes into account the differential changes while the function goes up and then goes down, and it 'sums' these changes so that if the function is supposed to return to the previous level ,the sum is the double value of just the response part until it reaches the maximum level.

[I] Tool available at `http://www.mathworks.com/matlabcentral/fileexchange/24042`

The index J_2 represents the total variation of the protein C (x_8) normalized with respect to changes in x_9. This corresponds to the precision of the circuit.

Finally, the index J_3 represents the inverse of absolute total variation of the concentration of the protein B (x_6) normalized with respect to changes in the external x_9.

In this work, attention is paid mainly to the first two indices, corresponding to sensitivity and precision of the circuit. The minimization of these two objectives by means of the optimization tool will lead us to determine in this dynamic system the Pareto Set $\theta_{\mathbf{P}}$, and its projection in the objectives space as the Pareto Front $\mathbf{J_P}$.

The MOO problem tries to approximate the best parameters $\theta^*_{\mathbf{P}}$ in a Pareto-optimal box $\theta_{\mathbf{P}}$ that give the best Pareto-front approximation $\mathbf{J}^*_{\mathbf{P}}$. Such search could be done through a random Monte-Carlo sampling in the decision variables space θ –the set of parameters determining our biological model–, and then filter the solutions in order to obtain the $\theta^*_{\mathbf{P}}$ that define the front $\mathbf{J}^*_{\mathbf{P}}$. For problems with few parameters this can result in a good option, but for problems with a large number of parameters as in this project case, with $(mxn) = (26x3)$, it is more efficient to use a good MOO algorithm to approximate this solution. Even so, an additional analysis with Monte-Carlo sampling, forcing parameters to move out of range, will be done.

The algorithm used for this case is $sp-MODE$[II][24], which is a version of the multi-objective differential evolutionary algorithm (MOEA) with spherical pruning described in [24].It is a MOEA that served us for the mentioned purpose. Basic features of the algorithm are:

- Improving Convergence by using an external file to store solutions and include them in the evolutionary process.

- Improving Spreading by using the spherical pruning mechanism.

- Improving Pertinency of solutions by a basic bound mechanism in the objective space as described in [8].

As for the visualization tool, the level diagram (LD) is based on the classification of calculated optimal parameters θ^*_P making each objective $J_q(\theta)$ to be normalized with respect to its minimum and maximum value. For each normalized vector $\hat{\mathbf{J}}(\theta)$, the norm-p is applied as

$$\|\hat{\mathbf{J}}(\theta)\|_p := \left(\sum_{q=1}^{m} \|\hat{J}(\theta)_q\|^p \right)^{1/p}$$

[II]Tool available in `http://www.mathworks.com/matlabcentral/fileexchange/39215`

so as to evaluate the distance to the ideal solution $\mathbf{J}^{ideal} = \mathbf{J}^{min}$.

A LD is an alternative for the m-dimensional set and front visualization and analysis. This visualization and analysis is not a trivial task when the number of objectives is larger than 3 and/or the number of decision variables in Pareto package is large like in this case.

In the LD (see Figure 5.2) a graph for each objective $q \in [1,\ldots,m]$ is displayed, and another one for each decision variable $l \in [1,\ldots,n]$ (see Figure 5.3) where the Y-axis is $\|\hat{\mathbf{J}}(\theta)\|_p$ and the X-axis corresponds to the objective value or decision variable depending on the case. So, a given solution will have the same value -y in all graphs. Intentionally, the LD has been modified so that points of both graphs use a code ranging from blue, that represents low values of $J_1(\theta)$, to dark red symbolizing high values of $J_1(\theta)$. This colors correspondence will help to evaluate general tendencies along the Pareto front and compare solutions according to the selected norm.

Additionally and with the purpose of facilitate the analysis, also the dynamic response of species from the model have been simulated along with the *transcription/degradation mRNA* and *translation/degradation protein* proportions using the same color code.

5.2 COMPUTATIONAL IMPLEMENTATION

One can think that a high production of the protein B, which acts as repressor for the C promoter, would make the C response return rapidly to its pre-stimulated value with even less error. Following this intuition, it was tried to optimize the three objectives mentioned in the previous section:

1. J_1 , the inverse of **absolute total variation** of C protein concentration normalized with respect to changes in external inducer.
2. J_2, the **total variation** of C protein normalized with respect to changes in external inducer, that means the precision.
3. J_3 , the inverse of **absolute total variation** of B protein concentration normalized with respect to changes in external inducer.

In other words, it was tried to optimize the index corresponding to the pick height and the steady state error of the C protein, and the pick height of the B protein respectively.

The inclusion of J_3 as an objective of our MOO shouldn't be taken it with special rigour, but as a soft constraint for our circuit design. Indeed, first simulations using just J_1 and J_2 as optimization indices gave some poor values for the production of protein B. Thus, it was decided to include this production as a soft objective for our problem.

The index J_3 was optimized in both for the case of maximization, and the case of minimization. That is, first maximizing the pick height in the production of B, and in a second approach by minimizing it. The reason for the second approach was the importance of knowing how much could we reduce the production of B. This is important, for instance, when attending to a reduction of internal cell resources.

Results showed that either maximizing B production or minimizing B production, the optimizer was able to find ranges for the problem parameters that resulted in the circuit output to perform adaptation. And even more, depending on the approach, slightly different designing rules could be inferred.

In the model, transcription rates and gene copy number multiply each other $(Km_i \cdot Cg_i)$ in the dynamic equations of the mRNA of each node $(i = 1,2,3)$. So they were eventually set together as a decision variable for the optimizer. When considered as separated variables it was observed that the optimizer tended to fix one of them and play with the variability of the other one.

The searching ranges for the optimizer were set widening the typically range with biological sense. So the analysis is slightly expanded out of the *a priori* possible values that we can find in nature.

Other relevant issue is that related to the limit from which it is considered that a circuit performs adaptation. That means that a pertinency on the objectives must be applied. The limits established in this work were as seen in table 5.1.

Table 5.1: Objectives pertinence

Objective	**Description**	**Expression**	**Pertinency range**	
			Min	Max
J_1	C protein sensitivity	$\frac{2(x_9(t_f)-x_9(t_0))}{\int_{t_0}^{t_f} \lvert \frac{dx_8}{dt} \rvert dt}$	$1e^{-3}$	100
J_2	C protein precision	$\frac{x_8(t_f)-x_8(t_0)}{x_9(t_f)-x_9(t_0)}$	$1e^{-4}$	0.5
J_3	B protein sensitivity	$\frac{1}{\int_{t_0}^{t_f} \lvert \frac{dx_6}{dt} \rvert dt}$	$2e^{-4}$	0.1

Note: the pertinency range for J_1 and J_3 is translated to [7500 0.75] and [5000 10] respectively in terms of pick height in nanomols (nM).

CODE.

From an existent previous *Matlab* code provided by some members of the research group *GCSC* of the *Instituto Universitario ai2*, who have already work in the deterministic and stochastic modelling of a synthetic genetic circuit [6], some modifications were done to apply the developed code to the reduced model resulting in this work.

A short description of the main functions integrating this code and justification of the value sets is given below. It has been divided in two groups: files more related to the model computational characterization, and files used by the optimizer, which link to the first set.

Model code

- ***model_3genes.m*** is essentially the **model system code**. It gives a vector with the 9 variables with which the ode algorithm works.

- ***principal_func.m*** gives the **three objectives values vector**, each time that a dynamic response for the whole model is obtained.

  The 9 variables are initialized, and the 26 parameters are not because the optimizer will work with a given range in its code.

  The *ode23s* algorithm gives the variables values Y for each t, using *model_3genes.m* . This *ode* algorithm was selected because our system model is what it is known as *stiff*, in terms of the numerical solution of ordinary differential equations, i.e. it has both slow and fast dynamics. An ordinary differential equation problem is stiff if the solution being sought is varying slowly, but there are nearby solutions that vary rapidly, so the numerical method must take small steps to obtain satisfactory results.

  GraphicsVarias.m is used once the ode algorithm has calculated the solutions of the model (the value of each species at every time instant).

  For computational simulation, it is necessary to fix a time (in seconds) different from t_0 when the step signal $Ke(t)$ (external supply) appears, to let the system stabilize itself before. The value

  $$time\_step = 300$$

  has been used. On the other hand, the amplitude (in nM) of this step signal has been fixed to

  $$amp\_step = 10$$

  With respect to the simulation parameters, the simulation sampling time (Ts) was fixed to $1e^{-3}$ minutes, and a total simulation time $Tsim = 600$ minutes was used.

- ***GraphicsVarias.m*** gives the **dynamic response of each variable plot** (9 subplots). Each set of the parameters (the 26 decision variables in our MOO), gives a dynamic

response for each variables or species. This means that this Matlab function will give the whole model dynamic response for the Pareto set provided by the optimizer.

- ***modelo3genes_poblacion.m*** is the head function that accumulates in a matrix the objective values vector given by *principal_func.m*.

MOO code

First, highlight that we can make use of the file ***Tutorial.m*** as help. In this tutorial, basic problems are solved using the spMODE algorithm, which is a version of the multi-objective differential evolution algorithm with spherical pruning described in [24].

The MOO code implements a version of the multi-objective differential evolution algorithm with spherical pruning.

The first step is to run the *spMODEparam* file to build the variable 'spMODEDat' with the variables required for the optimization. Here the number of objectives are defined, also the number of decision variables and the *'Cost Function'*, which brings the objectives matrix after previous *ode* simulations (by means of interlinked functions mentioned above, constituting in essence the problem 'nucleus' or characterization). The field of search, and bounds to improve pertinency of solutions in the objective space so as to cut solutions with no interest to the DM, are defined here too. Also other aspects, such as maximum Pareto optimal solutions required and a bound on the number of function evaluations.

Once the Pareto set and the Pareto front are found by the optimizer, results can be plot with optional features through the *Leveltool*. This tool provides the LD visualization for the MCDM (see section 3.4.1).

- ***spMODEparam.m*** generates the required parameters to run the spMODE optimization algorithm.

  In this file the variables regarding the multi-objective problem are defined. The values of interest for our problem are:

  1. Number of objectives.
     spMODEDat.NOBJ = 3
  2. Number of decision variables.
     spMODEDat.NVAR = 26
  3. Cost Function.
     spMODEDat.mop = str2func('CostFunction')
  4. Problem Instance.
     spMODEDat.CostProblem = 'modelo3genes'

5. Maximum and minimum values for the parameters or decision variables are fixed in order to give a range to the optimizer to search the optimal solutions, (spMODEDat.FieldD , see Table 5.2). k_d and d_{I_e} were fixed to avoid the optimizer to modify the model input I_e, as we want an step input determined by $K_e(t)$. If k_d and d_{I_e} are not fixed the optimizer gives some I_e response with slow dynamics. The table 5.2 gives the search ranges chosen for all the decision variables in the problem.

Table 5.2: Optimizer searching range

Decision variable	Range	Biological values	Reference
$K_{m_A} \cdot C_{g_A}$	[1 200]	-	
$K_{m_B} \cdot C_{g_B}$	[1 200]	-	
$K_{m_C} \cdot C_{g_C}$	[1200]	-	
k_d	0.06 *fixed*	[0.01 0.1] $\sim$ 0.06 min^{-1}	[6]
d_A	[0.01 0.1]	0.035 min^{-1}	[21]
d_B	[0.01 0.1]	0.035 min^{-1}	[21]
d_C	[0.01 0.1]	0.035 min^{-1}	[21]
θ_1	[200 600]	-	
θ_2	[0.01 0.2]	-	
θ_3	[0.0001 1]	-	
θ_4	[0.0005 10]	-	
θ_5	[0.1 10]	-	
k_{p_A}	[1 100]	50 $prot \cdot (mRNA \cdot min)^{-1}$	[21]
k_{p_B}	[1 100]	50 $prot \cdot (mRNA \cdot min)^{-1}$	[21]
k_{p_C}	[1 100]	50 $prot \cdot (mRNA \cdot min)^{-1}$	[21]
k_2	[1 20]	0.01 $(nM \cdot min)^{-1}$	[6]
k_3	[0.1 5]	0.05 $(nM \cdot min)^{-1}$	[6]
k_{d_2}	[100 250]	100 nM	[6]
k_{d_3}	[1 30]	20 nM	[6]
d_{m_A}	[0.01 0.5]	0.3624 min^{-1}	[6]
d_{m_B}	[0.01 0.5]	0.3624 min^{-1}	[6]
d_{m_C}	[0.01 0.5]	0.3624 min^{-1}	[6]
d_I	[0.001 0.5]	0.0164 min^{-1}	[6]
d_{I_e}	0.0164 *fixed*	0.0174 min^{-1}	[6]
d_{AI}	[0.01 0.5]	0.0174 min^{-1}	[6]
d_{AI2}	[0.01 0.5]	0.0174 min^{-1}	[6]

6. Bounds on objectives.
   spMODEDat.Pertinency=[1E-3 100; 1E-4 0.5; 2E-4 0.1] A row for each objective, minimum and maximum values desired.

- ***CostFunction.m*** calls the cost function of your own multi-objective problem, in this case *modelo3genes_poblacion.m*, and include a mechanism to improve basic pertinency (Objective space bounded).

5.3 SIMULATIONS AND RESULTS

As three different approaches were taken into account, results will be shown and commented separately. Then in section 5.5, design principles will be derived from the conclusions drawn from the analysis of the results obtained from these approaches: MOO maximizing the B protein production, MOO minimizing the B protein production, and Monte-Carlo Sampling additional analysis.

It is also worth mentioning again the two criteria that will be addressed: 'High sensitivity' and 'High precision'. Since we are dealing with two conflicting objectives, this differentiation has to be taken into account for the MCDM stage.

5.3.1 Multi-objetive Optimization Maximazing B protein production

Recalling previous sections, in the LD a graph for each objective is displayed (see Figure 5.2), and another one for each decision variable (see Figure 5.3), where the Y-axis is $\|\hat{\mathbf{J}}(\theta)\|_p$ and X-axis correspond to the objective value or decision variable depending on the case. So, a given solution will have the same value -*y* in all graphs. Intentionally, the LD has been modified so that points of both graphs use the same color code, ranging from blue that represents low values of $J_1(\theta)$, to dark red symbolizing high values of $J_1(\theta)$.

The dynamic response of the system can be seen in Figure 5.4, where solutions keep the same colour convention as those from the other graphs (figures 5.2, and 5.3).

Additionally, and with the purpose of facilitating the analysis, also the dynamic responses of the model species have been simulated along with the *transcription/degradation mRNA* and *translation/degradation protein* proportions using the same color code (Figures 5.5 and 5.6 respectively). These expressions emerge from the study of the system when it is at the equilibrium.

For instance, the extreme point X (see Figure 5.2, the turquoise blue point with the lower norm), blue coloured because of its low $J_1(X) = 0.025$ value, has a norm $\|\hat{\mathbf{J}}(\mathbf{X})\| = 1$. The same point X is represented both in the others objective graphs, in the decision variable graph

5.3, and in the time evolution system representation 5.4, with the same color and the same norm. Such relationship helps to evaluate and identify general tendencies within the Pareto set and the Pareto front, letting to infer design rules for the genetic circuit.

Notice that in this MOO, all desired objectives had to be minimized. For this reason the J_1 index represents the inverse of the pick height. Thus, the smaller the value of J_1 , the bigger the sensitivity (pick height).

Sensitivity and precision features being conflicting objectives is clearly appreciated in the graph 5.2. The higher the precision (red points in the second graph), the worse the sensitivity (red points in the first graph).

In Figure 5.4 the adaptation behaviour in the protein C is clearly demonstrated for all points in the Pareto front.

In table 5.3, we can see the different tendencies that the model parameters follow to optimize the problem (according to Figure 5.3). In some of them it can be appreciated even the tendency to certain values according to the criterion of 'High sensitivity' or the criterion of 'High precision', that is, blue and red points appear gathering separately in different values or ranges.

Figure 5.2: Pareto Front representation for the three objectives using the Level Diagrams provided by the MOO procedure. Solutions with the same level in LD have a direct relationship in each graph. The graph in this figure corresponds to the approach looking for B protein production maximization. Sensitivity and precision features being conflicting objectives is clearly appreciated in this graph: the higher the precision (red points in the second graph), the worse the sensitivity (red points in the first graph).

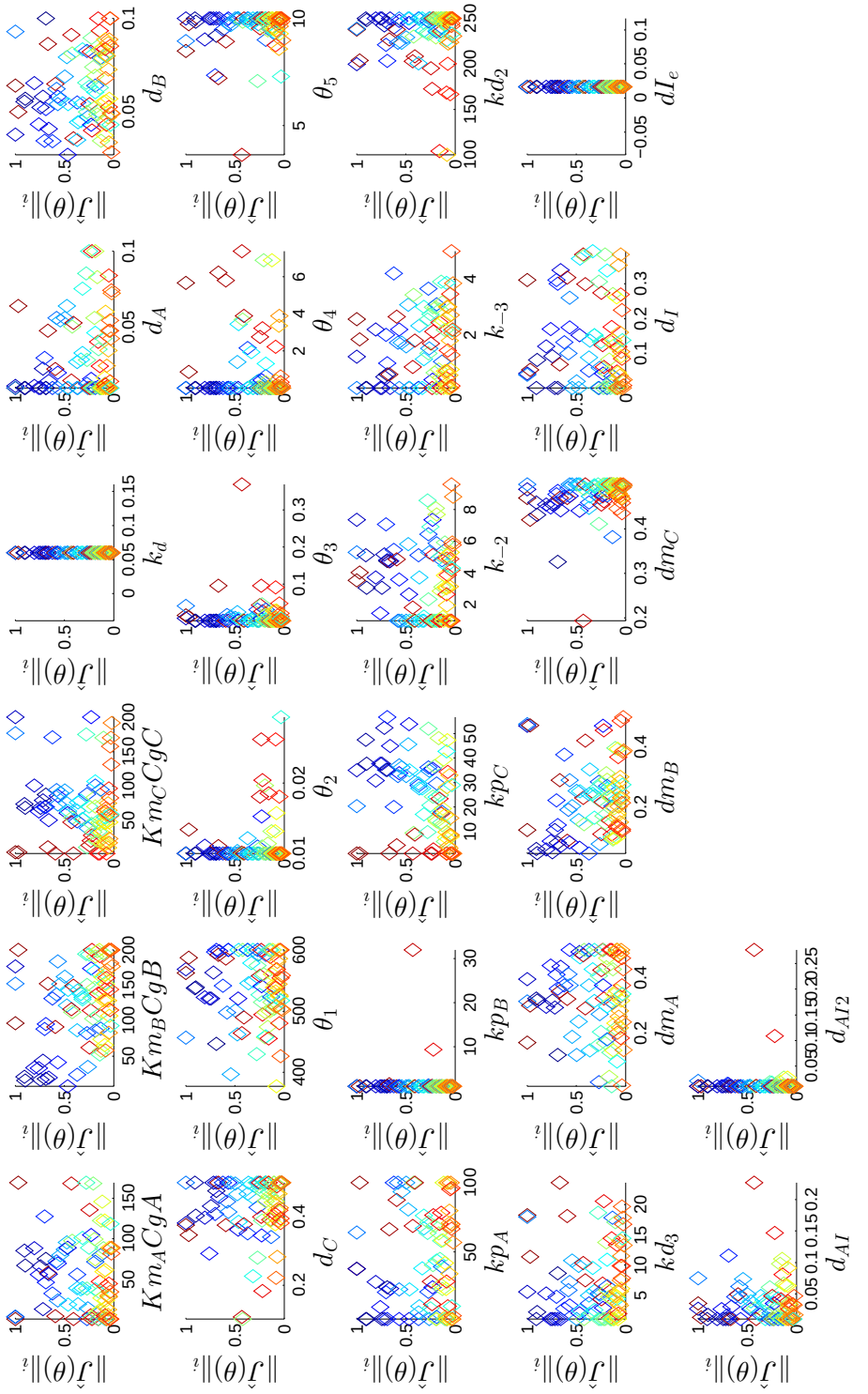

Figure 5.3: Pareto Set representation for the three objectives using the Level Diagrams provided by the MOO procedure. Solutions with the same level in LD have a direct relationship in each graph. The graph in this figure corresponds to the approach looking for B protein production maximization. The graphs show the values of the parameters corresponding to different regions of the Pareto Front.

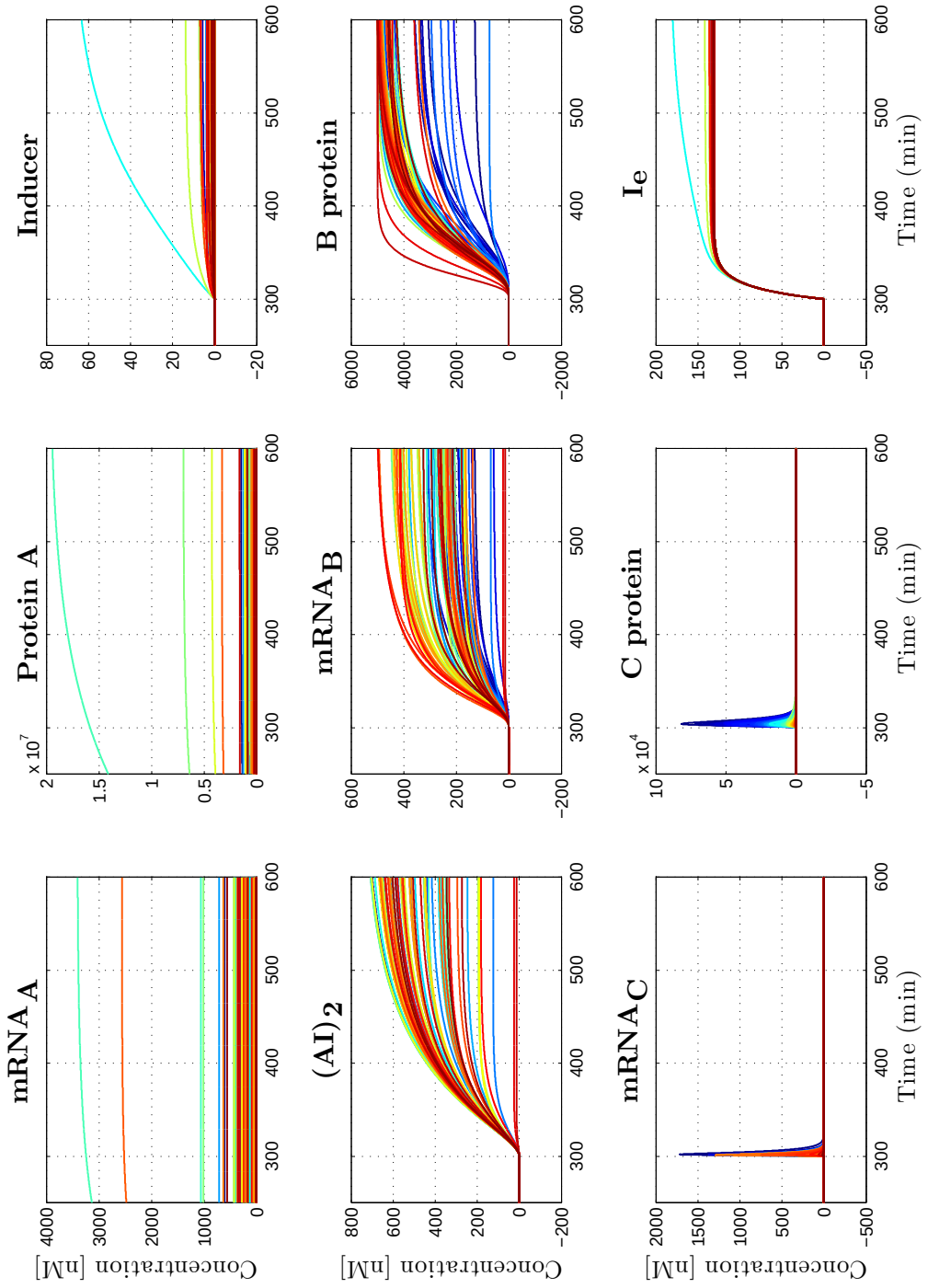

Figure 5.4: Time evolution of the 9 biochemical species of the model. The adaption behaviour in the C protein is clearly seen. The graph in this figure corresponds to the approach looking for B protein production maximization.

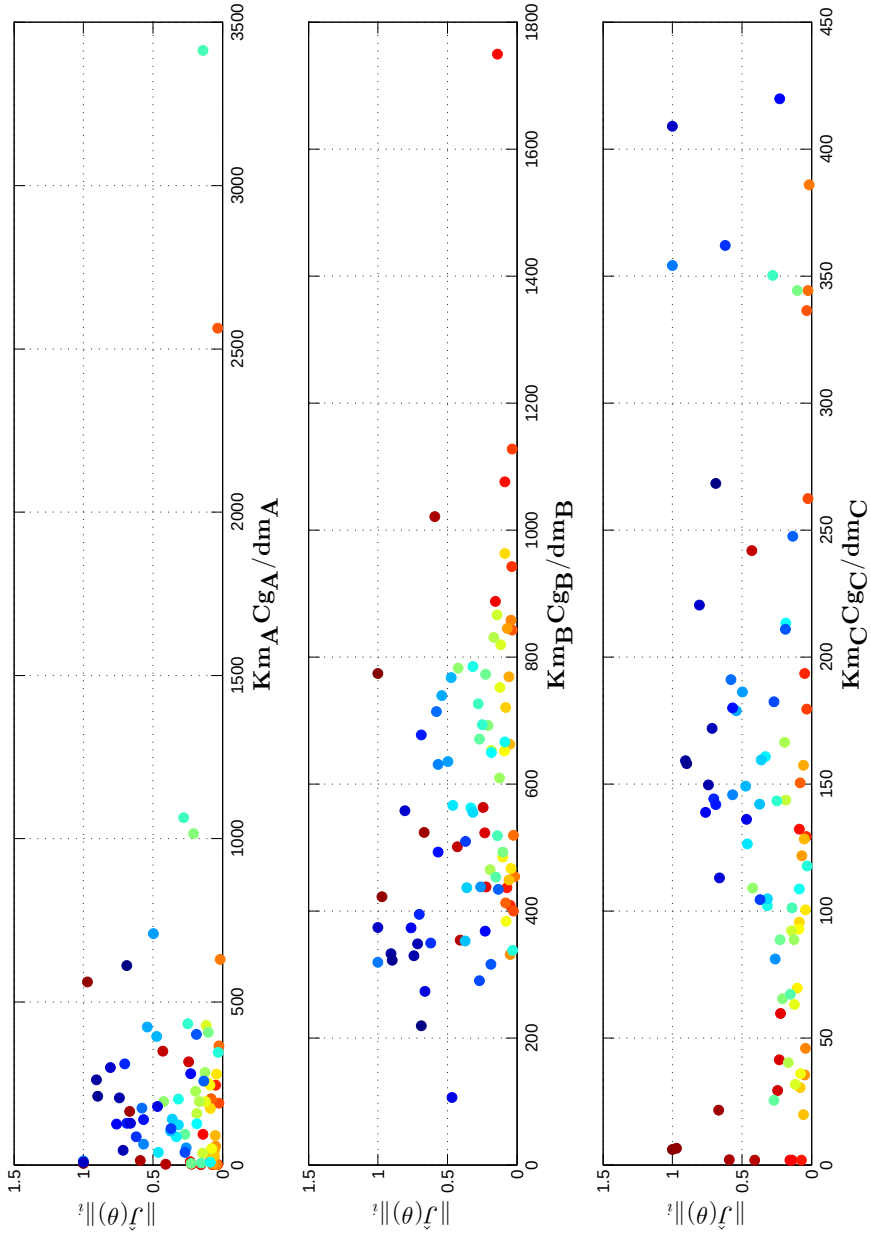

Figure 5.5: Transcription/degradation protein proportions. A) K_{m_A}/d_mA doesn't seem to affect directly the protein C sensitivity or precision (gene A is constitutive) B) K_{m_B}/d_mB with a little wider range of variation than K_{m_A}/d_mA and K_{m_C}/d_mC, doesn't seem to affect directly the C sensitivity or precision C) For a high sensitivity high values of K_{m_C}/d_mC are required, whereas if the design target looks for a high precision, low values of K_{m_C}/d_mC are preferred.

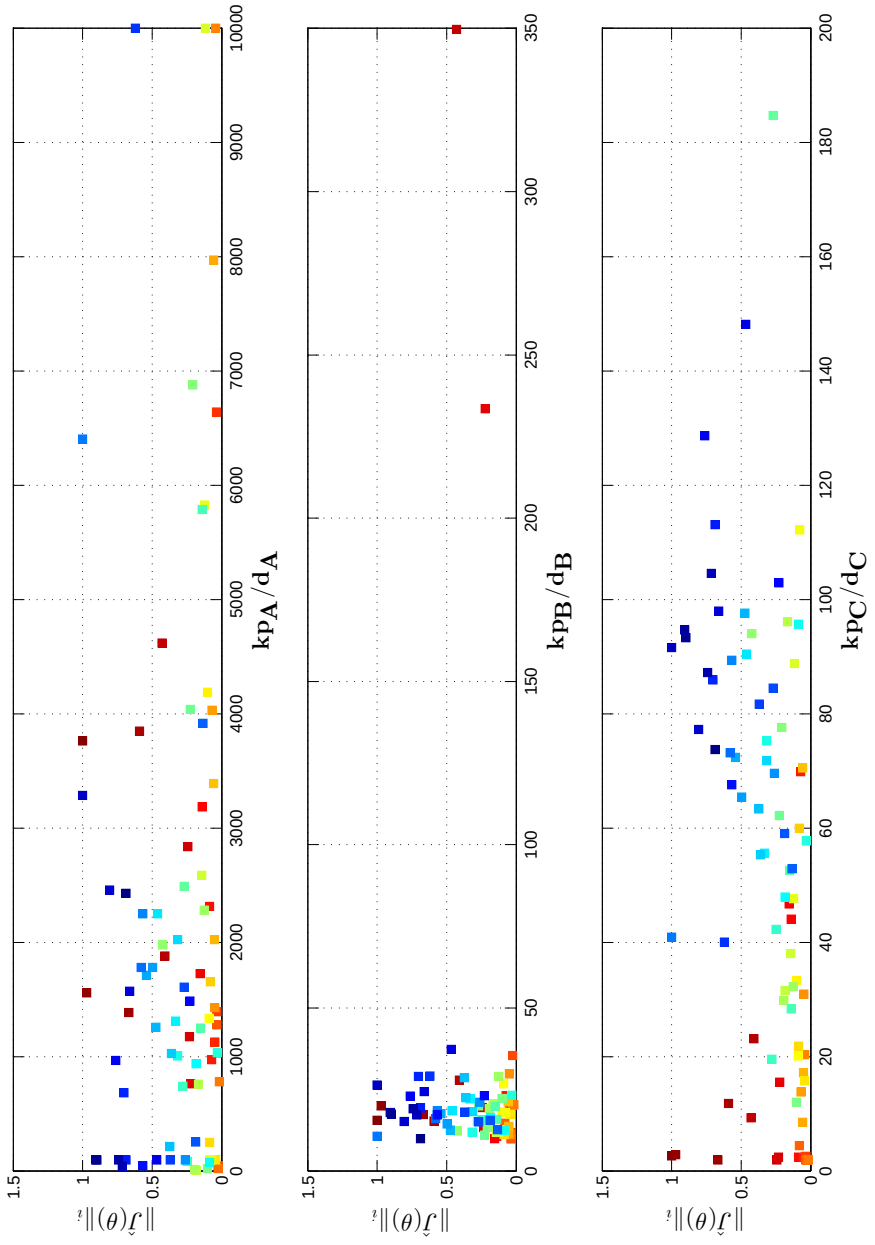

Figure 5.6: Translation/degradation protein proportions. A) k_{pA}/d_A rate takes significantly higher values than the others, but it doesn't affect directly in C sensitivity or precision. B) The quotient between B protein expression and its degradation hardly varies with respect to the sensitivity and precision presented circuit, while it keeps in the shown range. C) For a high sensitivity it is required high values of k_{pc}/d_C too, whereas if the design target looks for a high precision, low values of k_{pc}/d_C are preferred

Certain parameters show wide variability inside the sampling range. Nevertheless, others show clear tendencies to attain high or low values. For instance, the degradation rates of the monomer and the dimer tend to low values (since less degradation of this substance means more amount of it, and it is necessary for the activation and production of the C protein). Something similar happens with the degradation rate of the A protein; it tends to take low values. This makes sense since a high amount of the A protein affects the production of B. As a component of the activator, a high amount of A will produce a high amount of B, and in this case it is expected to maximize B.

Other parameter that presents a very clear tendency is the degradation rate of the C protein. With high values of this parameter, the concentration of the C protein is expected to return faster to its original level. So it is a key parameter to correctly achieve adaptation.

The parameters θ in the hybrid promoter also are forced by the optimizer to take certain values for the system to attain the adaptive behaviour. These parameters affect as coefficients of the x_4 and x_6 terms (mRNA of B and C) increasing or decreasing its importance in the equation corresponding to the hybrid promoter.

With respect to the transcription and translation ratios, which in essence constitute the process gains, it is interesting to highlight that just for the C protein transcription and translation it is possible to see a clear differentiation in the values depending on the selected criterion. These ratios $\frac{K_{mC} \cdot C_{gC}}{dmC}$ and $\frac{K_{pC} \cdot C_{gC}}{dC}$ seem to work as *tuning knobs* for the level of fulfilment of the objectives. The 'High sensitivity' criterion is seen to be less restrictive when selecting a value, as the range of blue points is wider than that one for the red ones. The $\frac{K_{pB}}{dB}$ ratio resulted to be very low in comparison with the rest.

5.3.2 Multi-objetive Optimization Minimazing the production of B protein

Similarly to the previous section, for this approach the results for each objective are displayed in Figure 5.7. The resulting decision variables (parameters) are shown in Figure 5.8, and the system dynamic response in Figure 5.9, and in Figure 5.10 the transcription and translation ratios.

The fact that sensitivity and precision features are conflicting objectives is appreciated in the figure 5.7. The higher the precision (red points in the second graph), the worse the sensitivity (red points in the first graph).

In Figure 5.9 the adaptive behaviour of the C protein is clearly demonstrated for all the values of the parameters within the Pareto set.

In the table 5.4, the different tendencies that model parameters follow to optimize the problem can be observed (related to Figure 5.8). In some cases it can be clearly appreciated

the tendency to certain values according to the criterion of ‘High sensitivity’ or the criterion of ‘High precision’ being chosen. Indeed, blue and red points appear gathering separately in different values or ranges.

As it was the case for the first approach, certain parameters show wide variability inside the simulation sampling range. Nevertheless, others show clear tendencies to get high or low values. For this approach (minimization of B as third objective) the value of the A protein degradation rate tends to be higher than when the B production was being maximized. That makes sense, since in this approach we deal with the opposite goal (to minimize B). The monomer and dimer degradation rates tend to low values as in the case before, and the C protein degradation rate to high values as was expected intuitively.

With respect to the transcription and translation ratios, which in essence constitute the process gains, also the significant *tuning knobs* seem to be the ones corresponding to the protein C.

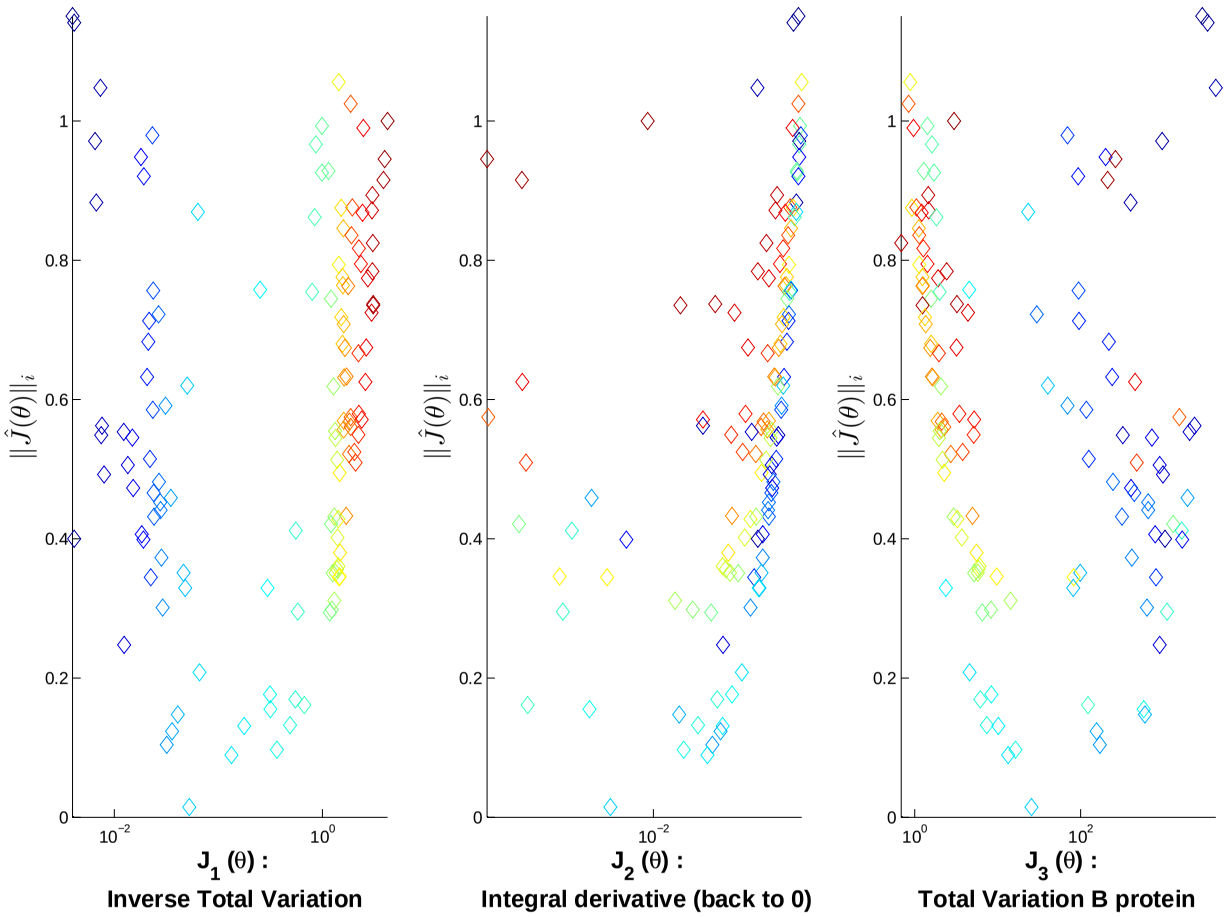

Figure 5.7: Pareto Front representation for the three objectives using the Level Diagrams provided by the MOO procedure. Solutions with the same level in LD are related in each graph. These results correspond to the minimization of the protein B. Like in the approach to maximize the B protein production, the fact that sensitivity and precision features are conflicting objectives is appreciated in this graph: the higher the precision (red points in the second graph), the worse the sensitivity (red points in the first graph).

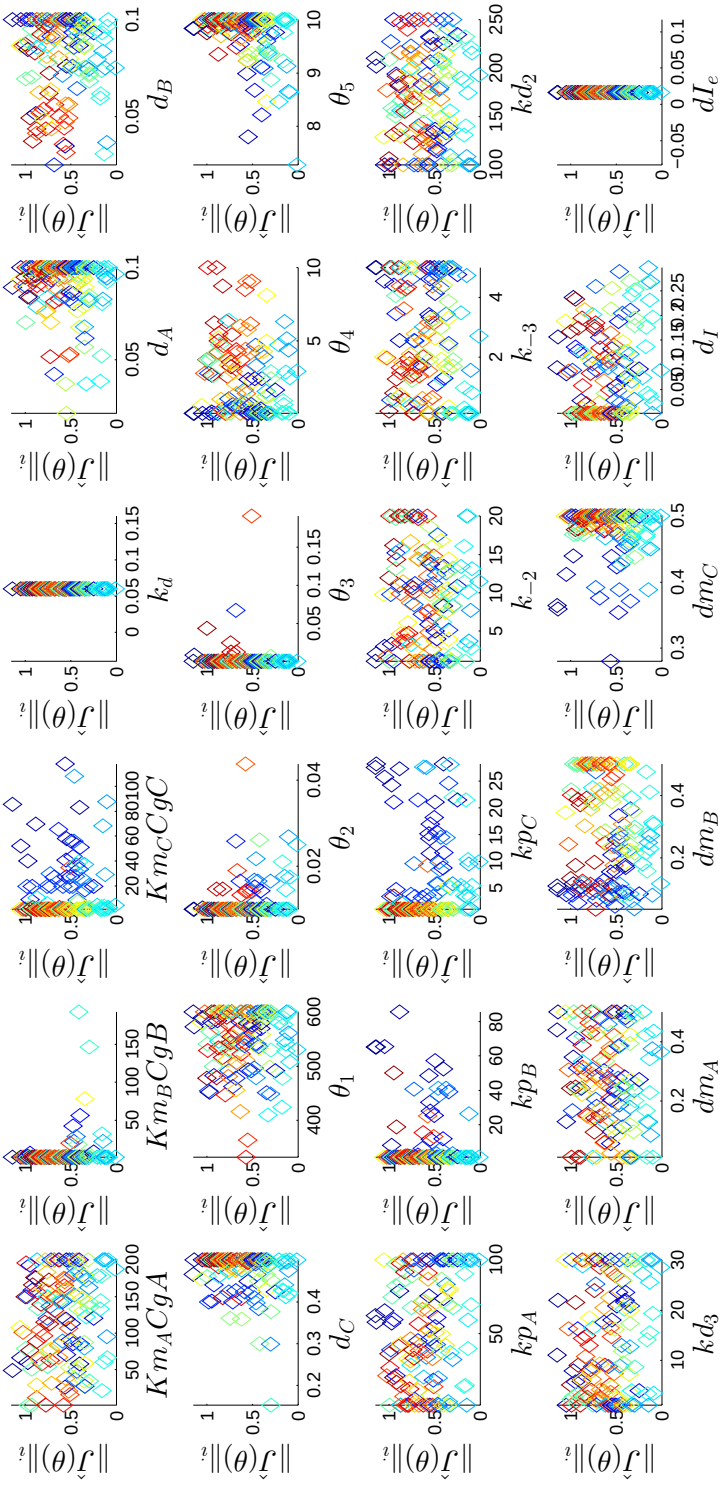

Figure 5.8: Pareto Set representation for the three objectives using the Level Diagrams. Solutions with the same level in LD are related in each graph. These results correspond to the minimization of the protein B.

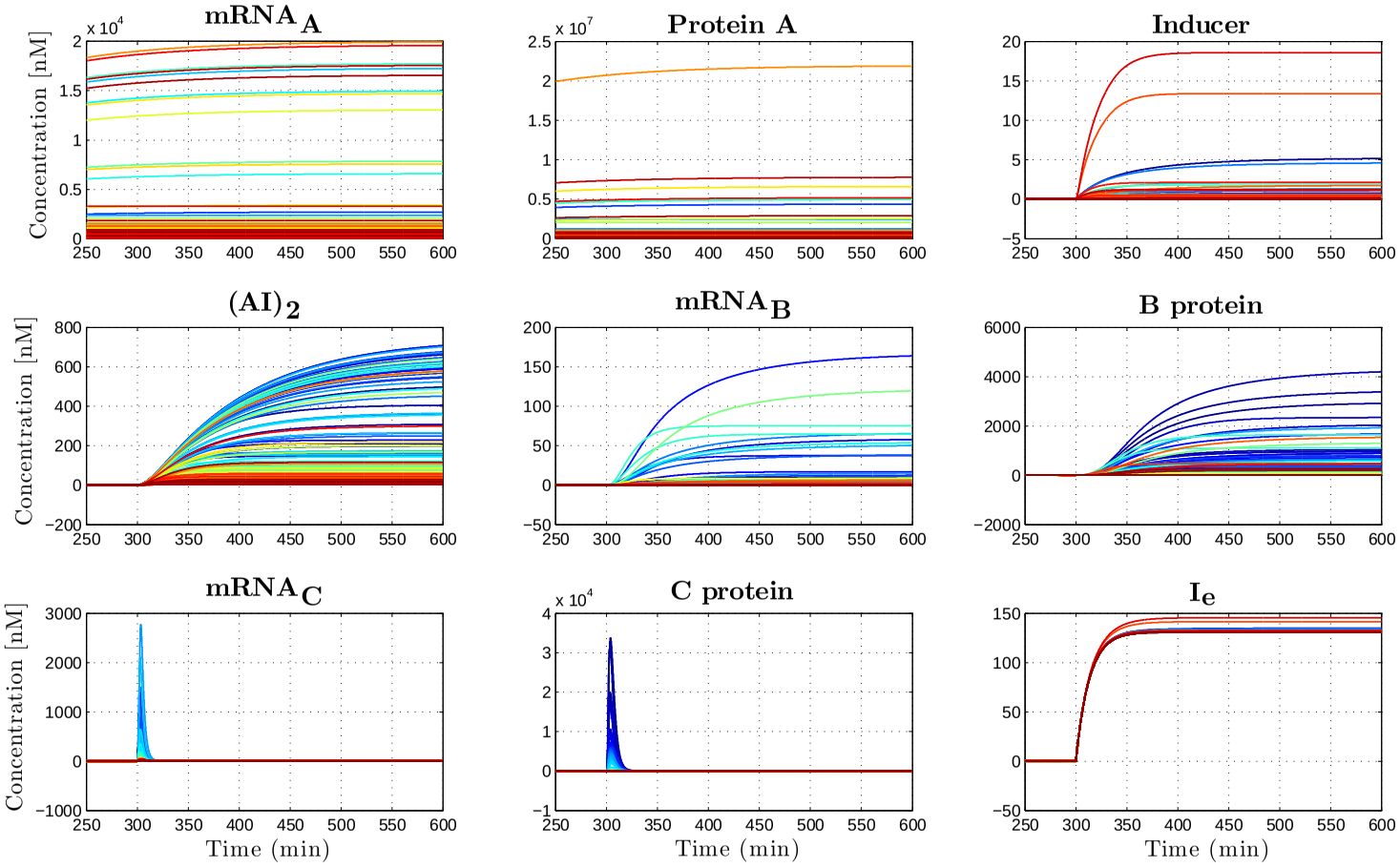

Figure 5.9: Time evolution of the 9 biochemical species of the model. The adaption behaviour in the protein C is clearly demonstrated for all parameters values in the Pareto set. These results correspond to the minimization of the protein B.

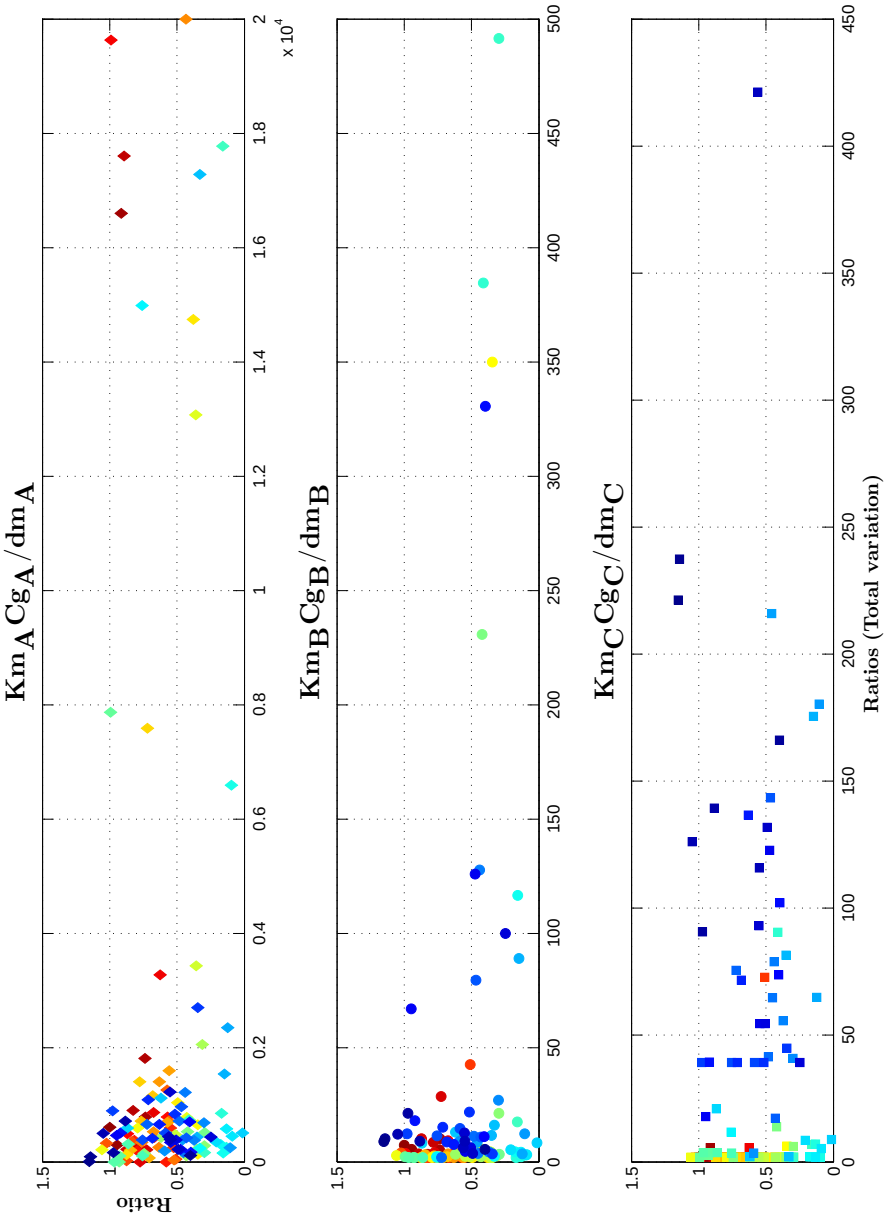

Figure 5.10: Transcription/degradation protein proportions. These results correspond to the minimization of the protein B. A) K_{mA}/d_{mA} doesn't seem to affect directly the C sensitivity or precision (gene A constitutive). B) K_{mB}/d_{mB} take low values with respect to K_{mA}/d_{mA} and doesn't seem to affect directly the C sensitivity or precision. C) For a high sensitivity high values of K_{mC}/d_{mC} are required, whereas if the design target looks for a high precision, low values of K_{mC}/d_{mC} are preferred.

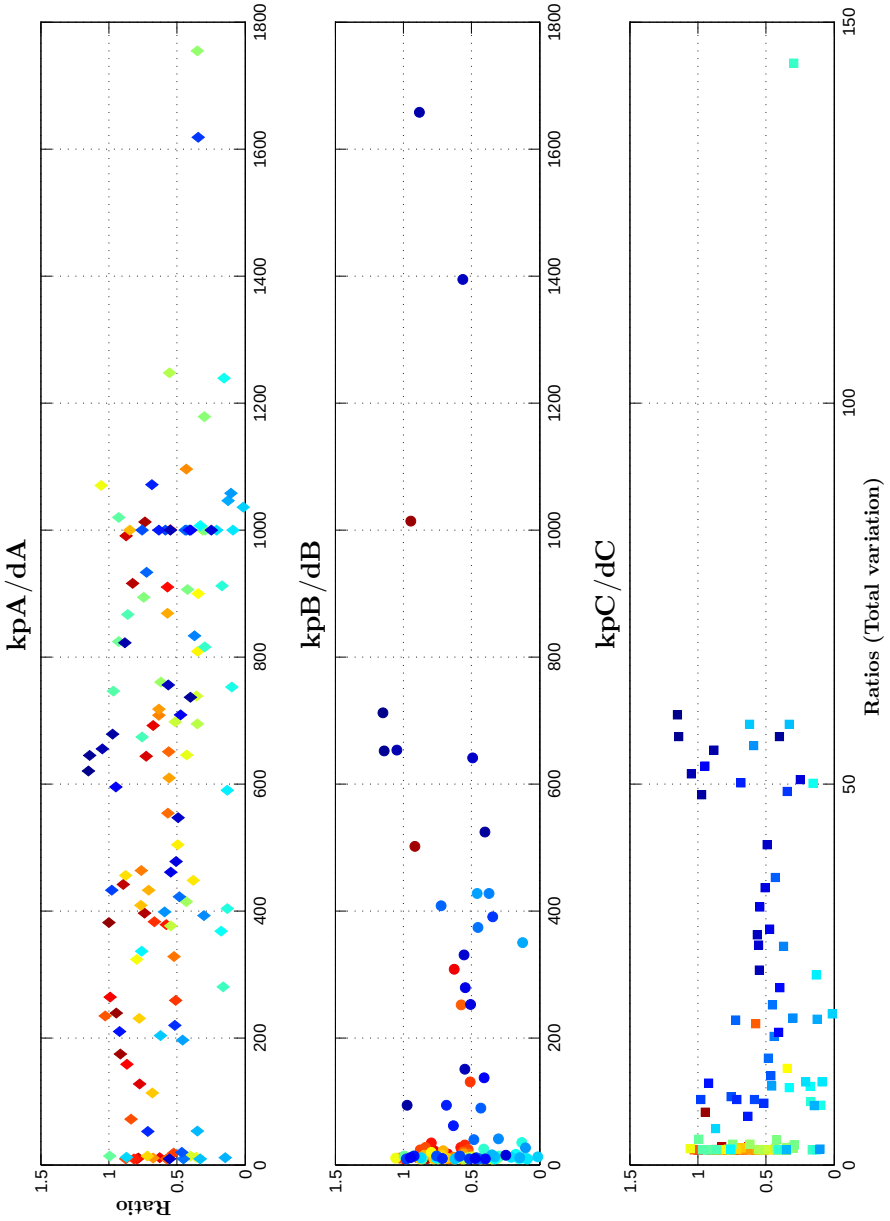

Figure 5.11: Translation/degradation protein proportions. A) k_{pA}/d_A rate takes significantly higher values than the others, but it doesn't affect directly the C sensitivity or precision. B) The quotient between the B protein expression and its degradation hardly varies with respect to the sensitivity and precision presented by the circuit while it keeps in the range shown. C) For a high sensitivity high values of k_{pC}/d_C are required too, whereas if the design target looks for a high precision, low values of k_{pC}/d_C are preferred.

5.4 ADDITIONAL ANALYSIS: MONTE-CARLO SAMPLING

This extra analysis was done to figure out if this biologic system was robust by itself, that is, with structural robustness. The goal is to see whether the circuit will always present adaptive behaviour –better or worse, but adaptive– irrespective of the parameters values. For this purpose, a Monte-Carlo sampling was carried out. Those parameters that showed variability inside the given range according to the optimizer, were fixed to an approximate mean value. Whereas the parameters which showed a strong tendency to relatively high or low values, were let to vary by means of the sampling simulation. This decision would give some points out of the optimal solution, since parameters which had to take certain restrictive values to be in the Pareto set were forced to take values far from those ones. The results of this approach are shown in Figure 5.12.

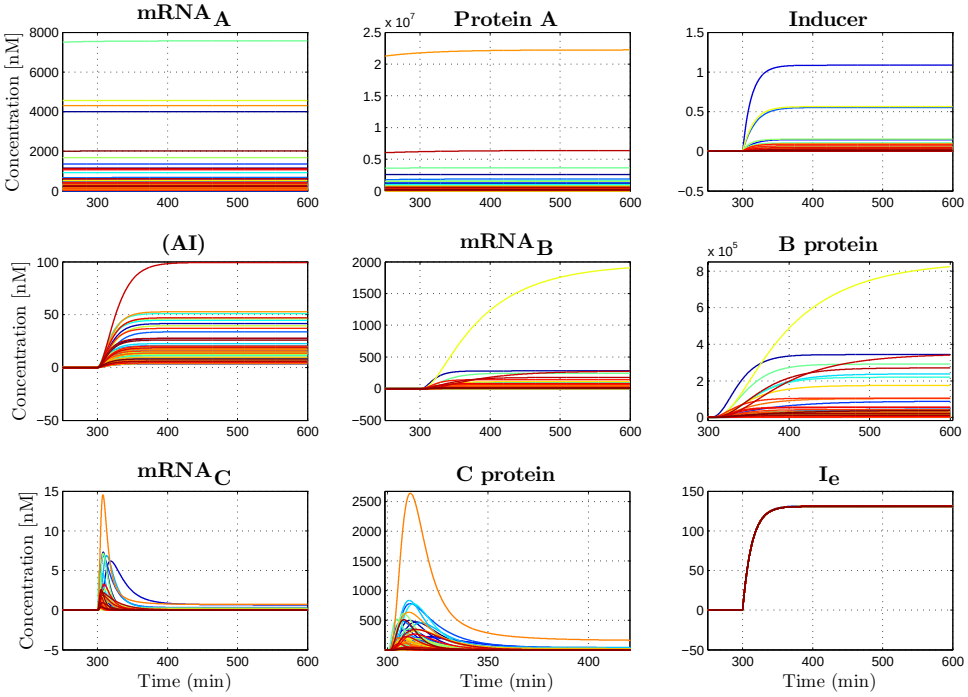

Figure 5.12: Model dynamic response with Monte-Carlo Sampling simulation. A wide variety of performances take place, with a significative number of them hardly returning to zero, i.e. presenting very poor performance. The optimizer did 30000 evaluations. 80 of them were sampled to check for their behaviour. Out of these 80, a significative number did not present adaptive behavior.

According to the results shown in the graph 5.12, it cannot be affirmed that the system has structural robustness. The optimizer did 30000 evaluations. 80 of them were sampled to check for their behaviour. Out of these 80, a significative number did not present adaptive behaviour. A larger sampling was made around the same level in the LD that the one of the optimizer. The results of this larger sampling are shown in the figure 5.14 in section

5.5 along with the Pareto Front provided by the optimizer to make the comparison more easy. These results confirm that the circuit looses the adaptation capability for values of the parameters out of an appropriate region. This result also confirms the need of using the MOO approach to solve the problem posed in this project.

5.5 DISCUSSION

In order to make a correct analysis, it is recommendable not just seeing the results of each of the two approaches separately but together. To this respect, differences and similarities are observed.

If we look at Figures 5.3, and5.8, we find the main differences in the parameters and ratios dA, $\frac{K_{mB} \cdot C_{gB}}{d_{mB}}$, dB, k_{d2} and k_{d3}. The main similarities are in the variability of $\frac{K_{mA} \cdot C_{gA}}{d_{mA}}$ and K_{pA}. This makes sense since the gene A is constitutive and should not be too enclosed *a priori*. Thus, in both approaches we observe high values of dC and d_{mC} (for faster degradation to perform adaptation), low values of K_{pB}, d_{AI} and d_{AI_2}, and finally, the same differentiation between red points and blue points for the parameters $K_{mC}C_{gC}$ and K_{pC}, (blue points with high values).

In order to get robust and adaptive behaviour, the θ values corresponding to the hybrid promoter tend strongly to certain values either when it is expected to maximize B, or minimize B, and even with no dependency on the desired criterion ('High sensitivity' or 'High precision'). θ_1 and θ_5 tend to high values inside their range. θ_2, θ_3 and θ_4 to values close to zero.

Analysing the transcription and translation equations at the equilibrium, the levels of mB and mC, and in turn the production levels of B and C, could be tuned by two distinguishable components: the gain process (defined by the reaction rates) and the mathematical characterization of the promoter (defined by the θ values and activator and repressor species).

Only the transcription and translation ratios of the protein C are a way to tune either high sensitivity or high precision. For better analysis of these relations see the Figure 5.13.

The Monte-Carlo sampling gave a set of random values for the parameters that made the system perform in different ways. In Figure 5.14 it is possible to see the representation of the Pareto Front for two objectives with the MOO (red line), among with the random sampling coloured in green and blue. The blue line results from the filtering of this random sampling, generating also a Pareto front for this randomly sampled points. Green points do not fulfil the *precision* pertinence of $J_2(\theta)$. Blue points, although present a bad sensitivity, let the system to respond to the perturbation giving little error (good precision). It can be appreciated that the Pareto front given by the optimizer is in a forward position with respect to the Pareto

front resulting from the filtering of the Monte-Carlo sampling points. This shows that the optimization process that we have used was necessary and effective.

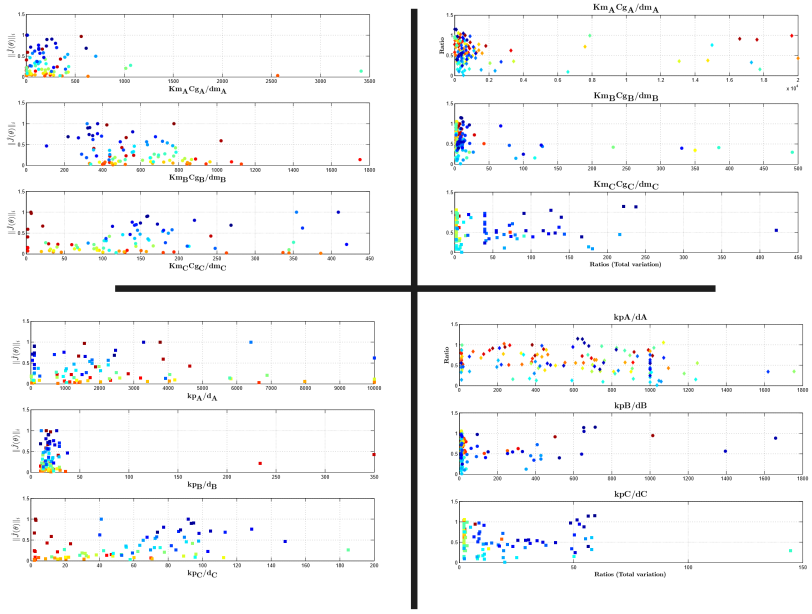

Figure 5.13: Left: transcription and translation ratios when maximizing B protein production. Right: transcription and translation ratios when minimizing B protein production.

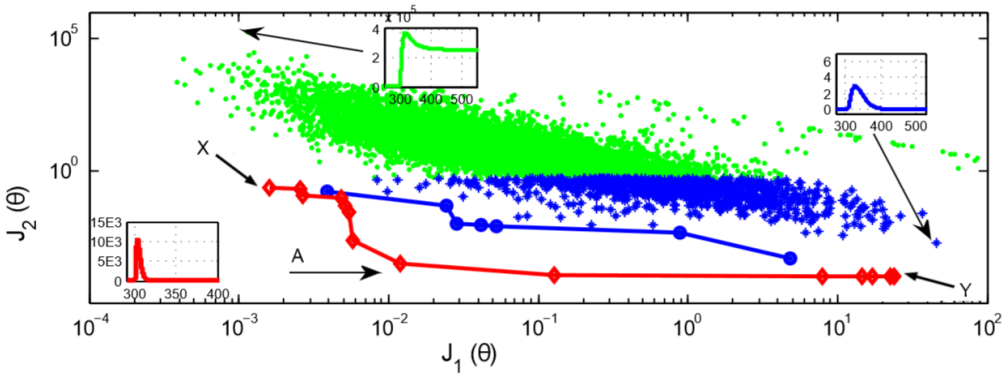

Figure 5.14: Pareto Front representation for two objectives obtained with the MOO (red line), along with the random sampling coloured in green and blue with its respective Pareto front located behind. Three responses of the C protein for three representative points are shown. Green points do not fulfil the precision *pertinence of $J_2(\theta)$. Blue points, although present a bad sensitivity, let the system to respond to the perturbation giving little error (good precision). Extreme points X and Y enclose the Pareto Front obtained. Point A presents a trade-off between both objectives.*

Finally, with a more mathematical perspective and the new results given by the optimizer, it is interesting to see how they affect the model equations. Let us analyse what happens with the hybrid promoter mathematical characterization. According to the optimizer, for the C protein to performe adaptation θ_2, θ_3 and θ_4 tend to very low values. However θ_3 and θ_4 tend to much lower values than θ_2. Then:

$$\frac{x_4}{\theta_2 + \theta_3 x_4 + \theta_4 x_6 + \theta_5 x_4 x_6} \simeq \frac{x_4}{\theta_2 + \theta_5 x_4 x_6} = \frac{1}{\frac{\theta_2}{x_4} + \theta_5 x_6}$$

if $\lim_{x_6 \to 0}$ (no repressor action) we have:

$$\lim_{x_6 \to 0} \frac{1}{\frac{\theta_2}{x_4} + \theta_5 x_6} = \frac{x_4}{\theta_2}$$

with θ_2 low. But notice that as soon as there is even a little amount of x_6, the effect of x_6 is significantly multiplied repressing in turn the transcription process, since θ_5 is very high.

5.5.1 Design principles

The capability of cells to extract and process information from their environment allows them to optimize their responses and their allocation of resources, thus bequeathing selective advantages to the organism. However, such an ability must necessarily be a robust property for it to be effective in the cell's noisy and uncertain environment. A key aim of systems biology is to identify the mechanisms through which robustness is achieved in cellular processes. Such sources of robustness can be identified through the analysis of models of biological systems as has been done in this work.

After the results discussion and by extracting the essential information from the graphs, the following tables are provided to see in an organized way the design principles derived in this work.

1. **Parameters range in order to maximize B**: see Table 5.3
2. **Parameters range in order to maximize B**: see Table 5.4
3. **Transcription and Translation Gain**: see Table 5.5

The ranges for these tables have been taken by rejecting *outliers* or values significantly distant from tendencies. The coloured arrows refer to the blue points and red points seen in the previous graphs, and the kind of tendency address the frequency of appearance to the pointed values.

Table 5.3: Parameters range in order to maximize B

Parameter	**Variation range by optimizer**	**Range selection criterion**		**Tendency**	
		High sensitivity	High precision		
KmACgA	[1 200]	[1 50]	[1 100]	$\longrightarrow$ $\longleftarrow$	Slight
KmBCgB	[1 200]	[1 150]	[100 200]	$\longleftarrow$ $\longrightarrow$	Slight
KmCCgC	[1 200]	[40 100]	[1 10]	$\longrightarrow$ $\longleftarrow$	Strong
k_d	0.0164 *fixed*				
d_A	[0.01 0.1]	[0.01 0.05]	[0.01 0.05]	$\longleftarrow$ $\longleftarrow$	Strong
d_B	[0.01 0.1]	[0.01 0.1]	[0.02 0.01]	$\longleftarrow$ $\longleftrightarrow$	Slight
d_C	[0.1 0.5]	[0.3 0.5]	[0.3 0.5][	$\longrightarrow$ $\longrightarrow$	Strong
θ_1	[200 600]	[500 600]	[500 600]	$\longrightarrow$ $\longrightarrow$	Strong
θ_2	[0.01 0.2]	[0.01 0.02]	[0.01 0.02]	$\longleftarrow$ $\longleftarrow$	Strong
θ_3	[1e-4 1]	[1e-4 0.02]	[1e-4 0.02]	$\longleftarrow$ $\longleftarrow$	Strong
θ_4	[5e-4 10]	[5e-4 2]	[2 l0]	$\longrightarrow$ $\longleftarrow$	Strong
θ_5	[0.1 10]	[6 10]	[6 10]	$\longrightarrow$ $\longrightarrow$	Strong
k_{pA}	[1 100]	[1 50]	[50 100]	$\longleftarrow$ $\longleftrightarrow$	Strong
k_{pB}	[1 100]	[1 2]	[1 2]	$\longleftarrow$ $\longleftarrow$	Strong
k_{pC}	[1 100]	[20 60]	[1 10]	$\longrightarrow$ $\longleftarrow$	Strong
k_2	[1 20]	[1 10]	[1 10]	$\longleftrightarrow$	Variable
k_3	[0.1 5]	-	-	$\longleftrightarrow$	Variable
k_{d2}	[100 250]	[160 250]	[160 250]	$\longrightarrow$ $\longrightarrow$	Strong
k_{d3}	[1 30]	[1 20]	[1 20]	$\longleftarrow$ $\longleftrightarrow$	Strong
d_{mA}	[0.01 0.5]	-	-	$\longleftrightarrow$	Variable
d_{mB}	[0.01 0.5]	-	-	$\longleftrightarrow$	Variable
d_{mC}	[0.01 0.5]	[0.4 0.5]	[0.4 0.5]	$\longrightarrow$ $\longrightarrow$	Strong
d_I	[0.001 0.5]	-	-	$\longleftrightarrow$	Variable
d_{Ie}	0.0164 *fixed*				
d_{AI}	[0.01 0.5]	[0.01 0.1]	[0.01 0.1]	$\longleftarrow$ $\longleftarrow$	Strong
d_{AI2}	[0.01 0.5]	[0.01 0.03]	[0.01 0.03]	$\longleftarrow$ $\longleftarrow$	Strong

Table 5.4: Parameters range in order to minimize B

Parameter	**Variation range by optimizer**	**Range selection criterion** High sensitivity	High precision	**Tendency**	
kmACgA	[1 200]	-	-	⟷	Variable
kmBCgB	[1 200]	[1 50]	[1 2]	⟵ ⟵	Strong
kmCCgC	[1 200]	[20 80]	[1 3]	⟷ ⟵	Strong
k_d	0.0164 *fixed*				
d_A	[0.01 0.1]	[0.08 0.1]	[0.08 0.1]	⟶ ⟶	Strong
d_B	[0.01 0.1]	[0.07 0.1]	[0.04 0.1]	⟶ ⟷	Strong
d_C	[0.1 0.5]	[0.4 0.5]	[0.4 0.5][	⟶ ⟶	Strong
θ_1	[200 600]	[450 600]	[450 600]	⟶ ⟶	Strong
θ_2	[0.01 0.2]	[0.01 0.02]	[0.01 0.02]	⟵ ⟵	Strong
θ_3	[1e-4 1]	[1e-4 0.0l]	[1e-4 0.0l]	⟵ ⟵	Strong
θ_4	[5e-4 10]	[5e-4 2]	[2 l0]	⟷ ⟷	Strong
θ_5	[0.1 10]	[8 10]	[9 10]	⟶ ⟶	Strong
k_{pA}	[1 100]	-	-	⟷ ⟷	Variable
k_{pB}	[1 100]	[1 70]	[1 18]	⟵ ⟵	Strong
k_{pC}	[1 100]	[3 30]	[1 4]	⟶ ⟵	Strong
k_2	[1 20]	-	-	⟷	Variable
k_3	[0.1 5]	-	-	⟷	Variable
k_{d2}	[100 250]	-	-	⟷	Variable
k_{d3}	[1 30]	[1 20]	[1 20]	⟷	Variable
d_{mA}	[0.01 0.5]	-	-	⟷	Variable
d_{mB}	[0.01 0.5]	-	-	⟵ ⟷	Slight
d_{mC}	[0.01 0.5]	[0.37 0.5]	[0.47 0.5]	⟶ ⟶	Strong
d_I	[0.001 0.5]	[0.001 0.3]	[0.001 0.3]	⟷	Variable
d_{Ie}	0.0164 *fixed*				
d_{AI}	[0.01 0.5]	[0.01 0.1]	[0.01 0.5]	⟵ ⟷	Strong
d_{AI2}	[0.01 0.5]	[0.01 0.05]	[0.01 0.3]	⟵ ⟷	Strong

Table 5.5: Transcription and Translation Gain

	Maximize		**Minimize**		**Tendency**	
Stage / **Node**	Transcription	Translation	Transcription	Translation	Color	Kind
gA	[0 500]	[0 10000]	[0 2000]	[0 1200]	-	Variable
gB	[200 1000]	[0 50]	[0 2000]	[0 400]	-	Variable
gC	[0 200]	[0 100]	[0 150]	[0 60]	⟵ ⟶	Strong

Chapter 6

Prototyping

Finding the biological species which can fit with the design principles derived from this work is not a trivial task.

On one hand, the task of get a set of genes whose products act as the desired way, repressing or activating each other according to the circuit that want to be implemented, (in this case a three-node Incoherent Feedforward Loop Type I), could result in a long search in biologic databases because of the many interactions among species and also the high diversity of species. And even though the search come up with a theoretical solution, another issue is that if it is wanted the circuit to empirically implement, it is necessary to have a mean to implement connect all parts (coding sequences). At this respect, it will be introduced the notions of *plasmids* (as vectors) and *biobricks* in next section.

On the other hand, once one has the set of genes that behaves following the dynamic model of the aforesaid motif, it could happen that the biologic parameters of these species don't fulfill with the derived design principles, and in turn not being able to get less or more pick height and error, or even perform adaptation.

If we have a look to the regions involved during the synthesis of a protein (Figure 6.1), we find that changing any of them (and other external factors not mentioned) could affect to reaction rates. So these regions mean a way of tuning the parameters model, depending on the chains or nucleotides sequences comprising this regions.

- **Coging region.**
  The coding region contain the production information about a protein, so it defines the **product** to synthesize, which is the principal target. Also the **degradation protein rate** depends fundamentally on this region.

- **Promoter.**
  The promoter is an specific DNA sequence. It includes the operator or *binding site*,

where a certain molecule binds in order to initiate (or repress when it is not constitutive) the transcription process. In that sense the promoter controls the starting of this process, playing a regulatory role. It determines in some way the **mRNA degradation rate** and **transcription rate**.

- **Ribosome Binding Site.**
  The Ribosome Binding Site (RBS) use to be referred to the mRNA region (see generalities of Central Dogma in 3.1), to which ribosomes can bind and initiate translation. Here the translation rate is fundamentally determined.

- **Terminator.**
  Terminators are genetic parts that usually occur at the end of a gene or operon and cause transcription to stop. Depending on the sequence, its efficiency changes.

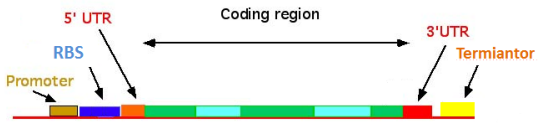

Figure 6.1: Gene relevant regions for the protein synthesis

6.1 IMPLEMENTATION

The way to implement a genetic circuit and introduce it in a cell (for instance prokaryotic bacteria cell like *E.coli*), is using *biobricks* as parts and *plasmids* as a vectors (which can also integrate a biobrick).

Biobricks are DNA sequences which have been standardized and conform to the BioBrick assembly standard [22].These Lego-like building blocks are used to design and assemble synthetic biological circuits, which would then be incorporated into living cells to construct new biological systems. Examples of BioBrick parts include promoters, ribosomal binding sites (RBS), coding sequences and terminators. A wide catalog can be found in `http://parts.igem.org/Main_Page`.

A **vector** is a plasmid into whose genome a fragment of foreign DNA is inserted; used to introduce foreign DNA into a host cell in the cloning of DNA. A **plasmid** is a circular, double-stranded unit of DNA that replicates within a cell independently of the chromosomal DNA and is most often found in bacteria; it is used in recombinant DNA research to transfer genes between cells.

The purpose of this last target goes about finding the suitable standard parts for the I1-FFL. As said before, finding the set of genes that fulfill the correct performance and integrates the specific genetic circuit is not a trivial task. For this reason, knowing about the existence of a previous work in which was empirically implemented this circuit with certain species [28] (but with other focus in its studies), this work will take some idea for the species to use.

After searching for this parts in the Registry of Standard Biological Parts, the selection of Biobricks is shown in the Table 6.1.

The three genes of the circuit:

LuxR : LuxR gene produce luxR proteins. This family consist of bacterial regulatory proteins, but also are in a variety of organisms. In the gene code there is a region often containing an autoinducer-binding domain in the N-terminal region. Most luxR-type regulators act as transcription activators, but some can be repressors or have a dual role for different sites. In this circuit it is wanted the luxR protein to bind to the autoinducer AHL (N-acyl-L-homoserine lactone), which will be added from cell's outside.

GFP-LVA : LVA tag consists of a short peptide sequence (AANDENYALVA) and is attached to the C-terminal end of GFP. Green fluorescent protein (GFP) is often used as a reporter protein, because it allows easy and nondestructive in situ monitoring of cellular processes. It can be expressed in a wide range of organisms and it does not require the addition of a special substrate in order to detect green fluorescence. Nevertheless, the protein has one major drawback, it seems to be very stable. Once the expression of GFP is started, it will fluoresce for a very long period of time. This makes GFP unsuitable for monitoring rapid changes in gene expression. [14] had already indicated that proper tagging of GFP will make the protein less stable. LVA tag seemed to be the most efficient tag to make GFP unstable.

CI-LVA : The standard name for this gene is *Phage lambda cI*. According to results of this work, protein degradation rate for gene B is desired to be from 0.04 to 0.1. In *E.coli* this rate is considered similar to protein dilution rate in the cell (0.035 min-1). So LVA tag is interesting to make higher the degradation rate.

Table 6.1: Biobricks or standard parts for the genes

Gen	Name	Promoter	RBS	Coding region	Terminator
gA	LuxR	BBa_J23106	BBa_B0034	BBa_C0062	BBa_B1006
gB	CI-LVA	BBa_R0062	BBa_B0034	BBa_K327018	BBa_B1006
gC	GFP-LVA	BBa_K415032	BBa_B0034	BBa_K18001	BBa_K259006

Some clarifications and justifications for this selection must be done:

- Gene A promoter was searched to be constitutive.
- Hybrid promoter for gene C was chosen among these other options: BBa_1751501 and BBa_I1051. These were rejected because of including two operators instead of one, as it was tried to resemble the part from [28].
- There are other terminators that could fit in, but it was tried to look for the more efficient ones.
- The Biobrick for RBS was selected because it has many times been used, according to the website where it was searched (up to 2935).
- All coding regions *parts* include the AUG starting codon.
- Also a plasmid is needed to connect properly the parts. The selected one was BBa_J64100

This is just a first approximation of IFF GNR implementation. The *Benchling* free software will be used for standard parts ensemble simulation. The future development of this work will probably bring modifications in this given prototyping format.

Bibliography

[1] Feed forward (control)[online]. http://en.wikipedia.org/wiki/Feed_forward_(control), note = Accessed: 05-07-2014.

[2] Network motifs [online]. http://homepages.ulb.ac.be/~dgonze/TEACHING/network_motifs.pdf, note = Accessed: 05-07-2014.

[3] Uri Alon. *An Introduction To systems biology. Design Principles of Biological Circuits*. Chapman & Hall/ CRC Mathematical and computational Biology Series, 2006.

[4] Jordan Ang, Brian Ingalls, and David McMillen. Probing the input-output behavior of biochemical and genetic systems system identification methods from control theory. *Methods Enzymol*, 487:279–317, 2011.

[5] X. Blasco, J.M. Herrero, J. Sanchis, and M. Martínez. A new graphical visualization of n-dimensional pareto front for decision-making in multiobjective optimization. *Information Sciences*, 178(20):3908 – 3924, 2008.

[6] Yadira Boada. Gene expression modelling and control: model reduction and noise approximation. Master's thesis, Universidad Politécnica de Valencia, 2013.

[7] A. Goldbeter D. E. Koshland and J. B. Stock. Amplification and adaptation in regulatory and sensory systems. *Science*, pages 220–225, 1982. 217(4556).

[8] X. Blasco J.M. Herrero. G. Reynoso-Meza, J. Sanchis. Evolutionary algorithms for multivariable pi controller design. 39:7895–7907, July 2012.

[9] Javier Sanchis Gilberto Reynoso-Meza, Sergio García-Nieto and F. Xavier Blasco. Controller tuning by means of multi-objective optimization algorithms: A global tuning framework. 2011.

[10] Javier Sanchis Miguel Martínez Gilberto Reynoso-Meza, Xavier Blasco. Controller tuning using evolutionary multi-objective optimisation: Current trends and applications. 2014. Instituto Universitario de Automática e Informática Industrial.

[11] Abdullah Hamadeh, Eduardo Sontag, and Brian Ingalls. Response time re-scaling and weberś law in adapting biological systems. In *American Control Conference (ACC)*, 2013.

[12] Shinagawa Higashi-Gotanda. pages 3–14–13. Sony Computer Science Laboratories Tokyo 141-0022, Japan.

[13] Christopher A Voig Jennifer A N Brophy. Principles of genetic circuit design. *Nature America*, 2014.

[14] Lars Kongsbak Poulsen Sara Petersen Bjørn Michael Givskov Jens Bo Andersen, Claus Sternberg and Søren Molin. New unstable variants of green fluorescent protein for studies of transient gene expression in bacteria. 1998.

[15] Marc W. Kirschneremail Uri Alonemail Lea Goentoro, Oren Shoval. The incoherent feedforward loop can provide fold-change detection in gene regulation.

[16] Rinaldi N. J. Robert F. Odom D. T. Bar-Joseph Z. Gerber G. K. Hannett N. M. Harbison C. T. Thompson C. M. Simon I. et al. Lee, T. I. *Science*, pages 298, 799, 2002.

[17] S. Mangan and U. Alon. Structure and function of the feed-forward loop network motif. 2003.

[18] R.T. Marler and J.S. Arora. Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization*, 26:369–395, 2004.

[19] Kaisa Miettinen. *Nonlinear Multiobjective Optimization*, volume 12. Kluwer Academic Publishers, Boston, 1999.

[20] Shen-Orr S. Itzkovitz S. Kashtan N. Chklovskii D. & Alon U. (2002) Milo, R. *Science*, pages 298, 824, 2002.

[21] Jesús Picó. *Modern circuits*, chapter 3, page 32. 2014.

[22] Drew Endy Reshma P Shetty and Thomas F Knight. Engineering biobrick vectors from biobrick parts. *Biological Engineering*, 2008.

[23] Gilberto Reynoso-Meza, Xavier Blasco, Javier Sanchis, and Juan M. Herrero. Comparison of design concepts in multi-criteria decision-making using level diagrams. *Information Sciences*, 221:124–141, 2013.

[24] Gilberto Reynoso-Meza, Javier Sanchis, Xavier Blasco, and Miguel Martínez. Design of continuous controllers using a multiobjective differential evolution algorithm with spherical pruning. *Applications of Evolutionary Computation*, pages 532–541, 2010.

[25] Milo R. Mangan S. & Alon U. Shen-Orr, S. S. *Nat. Genet. CrossRefMedlineWeb of Science*, pages 31, 64, 2002.

[26] Mangan S Alon U. Shen-Orr SS1, Milo R. Network motifs in the transcriptional regulation network of escherichia coli. 2002.

[27] Eduardo D. Sontag. Adaptation and regulation with signal detection implies internal model. 2002. Department of Mathematics, Rutgers University, New Brunswick, NJ 08903, USA.

[28] Stephan Thiberge Ming-Tang Chen Subhayu Basu, Rishabh Mehreja and Ron Weiss. Spatiotemporal control of gene expression with pulse-generating networks. 2004.

[29] Hana El-Samad-Wendell A. Lim Wenzhe Ma, Ala Trusina and Chao Tang. Defining network topologies that can achieve biochemical adaptation. *Cell*, 2009.

Part II

BUDGET

Table of contents

Chapter 1

Budget

1.1 INTRODUCTION

This document covers the estimated costs of the design, prototyping and implementation of an Incoherent Feedforward Genetic Regularory Network.

This budget is organized in the following way. Firstly manpower and material resources costs are defined and explained. Next, partial budgets of each Unit of Work are exposed and commented. The different Units of Work attend to three functional groups:

- **Theoretical analysis for design principles deduction**
  UW01 System modelling, simulations and discussion.
- **Documents edition**
  UW02 Project development.
- **Empirical implementation for prototypes**
  UW03 Obtaining the standard parts.
  UW04 Gibson Assembly.
  UW05 Final plasmid verification.

1.1.1 Manpower costs

In this section the costs associated to the staff employment are detailed. An Industrial Engineer and a Biotechnologist are required for the execution of this project.

The set of tasks executed by the Industrial Engineer include the modelling and theoretical analysis of the Genetic Regulatory Network, the text edition of this research, and finally the real implementation of the Genetic Regulatory Network in the laboratory in a interactively work with the Biotechnologist.

The set of tasks executed by the Biotechnologist include essentially the work in the laboratory, which correspond to the last three Units of Work mentioned above.

The three first task or Units of Work executed by the Industrial Engineer take three months (April, May and June), 5 days per week, 6 hours per day. The work in the laboratory lasts three weeks, six days per week, eight hours per day.

The measuring for the Industrial Engineer:

$$65days \cdot 6\frac{hours}{day} = 390hours$$
$$3weeks \cdot 6\frac{days}{week} \cdot 8\frac{hours}{day} = 144hours$$
$$390 + 144 = 534hours$$

The measuring for the Biotechnologist:

$$3weeks \cdot 6\frac{days}{week} \cdot 8\frac{hours}{day} = 144hours$$

Table 1.1: Manpower costs

Units	Concept	Working hours	Unitary price (€/h)	Income
1	Industrial Engineer	247		260.3
	Cost itemization:			
	Professional fees		10	
	Social Security		3.3	
				260.3
1	Biotechnologist	144		157.3
	Cost itemization:			
	Professional fees		10	
	Social Security		3.3	
				157.3
			TOTAL	417.6

1.1.2 Material resources costs

In this section the costs attached to the implementation of the Genetic Regulatory Network are listed and estimated.

For the acquirement of biological standard parts, costs are estimated in the *biobricks* delivery and, in case of factory defects, the eventual request of synthesis of certain standard parts (it has been estimated in two units). These standard parts and primers sequences are calculated according to the price of the pair of bases.

The eventual synthesis would correspond to 100 pb per part or sequence. Primers for Gibson Assembly are estimated in 40 pb. The group of standard parts is comprised by 3 promoters, 3 coding regions and 2 terminators. In sum are 8 parts to take into account, 2 primers per part are needed. So for Gibson Assembly are needed 16 primers. Primers for final plasmid sequencing are estimated in 20 pb. It is supposed to have two final plasmids. Since 2 primers for plasmid duplication is needed, that results in 4 primers.

A unit of purified base by HPSF (0.01 µmol) cost 0.49 euro. The respective measuring for each sequence is as follows.

Eventual synthesis of standard parts:

$$100pb \cdot 0.49\frac{euro}{pb} \cdot 2 = 98euro$$

Primers for Gibson Assembly:

$$40pb \cdot 0.49\frac{euro}{pb} \cdot 16 = 313.6euro$$

Primers for final plasmid sequencing:

$$20pb \cdot 0.49\frac{euro}{pb} \cdot 4 = 39.2euro$$

The equipment associated to an active laboratory has been included in the *fungible resources*, because of its difficulty of quantifying. It covers pipettes, microtubes, Petri dishes, cultivation pipes, restriction enzymes, PCR reactants (dNTPs, Polymerasa), etc. As an approximation, it has been considered to be the 30% of the sum of operations costs in each of the Units of Work.

UW03 operations/processes:

Eventual synthesis of standard parts.
Plasmid purification (Miniprep).

UW04 operations/processes:
Parts purification of PCR.

UW05 operations/processes:
Plasmid purification (Miniprep).
Plasmid verification.

Table 1.2: Material resources costs

Unit of Work	Concept	Measuring	Unitary price (€)	Income
UW03	Biobrick delivery	1	100	100
UW03	Eventual synthesis of standard parts	2	49	98
UW04	Primers for Gibson Assembly	16	19.6	313.6
UW04	Gibson Cloning Kit reactant	2	20	40
UW05	Primers for sequencing	4	9.8	39.2
	Fungible laboratory equipment (30%)	3		47.64
			TOTAL	638.44

1.2 PARTIAL BUDGETS PER UNIT OF WORK

Table 1.3: UW01 System modelling, simulations and discussion

Ud	Description	Measuring	Price	Income
	1. Theoretical analysis for design principles deduction			
h	Industrial Engineer Graduate	240	13.33	3199.2
			TOTAL	3199.2

Table 1.4: UW02 Project development

Ud	Description	Measuring	Price	Income
	2. Project development			
h	Industrial Engineer Graduate	144	13.33	1919.52
			TOTAL	1919.52

Table 1.5: UW03 Obtaining the standard parts

Ud	Description	Measuring	Price	Income
	4. Obtaining the standard parts			
h	Industrial Engineer Graduate	48	13.33	639.84
h	Biotechnologic Postgraduate	48	13.33	639.84
u	Biobrick delivery	1	100	100
u	Eventual synthesis of standard parts	2	49	98
u	Plasmid purification (Miniprep)	8	1.8	14.4
	Fungible laboratory equipment (30%)			33.72
			TOTAL	1492.08

Table 1.6: UW04 Gibson Assembly

Ud	Description	Measuring	Price	Income
	5. Gibson Assembly			
h	Industrial Engineer Graduate	48	13.33	639.84
h	Biotechnologic Postgraduate	48	13.33	639.84
u	Primers	16	19.6	313.6
u	Gibson Cloning Kit reactant	2	20	40
u	Parts purification of PCR	8	1.6	12.8
	Fungible laboratory equipment (30%)			3.84
			TOTAL	1646.08

Table 1.7: UW05 Final plasmid verification costs

Ud	Description	Measuring	Price	Income
	6. Final plasmid verification			
h	Industrial Engineer	48	13.33	639.84
h	Biotechnologist	48	13.33	639.84
u	Plasmid purification (Miniprep)	2	1.8	3.6
u	Plasmid verification	4	7.5	30
u	Primers for sequences	4	9.8	39.2
	Fungible laboratory equipment (30%)			7.28
			TOTAL	1359.76

1.3 FINAL BUDGET

The total budget of this project is obtained from the sum of each partial budget and the recharge of the VAT tax.

Table 1.8: Total budget

01	Theoretical analysis for design principles deduction	3199.2
02	Documents edition	1999.5
03	Empirical implementation for prototypes	4497.92
	Total Material Execution Budget	9696.62
	21% VAT	2036.29
	Total Budget	11732.92

The total budget rises to ELEVEN THOUSAND SEVEN HUNDRED AND THIRTY TWO EURO WITH NINETY TWO CENTS.