UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Master Dissertation

Integration of a metabolic burden model for the computational analysis of a synthetic genetic circuit

Liliana Gricel Oleas Palacios

Advisor: Prof. Jesús Picó

Co-advisor: Yadira Boada

Valencia,
January 11, 2018

Institut Universitari d'Automàtica i Informàtica Industrial
Universitat Politècnica de València

Dedicated to:
Dina and Gricelda,
you are my energy source.

My special appreciation to:
Prof. Jesús Picó for his time dedicated to supervision of this work,
Yadira Boada and Alejandro Vignoni for your assistance and direction in the development of this project and yours helpful comments on the manuscript,
thank you to everyone who supported me.

Abstract

The cells are systems with a great complexity and high number of interactions from diverse nature, they have several component proteins and genes. The understanding of these interactions is important, since they are those that regulate the fundamental cellular processes. The study of interactions between genes and proteins based on the design and construction of synthetic gene circuits, is contributing to comprehend the relation between the coordinated activity of certain genes and cellular functions. This has given rise to the birth of the so-called *Synthetic Biology (SB)*. The research group *Control of Complex Systems Group (CCSG) of Instituto de Automática e Informática Industrial (ai2)* is working on the design of optimal synthetic gene circuits. In this work we used a gene circuit known as type-1 incoherent feed-forward (I1-FFL) network to express a protein. This circuit presents an interesting biological feature called *Adaptation*. Adaptation is the characteristic of the biological systems to respond to a change in their input and return to the value that had been prior to the stimulus, even when the change of the input persists. Previous models are not able to capture the experimental results of I1-FFL. Here we combined the previous I1-FFL dynamic model with another mechanistic model to capture the whole experimental behavior. This new model explains in the better way how cells extract resources from their environment for protein synthesis. The parameters of this new model were optimized using a multi-objective optimization approach. We used a multi-objective optimization software as a good alternative to tackle a multi-objective problem. We performed a multi-objective optimization and designed the proper objectives indices to tune the model parameters taking into account the desired behavior (adaptation) and the experimental information. We obtained a model of the synthetic gene network that includes the effects of the metabolic burden. The model that we obtained together with its tuned parameters can be used in further applications such as model identification with experimental data and to analyze the effect of the over-expression of synthetic proteins in the growth rate of the microorganisms.

Resumen

Las células son sistemas de gran complejidad con un alto número de interacciones de diversa naturaleza entre sus componentes genes y proteínas. El estudio de las interacciones entre genes y proteńas basado en el diseo y construccin del gen sintético, está contribuyendo a comprender la relación entre la actividad coordinada de ciertos genes y las funciones celulares. Esto ha dado lugar al nacimiento de la llamada Biología Sintética (SB). El grupo de investigación Control of Complex Systems Group (CCSG) del Instituto de Automática e Informática Industrial (ai2) está trabajando en el diseo óptimos de circuitos géneticos sintéticos. En este trabajo utilizamos un circuito genético conocido como red de feed-forward incoherente tipo 1 (I1-FFL) para la expresión de proteínas. Este circuito presenta una interesante característica biológica llamada adaptación. La adaptación es la característica de los sistemas biológicos para responder a un cambio en su entrada y volver al valor anterior al estímulo, incluso cuando el cambio de la entrada persiste. Los modelos anteriormente utilizados no son capaces de capturar los resultados experimentales de I1-FFL. Aquí combinamos el anterior modelo dinámico I1-FFL con un modelo mecanicista que incorpora los procesos de traducción explicitamente para capturar todo el comportamiento experimental. Este nuevo modelo explica mejor cómo las células extraen recursos de su entorno para la síntesis de proteínas. Los parámetros de este nuevo modelo se optimizan utilizando un enfoque de optimización multiobjetivo. Utilizamos optimización multiobjetivo como una buena alternativa para abordar un problema multiobjetivo. Realizamos una optimización multiobjetivo y diseamos los ndices de objetivos adecuados para afinar los parámetros del modelo tomando en cuenta el comportamiento deseado (adaptación) y la información experimental. Como resultado se obtuvo un modelo de la red genética sintética que incluye los efectos de la carga metabólica. El modelo obtenido junto con sus parámetros ajustados puede ser utilizado en otras aplicaciones como la identificación de modelos con datos experimentales o para analizar el efecto de la sobreexpresión de proteínas sintéticas en la tasa de crecimiento de los microorganismos.

Contents

List of Figures

List of Tables

1

Introduction

1.1 Introduction

The cells are systems with a great complexity with high number of interactions from diverse nature. Between their component proteins and genes. The understanding of these interactions is important, since they are those that regulate the fundamental cellular processes. Many years of experimentation have been necessary to keep awake the molecular bases of these processes. The study of interactions between genes and proteins based on the design and construction of synthetic gene, is contributing to comprehend the relation between the coordinated activity of certain genes and cellular functions. This has given rise to the birth of the so-called *Synthetic Biology (SB)*, where Nonlinear dynamics, Physics of complex systems, Engineering and molecular biology play an important role.

SB is a new frontier in biological research where scientists and engineers design and create living systems to carry out useful tasks. SB re-assembles basic units of biochemistry into synthetic gene networks, using platform technologies such as DNA synthesis, genome engineering, simulation tools, and computer-aided-design.

One of the objectives of synthetic genetic networks is to reproduce, through living organisms, that fulfill a certain function. The methodology for designing new synthetic genetic networks is clear: first, the function to be performed by genetic network must be chosen, and then the elements of this network can be selected. For example, if we want to build a genetic network that oscillates, we must analyze what elements are necessary to produce oscillations in a simplest possible way.

In order to study biological processes, we have resorted to the design and construction of artificial regulation networks. These synthetic networks can contribute to the clarification of molecular basis for a given function. Synthetic genetic networks can be constructed in the laboratory and mathematically modeled, which also allows qualitative analysis to be carried out using numerical simulations. The modelling process itself results in hypothesis to be experimentally

tested, thereby iteratively producing refined models and insight about cellular mechanisms.

One of the objectives of the SB is to design genetic circuits using engineering criteria to find a solution that meets design specifications. Some of these requirements are mutually exclusive, in other words one requirement is improve, the other one is getting worst.This issue can be analyzed as the multi-objective optimization (MOP) problem.

In this work we use a reduced mathematical model of a synthetic three-node genetic circuit that presents adaptation. Adaptation is the ability of genetic circuits to respond to a change in their input and return to the value that was held prior to the stimulus, even when the input change persists. Then we integrated this model with a mechanistic metabolic model thet includes the cellular machinery such as house-keeping proteins energy, transport proteins and ribosomes to evaluate the effect of changes in the cellular environment.

This new model is a high dimmensional one, which requires finding the proper values of the parameters. We used a multi-objective optimization as a good alternative to tackle the problem of finding parameters compatible with the desired behaviour of the circuit and the environmental constrains of the cell.

1.2 Motivation

The Control of Complex Systems Group (CCSG) of Instituto de Automtica e Informtica Industrial (ai2) is working on the design of optimal synthetic gene circuits. In this work we used a gene circuit known as type-1 incoherent feed-forward (I1-FFL) network to protein expression. This circuit presents an interesting biological feature called adaptation. Previous models are not able to capture the experimental results of I1-FFL. Here we combined the previous I1-FFL dynamic model with another mechanistic model to capture the whole experimental behavior. This new model explains in the better way how cells extract resources from their environment for protein synthesis. We used the mechanistic model from [16].

1.3 Outline

The structure of this work goes as follows:

Chapter 2: An overview of the Central Dogma of molecular biology.

Chapter 3: The dynamic and metabolic model of the proposed gene circuit is described.

Chapter 4: The multi-objective optimization of model parameters is performed.

Chapter 5: The simulations and results are presented.

Chapter 6: Conclusions and future work.

2

Gene expression in gene synthetic networks

2.1 Introduction

Gene expression refers to the process of producing a specific and controlled amount of a protein in a spatio-temporal manner. Inside the cell, gene expression is highly regulated in many steps including: transcription phase, translation phase, export and degradation. Here, we define broadly these steps as a *Central dogma of molecular biology* enunciated by Crick in 1958.

Some biological problems demand a model that can represent a multicomponent, temporally evolving dynamics system [2]. In these terms, differential equation models come to the fore and the regulatory networks can be represented by ordinary differential equations (ODEs). These equations depend of a set of variables such as mRNAs and proteins concentrations, parameters, and rates that change in the time.

2.2 The Central Dogma of molecular biology

DNA (deoxyribonucleic acid) can be subdivided into segments of information called genes. They contain coded instructions for the production of a single protein with a specialized function inside each cell. Reading a gene and producing a chain of amino acids to generate a protein requires that the cells perform gene expression in two stages: transcription and translation [3]. This process is known as the *Central Dogma* of molecular biology, see Fig. 2.1.

The Central Dogma two three fundamental parts as follows.

1. **Transcription**: this process is triggered by the binding of RNA Polymerase to a determined region in DNA called promoter, see Fig. 2.3b. The promoter can be activated or inhibited by a range of promoter-specific proteins called **transcription factors** (see Fig. 2.2).

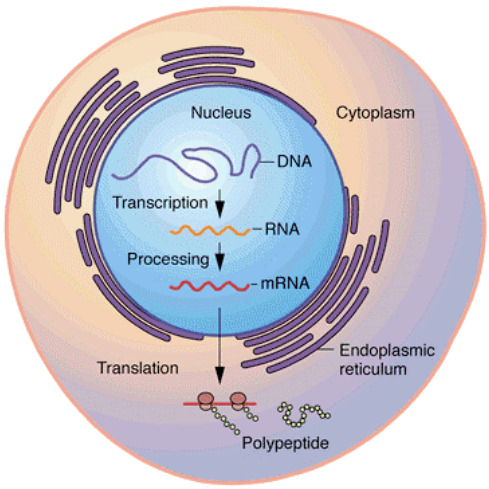

Figure 2.1: Two major stages of the Central Dogma: Transcription from a DNA molecule to a mRNA one, and Translation from a mRNA molecule to protein one.

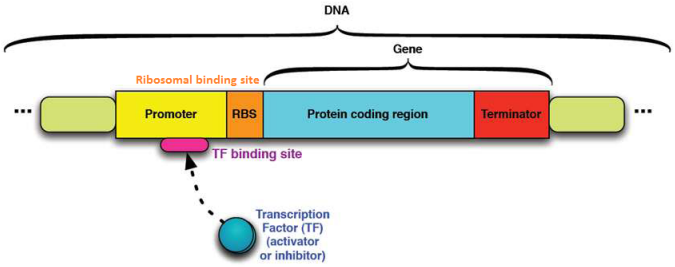

Figure 2.2: Transcription factors with the promoter region, inhibit or enable the transcription process. Source: Master dissertation *Gene expression modeling and simulation. Model reduction and noise approximation*, Yadira Boada, UPV, 2013.

This binding complements sequence of DNA after the replication, in other words, the template strands is complemented and generate the *messenger ribonucleic acid* (mRNA). mRNA 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 Fig. 2.3d. In eucaryotic 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 into the cytoplasm and the Translation starts. In prokaryotic cells, there is no nucleus and transcription and translation occurs in the same place: the cytoplasm.

2. **Translation**: one molecule of a protein is produced by ribosomes that bind to a specific site of mRNA called ribosomal binding site (RBS). Then, ribosomal units move along the

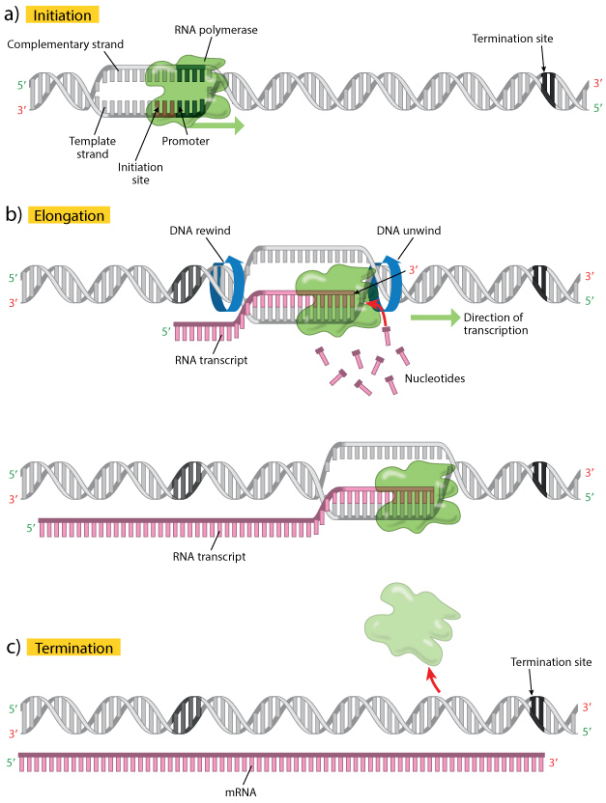

Figure 2.3: (a) The transcription process is initiated when the enzyme RNA polymerase binds to a DNA template at a promoter sequence. (b) During the elongation process, the DNA double helix unwinds. RNA polymerase reads the template DNA strand and adds nucleotides to the three-prime end of a growing RNA transcript. (c) When RNA polymerase reaches a termination sequence on the DNA template strand, transcription is terminated and both the mRNA and the RNA polymerase are released from the complex.

mRNA chain converting triplets of bases or "codons" on the mRNA, into a chain of amino acids defining the desired protein. Translation continues until the new amino acids chain is released from the ribosome as a complete protein, see Fig. 2.4.

Therefore, Transcription and Translation processes are known as ***gene expression*** and they can be unregulated and regulated, see Fig. 2.5. Unregulated expression is called *constitutive gene expression* whereas regulated expression is called *gene transcription regulation*. The following sections describe in more detail these two types of gene expression.

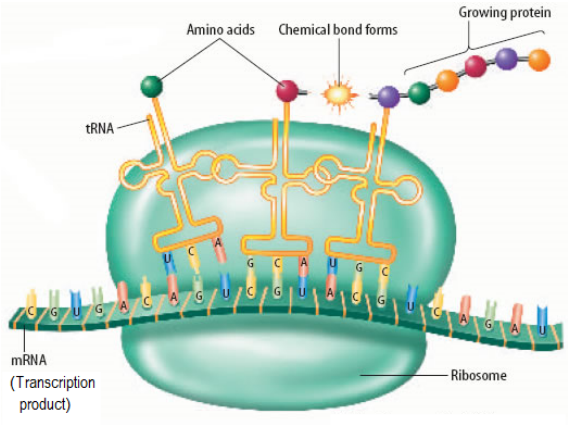

Figure 2.4: In translation, mRNA along with transfer RNA (tRNA) and ribosomes work together to produce one molecule of a protein.

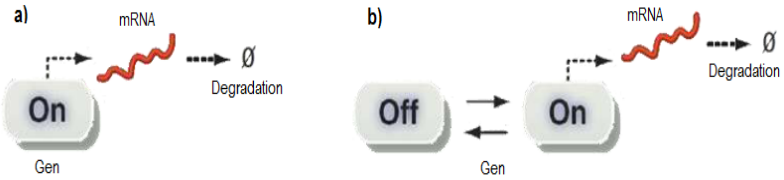

Figure 2.5: Gene expression can be: a) Unregulated or constitutive gene expression, the gen is always on, b) Regulated expression or gene transcription regulation is controlled by some proteins.

2.3 Constitutive gene expression

The Central Dogma can be summarized as biological reactions that work at different rates of product and degradation between the three principal elements: gene, mRNA, and protein.

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

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

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

Here, gene expression is unregulated or *constitutive* because the gene is always on. Using the law of mass action kinetics [3], the ODE's set for constitutive expression is given as:

$$\dot{m} = k_1 - d_1 m \tag{2.1}$$

$$\dot{p} = k_2 m - d_2 p \tag{2.2}$$

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 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) first term corresponds to the tendency of the protein to break down per unit of time, ii) 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 outside signals (DNA-binding proteins called transcriptions factors, metabolites, temperature, etc.) as discussed in the next section.

2.4 Gene transcription regulation

Transcription factors are transcription regulation proteins which can bind to determined sections of the promoter called *transcription factors binding sites*. They inhibit or active the transcription of a gene.

2.4.1 Gene transcription regulation by activators

The transcription level is **activated** by the cooperative binding of activators to the transcription factor binding site of the gene.

$$\text{Gene} \xrightarrow{\overset{\text{Activator}}{\downarrow}} \underset{\downarrow}{\text{mRNA}} \longrightarrow \underset{\downarrow}{\text{Protein}}$$
$$\qquad\qquad\qquad\quad \emptyset \qquad\qquad\quad \emptyset$$

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

$$\dot{m} = k_1 \frac{A^n}{K^n + A^n} - d_1 m \tag{2.3}$$

$$\dot{p} = k_2 m - d_2 p \tag{2.4}$$

where m, p and A are the mRNA, protein and activator concentrations respectively, k_1 is the maximal transcription rate, K is the activation coefficient, n is the *Hill* coefficient (number of activators that need to bind the promoter to trigger the activation of gene expression).

2.4.2 Gene transcription regulation by repressors

When the transcription is **repressed** by the cooperative binding of repressors to the transcription factor binding site, we say that the transcription factor is a repressor. For a repressor, the Hill function (see Fig. 2.6b) decreases from its maximal level of product concentration to the lowest level of concentration.

$$\text{Gene} \xrightarrow{\text{Repressor}} \text{mRNA} \longrightarrow \text{Protein}$$
$$\text{mRNA} \downarrow \emptyset \qquad \text{Protein} \downarrow \emptyset$$

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

$$\dot{m} = k_1 \frac{K^n}{K^n + R^n} - d_1 m \tag{2.5}$$

$$\dot{p} = k_2 m - d_2 p \tag{2.6}$$

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).

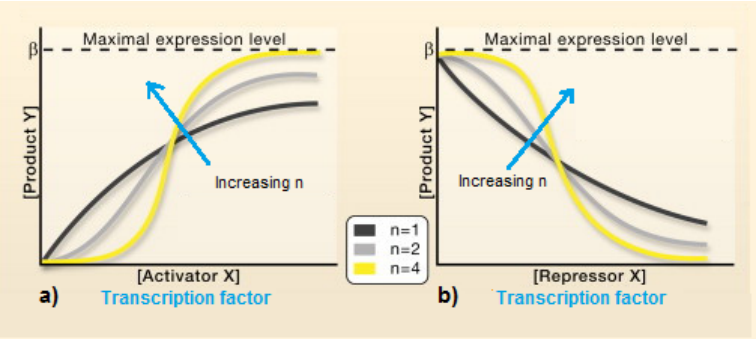

Figure 2.6: Hill function forms for transcription factors as (a) activators and (b) repressors.

When several transcription factors affect a gene, interacting each other in a nonlinear way, new behaviours occur [15]. These behaviors fall outside the scope of this work.

3

Dynamic model of synthetic gene circuit including metabolic burden

3.1 Adaptation in genetic circuits

*A*daptation is the characteristic of the biological systems to respond to a change in their input and return to the value that had been prior to the stimulus, even when the change of the input persists. Fig.3.1 shows the description of two indices to obtain adaptation: sensitivity and precision. The sensitivity measures the change (magnitude of the peak) of the output after the stimulation, while the precision detects if the stimulated output returned to the value before the change in the input, that is, the error of the output.

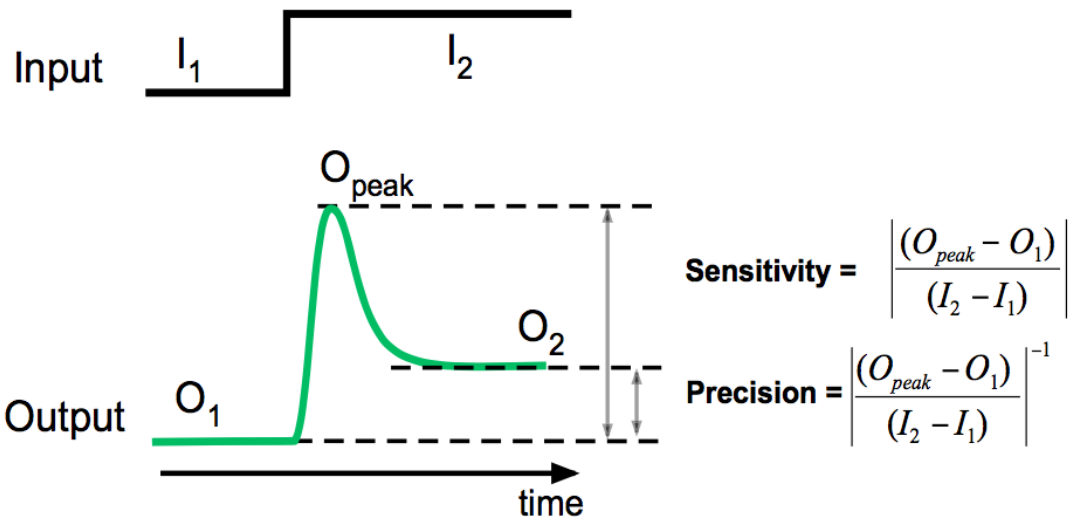

Figure 3.1: Graphical and mathematical representation of the adaptation of a genetic circuit.

3.2 Incoherent type-1 feed-forward loop (I1-FFL) network motif

The I1-FFL is one the most common network motifs. Different implementations are possible, including enzyme reactions networks, gene networks and in vitro transcriptional networks. In this case we used as an adaptive genetic network implementation. The adaptation is defined as the ability of biological circuits to respond to a change in its input and return to the value it had prior to the stimulus.

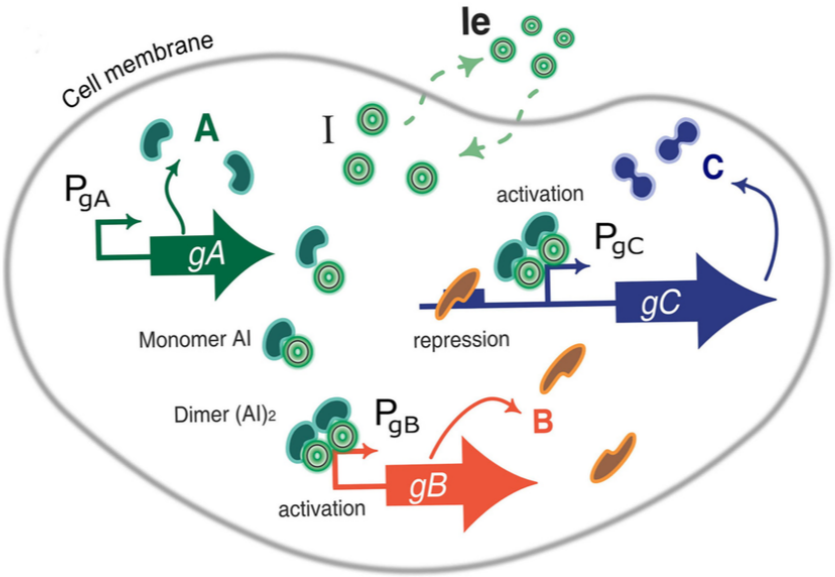

Figure 3.2: Cell representation incorporating an incoherent feed-forward loop synthetic circuit.

In the I1-FFL circuit (Fig.3.2) the protein A acts as a transcription factor and actives the expression of two genes B and C. In turn protein B represses expression of gene C. Therefore the protein A directly actives and indirectly represses gene expression of gene C .

The synthetic network consists of three genes gA, gB and gC that code to different proteins A, B and C. The external inductor molecule I_e is able to diffuse across the cell membrane, which allows to modify its concentration and the concentration of the internal inductor I. The protein A, which is the product of gene A, binds to the inducer I and forms the monomer (A.I). In turn, (A.I) dimerizes into the dimer $(A.I)_2$. This is the transcription factor that activates expression of gene C directly, and represses it indirectly via activation of the repressor 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.

This circuit was modeled using a deterministic approach by taking into account the key regulatory interactions between the main biochemical species. The circuit has the gene gC under the control

of the promoter PgC. The concentration of protein C is the output signal of the circuit.The expression of C is activated by the dimer $(A.I)_2$ that acts as transcription factor of the hybrid promoter PgC, and also PgC is repressed by the protein B. The dimer $(A.I)_2$ also acts as activator of the promoter PgB. The protein A is constitutively expressed. The inducer I passively can diffuse across the cell membrane. Finally the input signal of the circuit is the extracellular inducer concentration Ie. The last biochemical reactions are shown in the Fig.3.3 and listed in the table 3.1.

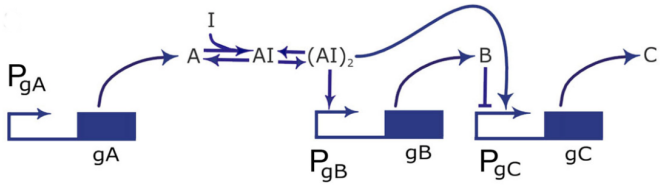

Figure 3.3: Incoherent type 1 feed-forward loop structure (I1-FFL).

Variable	Notation	Description
Inducer	$I->AHL$	N-Acyl homoserine lactone inside of cells.
	$Ie->AHLext$	N-Acyl homoserine lactone outside of cells input system.
Gene	$gA->luxR$	Luminescence gen
	$gB->cI$	Gen codes the repressor protein,cI
	$gC->gpf$	Gen codes GFP(Green Fluorescent Protein) protein.
Protein	$A->LuxR$	protein LuxR.
	$B->cI$	Repressor protein.
	$C->GFP$	Green Fluorescent Protein(GFP), output system.
Transcription factor	$(A.I)_2->(LuxR.AHL)_2$	Dimer formed after binding of 2 molecules of LuxR and AHL.
Promoter	$PgA->Pconst$	constitutive promoter.
	$PgB->Plux$	repressor promoter of cI.
	$PgC->Plux/cI$	hybrid promoter of the GFP.

Table 3.1: Componentes used in the I1-FFL synthetic network.

Finally, the reactions of our circuit goes as follows:

- The reactions corresponding to the gene A are the following:

$$gA + RNA_p \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} gA.RNA_p$$
$$gA.RNA_p \xrightarrow{km_A} gA + RNA_p + mA$$
$$mA \xrightarrow{kp_A} mA + A$$
$$mA \xrightarrow{dm_A} \emptyset$$
$$A \xrightarrow{d_A} \emptyset$$
$$A + I \underset{k_{-2}}{\overset{k_2}{\rightleftharpoons}} AI$$
$$\emptyset \xrightarrow{k_e} I_e$$
$$I_e \underset{k_{-d}}{\overset{k_d}{\rightleftharpoons}} I$$
$$I \xrightarrow{d_I} \emptyset$$
$$2(AI) \underset{k_{-3}}{\overset{k_3}{\rightleftharpoons}} (AI)_2$$
$$(AI) \xrightarrow{d_{AI}} \emptyset$$
$$(AI)_2 \xrightarrow{d_{AI_2}} \emptyset,$$

- those corresponding to the gene B:

$$gB + (AI)_2 \underset{k_{-4}}{\overset{k_4}{\rightleftharpoons}} gB.(AI)_2$$
$$gB.(AI)_2 + RNA_p \underset{k_{-7}}{\overset{k_7}{\rightleftharpoons}} gB.(AI)_2.RNA_p$$
$$gB.(AI)_2.RNA_p \xrightarrow{km_B} gB.(AI)_2 + RNA_p + mB$$
$$mB \xrightarrow{kp_B} mB + B$$
$$mB \xrightarrow{dm_B} \emptyset$$
$$B \xrightarrow{d_B} \emptyset,$$

- finally the corresponding reactions to the gene C:

$$gC + (AI)_2 \underset{k_{-5}}{\overset{k_5}{\rightleftharpoons}} gC.(AI)_2$$

$$gC.B + (AI)_2 \underset{k_{-6}}{\overset{k_6}{\rightleftharpoons}} gC.B.(AI)_2$$

$$gC.(AI)_2 + B \underset{k_{-8}}{\overset{k_8}{\rightleftharpoons}} gC.B.(AI)_2$$

$$gC + B \underset{k_{-9}}{\overset{k_9}{\rightleftharpoons}} gC.B$$

$$gC.(AI)_2 + RNA_p \underset{k_{-10}}{\overset{k_{10}}{\rightleftharpoons}} gC.(AI)_2.RNA_p$$

$$gC.(AI)_2.RNA_p \overset{km_C}{\longrightarrow} gC.(AI)_2 + RNA_p + mC$$

$$mC \overset{kp_C}{\longrightarrow} mC + C$$

$$mC \overset{dm_C}{\longrightarrow} \emptyset$$

$$C \overset{d_C}{\longrightarrow} \emptyset.$$

3.3 Model of I1-FFL synthetic gene network

We obtained a complete model based on the law of mass action kinetics [3]. Then this model was reduced using the Quasi steady-state approximation (QSSA)[17] and invariance principle on the conserved moieties. The next is the reduced model that represents the dynamics of our gene circuit:

$$\begin{aligned}
\dot{x_1} &= k_{mA}C_{gA} - d_{mA}x_1 \\
\dot{x_2} &= k_{pA}x_1 - d_A x_2 - k_2 x_2 x_3 + k_{-2}M \\
\dot{x_3} &= -k_2 x_2 x_3 + k_{-2}M + K_d(x_9 - x_3) - d_I x_3 \\
\dot{x_4} &= k_3 M^2 - k_{-3}x_4 - d_{AI2}x_4 \\
\dot{x_5} &= k_{mB}C_{gB}\frac{x_4}{\gamma_1 + x_4} - d_{mB}x_5 \\
\dot{x_6} &= k_{pB}x5 - d_B x_6 \\
\dot{x_7} &= k_{mC}C_{gC}\frac{x_4 + \beta_1\gamma_4 x_6 + \beta_2\gamma_5 x_4 x_6}{\gamma_2 + \gamma_3 x_4\gamma_4 x_6 + \gamma_5 x_4 x_6} - d_{mC}x_7 \\
\dot{x_8} &= k_{pC}x7 - d_C x_8 \\
\dot{x_9} &= k_{cells}k_d(-x_9 + x_3) - d_{Ie}x_9 \\
M &= -\frac{d_{AI} + k_{-2}}{4k_3} + \frac{1}{4k_3}\sqrt{(d_{AI} + k{-2})^2 + 8k_3(k_2 x_2 x_3 + 2k_{-3}x_4)}
\end{aligned} \tag{3.1}$$

where M is the monomer $A.I$ concentration, and $k_{cells} = \frac{V_{cell}N_{cells}}{V_{medium}}$ is required to take in to

account the concentration outside the cells. Note that the transport term $(x_3 - x_9)$, depends only on the difference of the concentrations inside and outside the cells. The k_{cells} constant reflects the amount that goes out (or in, depending on the sign) from all the cells into the extracellular volume and vice-versa. We used $V_{cell} = 110^{-15}L$, which is the typical volume of an E. coli cell, $N_{cells} = 2.410^{8} cells/mL * 0.18mL$ which is the number of cells in a $180\mu L$ culture with OD = 0.3 placed in a well containing $V_{medium} = 180\mu L$ of culture medium. Table 3.2 shows the species and their corresponding symbols.

The dynamic model has nine states equations plus one algebraic equation (M) and 26 parameters described in Table 3.3.

Variable	Description	Symbol	Units
x_1	$mRNA_{gA} -> mRNA_{luxR}$	mA	nM
x_2	$Aprotein -> LuxR$	A	nM
x_3	$Inducer -> AHL$	I	nM
M	$(A.I)monomer -> (LuxR.AHL)$	(A.I)	nM
x_4	$(A.I)_2 dimer -> (LuxR.AHL)_2$	$(A.I)_2$	nM
x_5	$mRNA_{gB} -> mRNA_{cI}$	mB	nM
x_6	$BProtein -> cI$	B	nM
x_7	$mRNA_{gC} -> mRNA_{gfp}$	mC	nM
x_8	$Cprotein -> GFP$	C	nM
x_9	$ExtracellularInducer -> AHL_{ext}$	I_e	nM

Table 3.2: List of variables used in the model.

3.4 Metabolic Burden Model

*O*nce we have the temporal dynamic model of the I1-FFL circuit, we combined it with the metabolic burden model from [16] . This is a mechanistic model of the cell that combines nutrient import and its conversion to cellular energy with the processes of transcription and translation. The metabolic model includes 14 intracellular variables: internal nutrient s_i, energy a, such as ATP; and four types of proteins along with their corresponding free and ribosome-bound mRNAs. The four types of proteins considered are (1) ribosomes r, (2) a transporter enzyme e_t and (3) a metabolic enzyme e_m, and (4) a class of house-keeping proteins q. The corresponding free mRNAs is denoted by m_x and ribosome-bound mRNA by c_x with $x \in r, t, m, q$. All the concentrations are measured in nM. The metabolic model goes as follows:

Fixed Parameter	Description	Value	Units
d_{mA}, d_{mB}, d_{mc}	m_A, m_B, m_C degradation rate	0.3624	min^{-1}
k_{pA}	m_A translation rate	80	min^{-1}
d_A	A degradation rate	0.035	min^{-1}
k_d	inducer diffusion rate	0.06	min^{-1}
k_2, k_3	$(A.I), (A.I)_2$ association rate	0.1	min^{-1}
k_{-2}	$(A.I)$ dissociation rate	20	min^{-1}
k_{-3}	$(A.I)_2$ dissociation rate	1	min^{-1}
γ_2	gC promoter coefficient	0.02	nM
β_1, β_2	gC promoter basal expression coefficient	0.05	$adim, nM^{-1}$
d_I, d_{Ie}	inducer degradation rate	0.0164	min^{-1}
d_{AI}, d_{AI2}	$(A.I), (A.I)_2$ degradation rate	0.035	min^{-1}
$k_{mA}C_{gA}$	Promoter strength and Plasmid origin of replication	30	min^{-1}
$k_{mB}C_{gB}$	Promoter strength and Plasmid origin of replication	10	min^{-1}
$k_{mC}C_{gC}$	Promoter strength and Plasmid origin of replication	30	min^{-1}
k_{pB}, k_{pC}	m_B, m_C translation rate	40,15	min^{-1}
d_B, d_C	B, C degradation rate	0.1733,0.1733	min^{-1}
γ_1	gB promoter Hill constant	50	nM
γ_3	gC promoter coefficient	0.1	$adim$
γ_4	gC promoter coefficient	1.42	$adim$
γ_5	gC promoter coefficient	70	nM^{-1}

Table 3.3: Parameters of the dynamic model

$$
\begin{aligned}
\dot{s_i} &= v_{imp}(e_t, s) - v_{cat}(e_m, s_i) - \lambda s_i \\
\dot{a} &= n_s v_{cat}(e_m, s_i) - \sum_{x \in r,t,m,q} n_x v_x(c_x, a) - \lambda a \\
\dot{r} &= v_r(c_r, a) - \lambda r + \sum_{x \in r,t,m,q} (v_x(c_x, a) - k_b r m_x + k_u c_x) \\
\dot{e_t} &= v_t(c_t, a) - \lambda e_t \\
\dot{e_m} &= v_m(c_m, a) - \lambda e_m \\
\dot{q} &= v_q(c_q, a) - \lambda q \\
\dot{m_x} &= w_x(a) - (\lambda + d_m) m_x + v_x(c_x, a) - k_b r m_x + k_u c_x \\
\dot{c_x} &= -\lambda c_x + k_b r m_x - k_u c_x - v_x(c_x, a), \qquad x \in r, t, m, q
\end{aligned}
\tag{3.2}
$$

3.5 Combined adaptive - metabolic burden model

We integrated the adaptive model together with the metabolic burden model of the I1-FFL synthetic network. This model includes additional species: three ribosomal complexes (cA, cB and cC) associated with the translation of the proteins A, B and C

3.5.1 Ribosomes binding to mRNAx

$$r + m_A \overset{k_b,k_u}{\longleftrightarrow} C_A$$
$$r + m_B \overset{k_b,k_u}{\longleftrightarrow} C_B$$
$$r + m_C \overset{k_b,k_u}{\longleftrightarrow} C_C$$

Where, k_b, k_u are the binding/unbinding rates $[nM \cdot min^{-1}]$

3.5.2 Ribosomes and ribosome-bound mRNAx complex dilution

$$r \xrightarrow{\lambda} \emptyset$$
$$C_A \xrightarrow{\lambda} \emptyset$$
$$C_B \xrightarrow{\lambda} \emptyset$$
$$C_C \xrightarrow{\lambda} \emptyset$$

Where, λ is the growth rate.

3.5.3 Translation of proteins

$$n_A a + C_A \xrightarrow{v_A} r + m_A + A$$
$$n_B a + C_B \xrightarrow{v_B} r + m_B + B$$
$$n_C a + C_C \xrightarrow{v_C} r + m_C + C$$

Where, a is the energy and n_x is the length (in number of aminoacids) of the protein x.

3.5.4 Ribosomes production

$$\begin{aligned}
&\xrightarrow{w_r} m_r \\
&r + m_r \overset{k_b,k_u}{\longleftrightarrow} C_r \\
&n_r a + C_r \xrightarrow{v_r} r + m_r + r \\
&C_r \xrightarrow{\lambda} \emptyset \\
&m_r \xrightarrow{\lambda+d} \emptyset
\end{aligned}$$

Where, w_r is the transcriptional rate of ribosomes and v_r is the translation rate of ribosomes. Theses species are involved in the reactions and lead to differential equations:

$$\begin{aligned}
&\dot{g_i} = v_{gi}(c_{gi}, a) - (\lambda + d_g)g_i \\
&\dot{m_{gi}} = w_{gi}(g_{i-1}, a) - (\lambda + d_{m,g})m_{gi} + v_{gi}(c_{gi}, a) - k_b r m_{gi} + k_u c_{gi} \\
&\dot{c_{gi}} = -\lambda c_{gi} + k_b r m_{gi} - k_u c_{gi} - v_{gi}(c_{gi}, a) \qquad\qquad i = 1, 2, 3
\end{aligned}$$

The transcription rates are defined by

$$w_{gi}(g_j, a) = w_g \frac{a}{\theta_{nr} + a}$$

The translation rates are defined by

$$v_{gi}(C_{gi}, a) = C_{gi} \frac{\gamma(a)}{n_g}$$

Include the additional consumption of energy and free ribosomes are defined by

$$\begin{aligned}
\dot{a} &= n_s v_{cat}(e_m, s_i) - \lambda a - \sum_{x \in r,t,m,q} n_x v_x(c_x, a) - \sum_{i=1,2,3} n_g v_{gi}(c_{gi}, a) \\
\dot{r} &= v_r(c_r, a) - \lambda r + \sum_{x \in r,t,m,q} (v_x(c_x, a) - k_b r m_x + k_u c_x) + \sum_{i=1,2,3} (v_{gi}(c_{gi}, a) - k_b r m_{gi} + k_u c_{gi})
\end{aligned}$$

Where, n_s is the nutritional efficiency, n_x is the length (in number of amino-acids) of the protein x , C_x is the ribosome-bound mRNA of species x and v_x is the translation rate of protein x.

Notice v_r twice because translation of ribosomes is made by ribosomes.

3.5.5 Growth rate at steady state

$$\dot{N} = \lambda N - d_N N$$

If exponential growth, each cell is at steady state, in the sense that its mass dynamics is:

$$\dot{M} = \lambda(a) R_t - \lambda M$$

Where $\lambda(a)R(t)$ decreases due dilution caused by cell population growth and λM increases proportionally to total number of translating ribosomes with proportionality equal to translational efficiency.

If cell mass does not change:

$$\dot{M} = 0 \qquad\qquad \lambda = \lambda(a)\frac{R(t)}{M}$$

$\frac{R(t)}{M}$ fraction de ribosomes. Compare with that of Scott [14]:

$$R_t = \sum_{x \in (r,t,m,q,A,B,C)} c_x$$

Translating ribosomes(number, no mass)

$$M = n_r \sum_{x \in (r,t,m,q)} c_x + \sum_{i=1,2,3} c_{gi}$$

$M \approx 10^8 aa$

Finally, the growth rate is defined by

$$\lambda = \frac{\gamma(a)}{M}\left(\sum_{x \in (r,t,m,q)} c_x + \sum_{i=1,2,3,q} c_{gi}\right)$$

The Combined adaptive - metabolic burden model is represented by the following equations:

Nutrient import and metabolism

Internalized Nutrient

$$\dot{s_i} = v_{imp}(e_t, s) - v_{cat}(e_m, s_i) - \lambda s_i \tag{3.3}$$

External Substrate

$$\dot{s} = v_{imp}(e_t, s)N = e_t \frac{v_t s}{k_t + s} N \tag{3.4}$$

Where N is the number of cells.

Number of Cells

$$\dot{N} = \lambda N - d_N N \tag{3.5}$$

Where d_N is the death rate

Import and metabolism rates

$$v_{imp} = e_t \frac{v_t s}{k_t + s}$$
$$v_{cat} = e_m \frac{v_m s_i}{k_m + s_i}$$

Translation of import and metabolism enzymes

$$\dot{e_t} = v_t(c_t, a) - \lambda e_t \tag{3.6}$$
$$\dot{e_m} = v_m(c_m, a) - \lambda e_m \tag{3.7}$$

Translation rates of import and metabolism

$$v_t = c_t \frac{\gamma(a)}{n_t}$$
$$v_m = c_m \frac{\gamma(a)}{n_m}$$

Translational efficiency

$$\gamma(a) = \frac{\gamma_{max} a}{K_\gamma + a}$$

Where, γ_{max} is the maximal translation rate.

Transcription of enzymes

Free mRNA

$$\dot{m_t} = w_t(a) - (\lambda + d_m)m_t + v_t - k_b r m_t + k_u c_t \tag{3.8}$$
$$\dot{m_m} = w_m(a) - (\lambda + d_m)m_m + v_m - k_b r m_m + k_u c_m \tag{3.9}$$

Where, $w_t = w_m = w_e$ assuming enzymes are co-expressed. Maximal transcription rate and d_m is the mRNA degradation rate.

Transcription of ribosomes

Free mRNA

$$\dot{m_r} = w_r(a) - (\lambda + d_m)m_r + v_m - k_b r m_r + k_u c_r \tag{3.10}$$

Ribosome-bound mRNA

$$\dot{c_r} = -\lambda c_r + k_b r m_r - k_u c_r - v_r \tag{3.11}$$

Transcription of non-growth related protein

Free mRNA

$$\dot{m_q} = w_q(a)R(q) - (\lambda + d_m)m_q + v_q - k_b r m_q + k_u c_q \tag{3.12}$$

Ribosome-bound mRNA

$$\begin{aligned}
\dot{c_q} &= -\lambda c_q + k_b r m_q - k_u c_q - v_q \\
v_q &= c_q \frac{\gamma(a)}{n_q} \\
R_q &= \frac{1}{1 + (\frac{q}{K_q})^q}
\end{aligned}$$

Translation of non-growth related protein

$$\dot{q} = v_q - \lambda q \tag{3.13}$$

Transcription of Genes A,B,C

Free mRNA

$$\dot{m_A} = w_A(a) - (\lambda + d_{mA})m_A + v_A - k_b r m_A + k_u c_A \tag{3.14}$$

$$\dot{m_B} = w_B(a) - (\lambda + d_{mB})m_B + v_B - k_b r m_B + k_u c_B \tag{3.15}$$

$$\dot{m_C} = w_C(a) - (\lambda + d_{mC})m_C + v_C - k_b r m_C + k_u c_C \tag{3.16}$$

Ribosome-bound mRNA

$$\dot{c_A} = -\lambda c_A + k_b r m_A - k_u c_A - v_A \tag{3.17}$$

$$\dot{c_B} = -\lambda c_B + k_b r m_B - k_u c_B - v_B \tag{3.18}$$

$$\dot{c_C} = -\lambda c_C + k_b r m_C - k_u c_C - v_C \tag{3.19}$$

Translation of Proteins

$$A = v_A - (\lambda + d_{mA})A$$
$$B = v_B - (\lambda + d_{mB})B$$
$$C = v_C - (\lambda + d_{mC})C$$

Transcription rates

$$w_r(a) = w_r \frac{a}{\theta_r + a}$$
$$w_t(a) = w_t \frac{a}{\theta_t + a} = w_m(a)$$

$$w_A(a) = K_{mA} C_{gA} \frac{a}{\theta_{nA} + a}$$
$$w_B(a) = K_{mB} C_{gB} \frac{a}{\theta_{nB} + a}$$
$$w_C(a) = K_{mC} C_{gC} \frac{a}{\theta_{nC} + a}$$

Notice $K_{mA}\frac{a}{\theta_{nA}+a} \equiv K_{mA}$ and analogous for B and C.

Translational Rates

$$v_A = c_A \frac{\gamma(a)}{n_A}$$
$$v_B = c_B \frac{\gamma(a)}{n_B}$$
$$v_C = c_C \frac{\gamma(a)}{n_C}$$
$$v_r = c_r \frac{\gamma(a)}{n_r}$$

Notice $\frac{\gamma(a)}{n_A} \equiv K_{pA}$ and analogous for B and C.

Finite Ribosomes Translation of ribosomes (free ribosomes dynamics)

$$\dot{r} = 2v_r - \lambda r + v_t + v_m + v_q + v_A + v_B + v_C - k_b r(m_t + m_m + m_q + m_A + m_B + m_C) + k_u(c_m + c_t + c_q + c_A + c_B + c_C) \tag{3.20}$$

Finite Energy- Available energy(or aa-equivalent) per cell

$$\dot{a} = n_s v_{cat} - \lambda a - n_r v_r - n_t v_t - n_m v_m - n_q v_q - n_A v_A - n_B v_B - n_C v_C \tag{3.21}$$

Finite Protein

$$M = -n_r r + v_m e_m + n_t e_t + n_m v_m + n_q v_q + n_A c_A + n_B c_B + n_C c_C + n_r(c_r + c_t + c_m + c_q + c_A + c_B + c_C)$$

Translating Ribosomes

$$R_t = \sum_{x \in (r,t,m,q,A,B,C)} c_x \qquad\qquad R_t = (c_r + c_t + c_m + c_q + c_A + c_B + c_C)$$

Growth Rate

$$\lambda = \frac{\gamma(a)}{M} R_t$$
$$\lambda = \frac{\gamma(a)}{M}(c_r + c_t + c_m + c_q + c_A + c_B + c_C)$$

In summary, we have the new combined model (3.22) that includes the metabolic burden effect in the protein production.

$$\begin{aligned}
\dot{x_1} &= w_A - (\lambda + d_{mA})x_1 + v_A - k_b r m_A + k_u c_A \\
\dot{x_2} &= v_A - (\lambda + d_A)x_2 - k_2 x_2 x_3 + k_{-2} M \\
\dot{x_3} &= -k_2 x_2 x_3 + k_{-2} M + K_d(x_9 - x_3) - d_I x_3 - \lambda x_3 \\
\dot{x_4} &= k_3 M^2 - k_{-3} x_4 - d_{AI2} x_4 - \lambda x_4 \\
\dot{x_5} &= w_B \frac{x_4}{\gamma_1 + x_4} - (\lambda + d_{mB})x_5 + v_B - k_b r m_B + k_u c_B \\
\dot{x_6} &= v_B - d_B x_6 \\
\dot{x_7} &= w_C \frac{x_4 + \beta_1 \gamma_4 x_6 + \beta_2 \gamma_5 x_4 x_6}{\gamma_2 + \gamma_3 x_4 \gamma_4 x_6 + \gamma_5 x_4 x_6} - (\lambda + d_{mC})x_7 + v_C - k_b r m_C + k_u c_C \\
\dot{x_8} &= v_C - (\lambda + d_C)x_8 \\
\dot{x_9} &= k_{cells} k_d(-x_9 + x_3) - d_{Ie} x_9 - \lambda x_9 \\
M &= -\frac{d_{AI} + k_{-2}}{4k_3} + \frac{1}{4k_3}\sqrt{(d_{AI} + k - 2)^2 + 8k_3(k_2 x_2 x_3 + 2k_{-3} x_4)}
\end{aligned} \tag{3.22}$$

3.6 Software implementation

All the previous models were implemented in Matlab. In the Appendix A it is possible to see all the resulting code.

4

Parameters tuning of the combined model: synthetic gene circuit including metabolic burden

4.1 Introduction

After integrating the metabolic burden model together with the adaptive model it is possible to use them for simulation in an integrated way. However, due to the high number of parameters involved in both models and the sloppiness of the metabolic burden model, the previous behavior of the synthetic gene network is corrupted and even lost. In order to deal with this problem, a parameter tuning of the combined model has to be performed. This parameter tuning involves searching for a compromise solution between the different requirements and can be stated as a multi-objective optimization problem. A common approach to address this kind of problem is to build a function that manages and agglutinates in a single index the objectives of interest. A linear vector of weights is used, however, the solution obtained depends on a correct choice of them and may not adequately reflect the preferences of the designer in relation to the desired performance balance.

4.2 Multi-objective optimization

*M*ulti-objective optimization (MOO), is an alternative to face a multi-objective problem [6]. For the In MOO all design objectives are important to the designer and all of them are optimized simultaneously. Thus, the solution rarely is unique, instead we obtain a set of solutions called the Pareto Front. In this sense, all solutions are Pareto-optimal and differ from each other in the objectives trade-off. In order to successfully implement the multi-objective optimization

approach, at least three fundamental steps are required [13]:

1. the multi-objective problem (MOP) definition: defining the circuit behavioral specifications in a proper way.
2. the optimization process: tuning the parameters using multi-objective global optimization (MOO).
3. and the multi-criteria decision making (MCDM) stage: obtaining tuning guidelines useful for the wet-lab implementation.

This overall multi-objective optimization design (MOOD) enables to analyze design objectives trade-offs to implementa preferable solution [13]. Furthermore, it may provide a better understanding of the problem at hand by the so called process of *innovization* through optimization as stated by [7]. Next we describe each of the steps in detail.

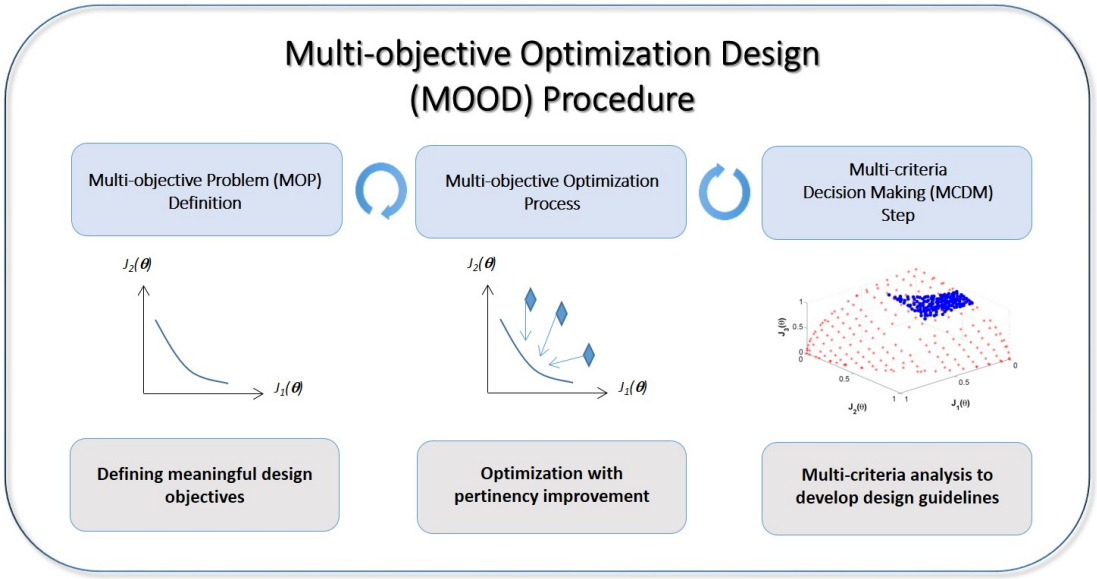

Figure 4.1: Steps for the multi-objective optimization design procedure [13].

Defining the circuit behavioral specifications

The starting point of the proposed methodology is the multi-objective problem definition, this is the specification of the desired dynamical behavior for the circuit to be designed. This can be done in several ways. From the designer's point of view, specifying the circuit behavior in terms of the desired output signal profile for a given input signal profile is a natural approach [4]. The input signal is chosen as the one that is going to be used in working conditions, or as simple standard

probing input-signals (e.g., step-like, sinusoidal, or pulse ones). Once the desired input-output relationship is defined, the set of circuit parameters can be obtained by optimization-based system identification [5]. This approach is useful for linear dynamical systems, as their time-response to these probing signals fully characterizes the circuit dynamical behavior. This is not the case for nonlinear circuits as the ones typically encountered in Synthetic biology. Thus, the particular signal to be used in working conditions should be chosen. Yet, this may be very restrictive. Indeed, usually the input signal to a circuit will have varying characteristics. In the best case, it will belong to a given class (e.g., step-like signal with varying amplitude). Therefore, the dynamical behavior, i.e., the desired circuit time-response to a given input signal, is better given as a set of input-output performance indexes to be optimized.

Specifying the desired circuit behavior in terms of performance indexes to be optimized has many advantages. In the general case, the indexes will take the form of functionals mapping the circuit trajectories to the reals. Thus, for a circuit with dynamics given by the model:

$$\begin{aligned} \dot{x} &= f(x, \theta) \\ 0 &= g(x, \theta) \end{aligned} \tag{4.1}$$

where $x \in \mathbb{R}^n$ is the state, $\theta \in \mathbb{R}^p$ the parameters, and function $g(.)$ represents algebraic constraints in the system. The indexes can be expressed as:

$$J_i(\theta) = \int_{t_0}^{t_f} h(x(\tau, \theta), \tau) d\tau \tag{4.2}$$

for some functions possibly time-dependent function $h(.)$ of the system trajectories during a time interval of interest $[t_0, t_f]$, being $i = 1 \dots n_i$ is the number of indexes. These can be made valid for a whole class of input signals, may consider other signals in the circuit besides the input and output ones, robustness with respect to uncertainty in the circuit parameters can be included, etc. They will typically consider the desired performance at steady state (*precision*), and some measures of the quality of the transient. proper definition of the optimization indexes representing the desired behavior is a key point. An incorrectly specified objective, not properly representing the actual desired behavior, will lead the optimization in a wrong direction, returning a parameter set that will give misleading design guidelines. Moreover, for the proper interpretation of results by the designer, one must propose meaningful design objectives.

Multi-objective parameters tuning

As mentioned above, representing the desired behavior will eventually lead to several objectives to be optimized. That is, the optimization problem will be a multi-objective one in the general case. Typically, some of the objectives will be in conflict, so a trade-off among solutions is required. *Ad-hoc* weighting of the different objectives may be used to transform the problem into a single-objective one [8]. Alternatively, thresholds on each of the objectives may be set in order

to run multiple times a single-objective optimization. Instead, we address the problem as a truly multi-objective optimization design (MOOD) one.

The multi-objective optimization (MOO) process seeks to approximate the best parameters θ_P^* that give the best Pareto-front approximation J_P^* [9]. 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**, followed by a filtering of the solutions in order to obtain the θ_P^* that defines the Pareto front approximation J_P^*. This could be a good option for problems with few decision variables. For problems with a large number of decision variables, as our case, it is more efficient to use an appropriate multi-objective optimization algorithm to approximate this solution.

We obtain the Pareto-optimal front of solutions via spMODE, a multi-objective optimization algorithm based on differential evolution [10; 12] implemented in Matlab, and it is available at Matlab Central [1]. The algorithm spMODE actively searches for all the solutions in the parameter space along the Pareto front. It:

- improves convergence by using an external file to store solutions and include them in the evolutionary process;
- improves spreading by using the spherical pruning mechanism [10];
- improves pertinency of solutions, i.e., getting interesting solutions from the designer's point of view, by means of a basic bound mechanism in the objective space, as described in [11].

[1] http://es.mathworks.com/matlabcentral/fileexchange/39215

5

Simulations and results

For tuning the parameters of the new model to achieve adaptation behavior, we applied the three steps of MOOD.

5.1 Multi-objective problem (MOP) definition

The first step of the MOOD framework is to formulate the circuit specifications as design objectives to be optimized. Our problem consists on finding the set of values for the **seven decision variables** θ (Table 5.1) corresponding to the parameters of the model (3.22), to **minimize four objectives :**

$$\min_{\theta \in \Re^7} J(\theta) = \quad [J_1(\theta), J_2(\theta), J_3(\theta), J_4(\theta)] \in \Re^4$$

with:

$$\begin{aligned} J_1(\theta) &= \frac{2\left(x_9(t_f) - x_9(t_0)\right)}{\int_{t_0}^{t_f} |\frac{dx_8}{dt}| dt} \\ J_2(\theta) &= \frac{x_8(t_f) - x_8(t_0)}{x_9(t_f) - x_9(t_0)} \\ J_3(\theta) &= \int_{t_0}^{t_f} t |\frac{dx_8}{dt}| dt \\ J_4(\theta) &= \sum_{k=1}^{N} \frac{|x_{exp}(k) - x_{sim}(k)|^2}{N} \end{aligned} \tag{5.1}$$

where t_f is the time length of the experiment, and the input stimulus is applied at t_0. Sensitivity is the inverse of $J_1(\theta)$. Notice that the total absolute variation of the C protein concentration is obtained as half of the accumulated absolute value of the time derivative of x_8. The lower $J_1(\theta)$

Parameter	Description
n_s	nutrient efficiency
v_t	max.nutrient import rate
K_t	nutrient import threshold
V_m	max. enzymatic rate
K_m	enzymatic threshold
γ_{max}	max. translation elongation rate
K_γ	max. translation elongation threshold

Table 5.1: Parameters of the model selected as decision variables for optimization.

(larger output peak w.r.t. input variation), the higher the sensitivity. Precision is the inverse of $J_2(\theta)$, i.e. the inverse of the ratio between the variation of the C protein concentration between t_0 and t_f, and the variation of the external inducer concentration between t_0 and t_f. If the C protein concentration x_8 at time t_f is the same as the initial one at time t_0, precision is infinite. $J_3(\theta)$ is the integral of the time weighted absolute error value (ITAE), it represents the total variation of x_8 (concentration of protein C) weighted in time, this minimizes the transient time of x_8. $J_4(\theta)$ is the squared mean error (SME) between the experimental cell growth N data and the results obtained with the model. The objectives are defined as their own inverses expression according to the optimization standards problems.

This objectives combine two types of objectives. On the one hand, we have design objectives like $J_1(\theta)$, $J_2(\theta)$ and $J_3(\theta)$ that are used to find the appropriate set of parameters giving adaptation behaviour. On the other hand, we have identification objectives, like $J_4(\theta)$, that help us to incorporate experimental data into the parameter search.

5.2 Multi-objective optimization

The second step is carried out by the dynamic optimization. The problem of MOO seeks to determine the best possible approximation Θ_P^* of the Pareto set Θ_P that generates the best Pareto approximation J_P^*. This search can be done by means of a random sampling (Monte-Carlo type) of the space of decision variables Θ, that is, the parameters of our model to then filter the solutions and obtain the ones Θ_P^* that generate the front J_P^*. For small problems this may be a good option, but for problems like ours $(mxn) = (7x4)$ it is more efficient to use MOO algorithms to find a better approach to that solution. In our case, we have used the sp-MODE algorithm [10], which is an evolutionary algorithm of multi-objective optimization (MOEA). With this, a better approximation to the Pareto front was obtained. The selection of a compromise solution according to the preferences of the designer takes place in an a-posteriori analysis on the Pareto

Parameter	Value
n_s	44.93
v_t	91.91
K_t	192
V_m	1000
K_m	354.8
γ_{max}	1084
K_γ	1.09e8

Table 5.2: Values for the decision variables with the objectives J_1 and J_2

front J_P^*. In this work, the Level Diagram tool (LD) [1] is used to visualize the calculated front. An LD is an alternative for the visualization and analysis of the set and m-dimensional front, which is not a trivial task when the number of objectives is greater than three or when the number of decision variables in the Pareto set is high, as in our case. The LDs are based on the classification of the Pareto set Θ_P^* calculated and making each objective $J_q(\theta)$ is normalized with respect to its minimum and maximum value. The LD generates two figures: the first one shows in X-axis the objective $J_(\theta)$ and the Y-axis represents the norm of this objective, the second plot shows the decision variables obtained from the optimization in the same way that the first one.

5.3 Multi-Criteria decision making optimization

The third step is to obtain guidelines and guidance for the implementation in the new model. We first make an optimization to adjust the parameters, so the new model has again the characteristic of adaptation, that is optimazing the seven decision variables using the objectives of sensitivity and precision. Both Fig.5.1 and Fig.5.2 show the LD where the decision variables and objectives are represented.

The results of the simulation are shown below in Fig. 5.3 using the solutions resulting from the optimization. In this case you can see that taking those solutions for the parameters the model gets the characteristic adaptation.

Then a new objective $J_3(\theta)$ was implemented allowing to minimize the transient time of protein C concentration. Both Fig.5.4 and Fig.5.5 show the LD where the decision variables and objectives are represented. The simulation of the model is shown in Fig.5.6, taking a set of requests obtained from the optimization as fixed parameters. We observed in the simulation Fig. 5.6 the model continues mantaining the adaptation and the transient time of the protein C concentration has also been minimized.

Finally, we implemented a fourth objective $J_4(\theta)$, the quadratic error between the experi-

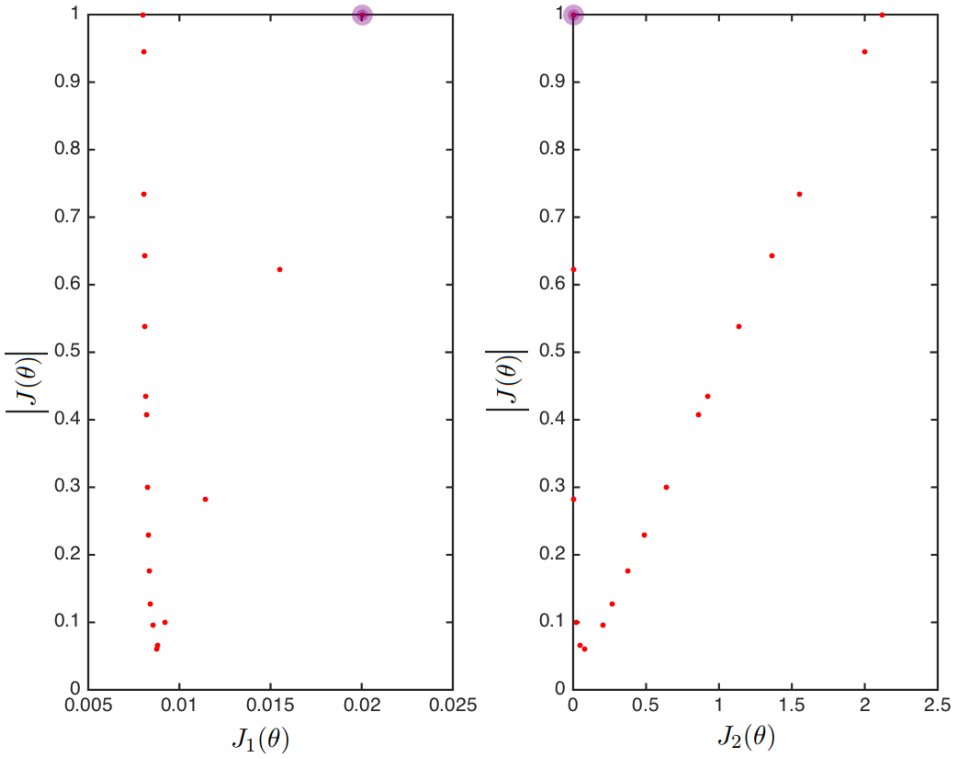

Figure 5.1: Value of the objectives J_1 (sensitivity) and J_2 (precision).The point in purple is the best solution according to our requirements.

Parameter	Value
n_s	0.661
v_t	1
K_t	495.6
V_m	1648
K_m	259.8
γ_{max}	670.4
K_γ	1.28e10

Table 5.3: Values of the decision variables for the selected solution (Three objectives case).

mental and simulated data of cell growth N. Fig.5.7 and Fig.5.8 show the LD where the decision variables and objectives are represented.

The simulation of the model is shown in Fig.5.9 and Fig.5.10, taking a set of requests obtained

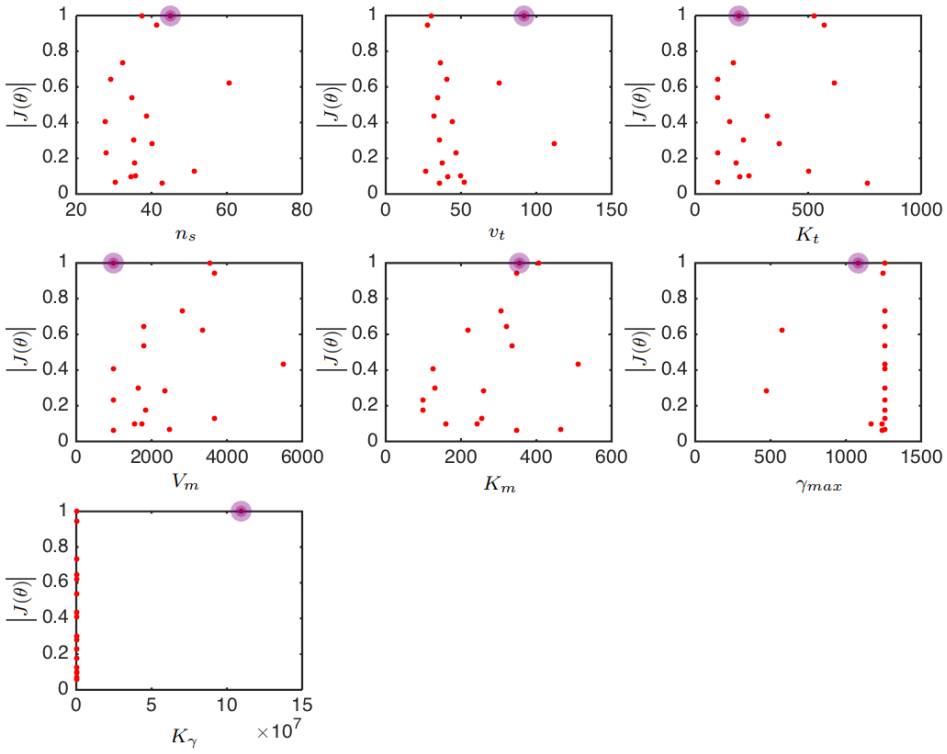

Figure 5.2: Values of the decision variables with the objectives J_1 (sensitivity) and J_2 (precision). The point in purple is the best solution according to our requirements.

Parameter	**Value**
n_s	85.9
v_t	25.8
K_t	501.2
V_m	1000
K_m	1000
γ_{max}	959
K_γ	3e8

Table 5.4: Values of the decision variables for the selected point (Four objectives case).

from the optimization as fixed parameters.

We observed in the simulation results, see Fig. 5.9, that the model continues maintaining the adaptation. Additionally, Fig.5.10 shows cell growth as we expected (second bottom plot

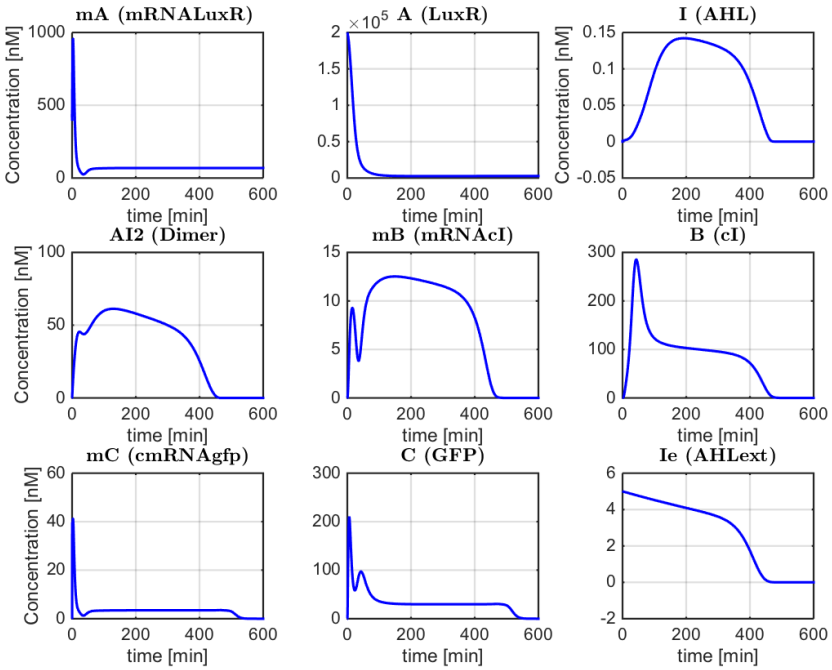

Figure 5.3: Temporal evolution of the biochemical species for the selected solution (Two objectives case).

OD(optical density)).

The model obtained together with the parameters identified with the four objectives gives results like the ones in Fig. 5.9. In Fig. 5.11 we present different experimental results obtained in the group SB2CL. As it is possible to see, **the model gives results that are qualitatively similar to the ones obtained in the simulations**. Thus, our model can capture the problem of metabolic burden when a protein is produced. Furthermore, it can be used to design the synthetic circuit with a good approximation of the behaviour that can be found in living microorganisms.

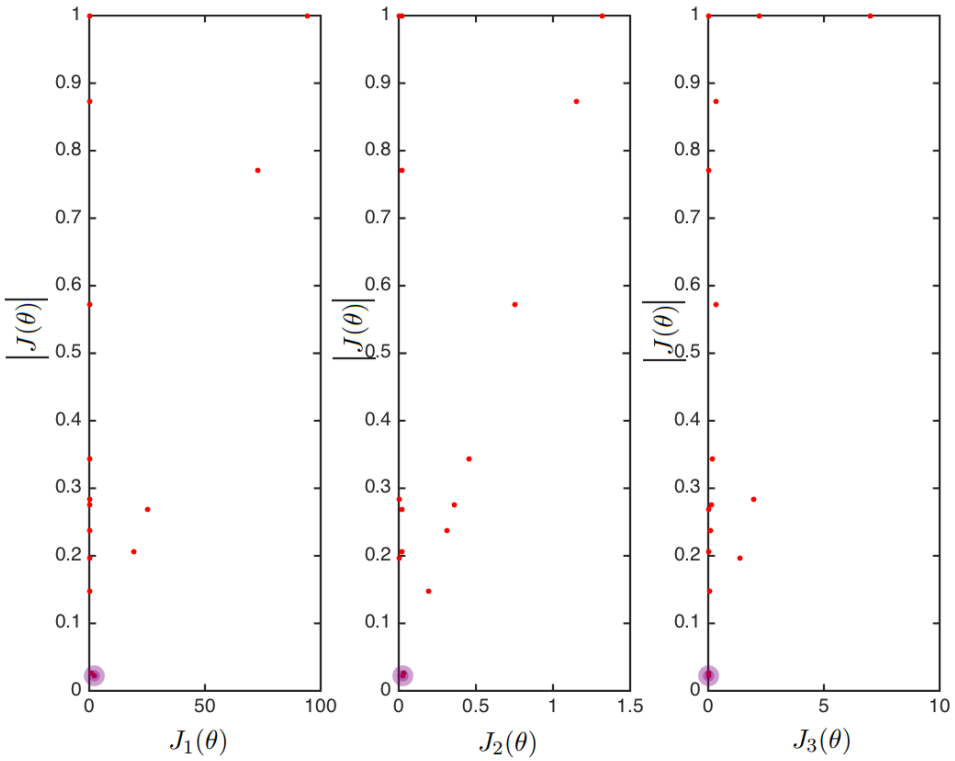

Figure 5.4: Values of the objectives J_1 (sensitivity), J_2 (precision) and J_3 (ITAE).

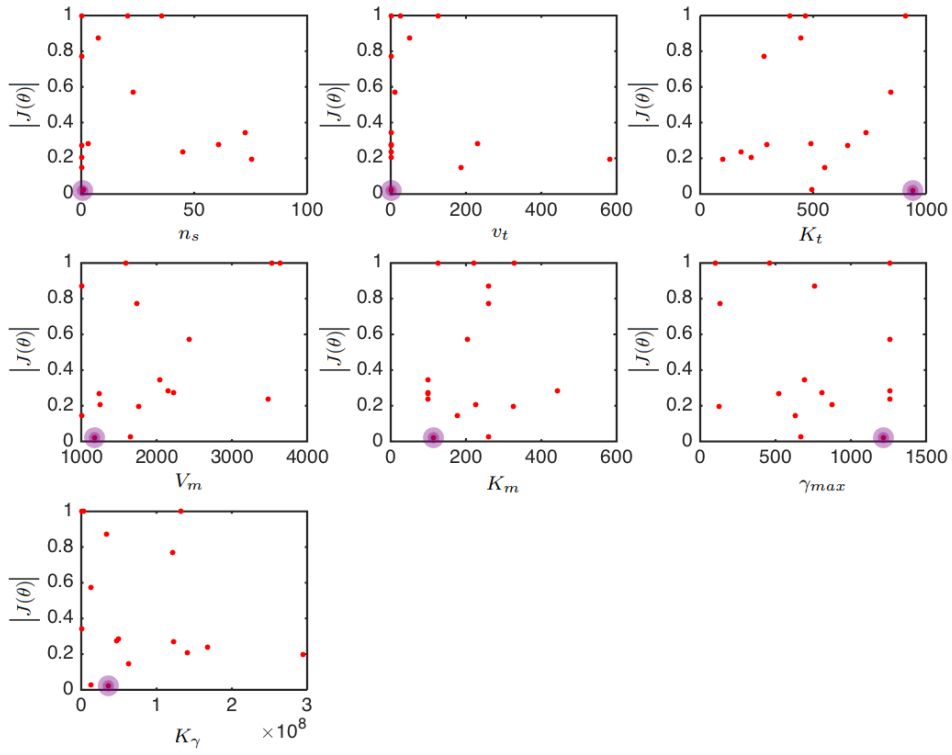

Figure 5.5: Values of the decision variables for objectives J_1 (sensitivity), J_2 (precision) and J_3 (ITAE).

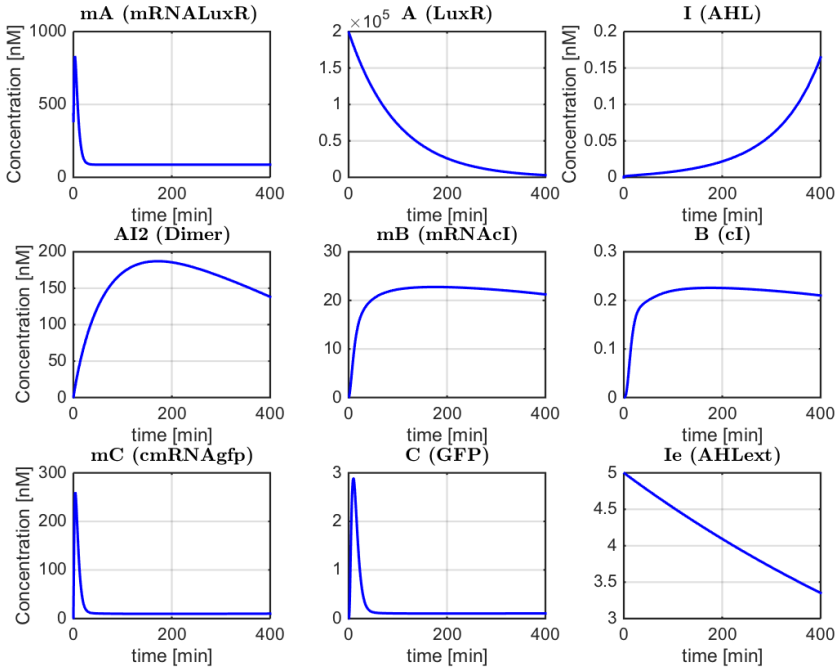

Figure 5.6: Temporal evolution of the biochemical species with the selected solution (Three objectives case).

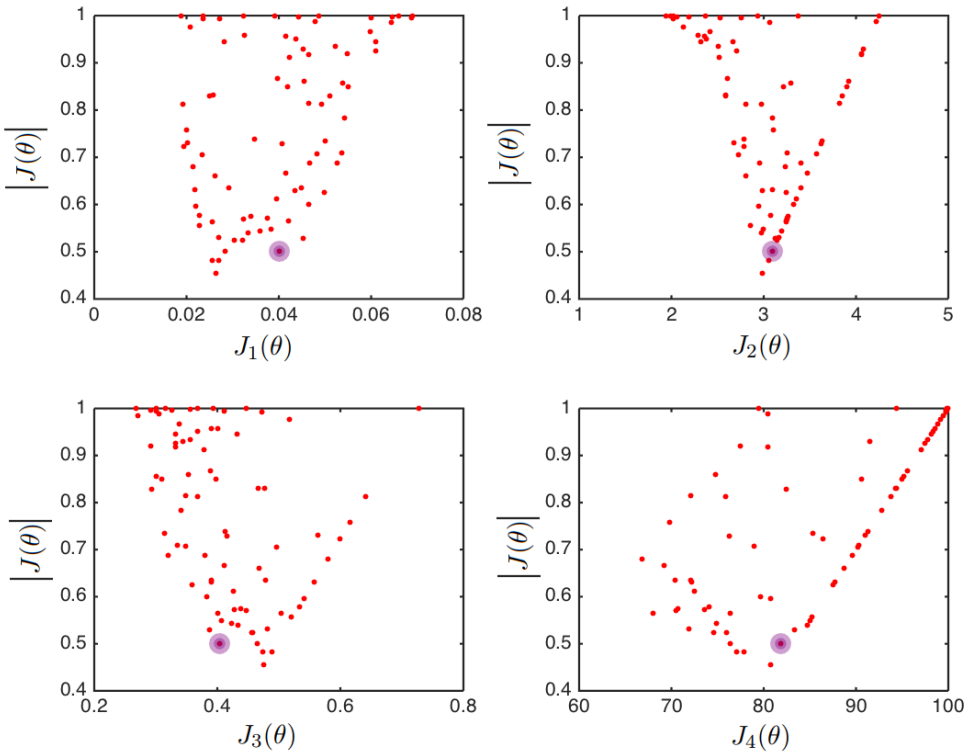

Figure 5.7: Values of the objectives J_1 (sensitivity), J_2 (precision), J_3 (ITAE) and J_4 (SME).

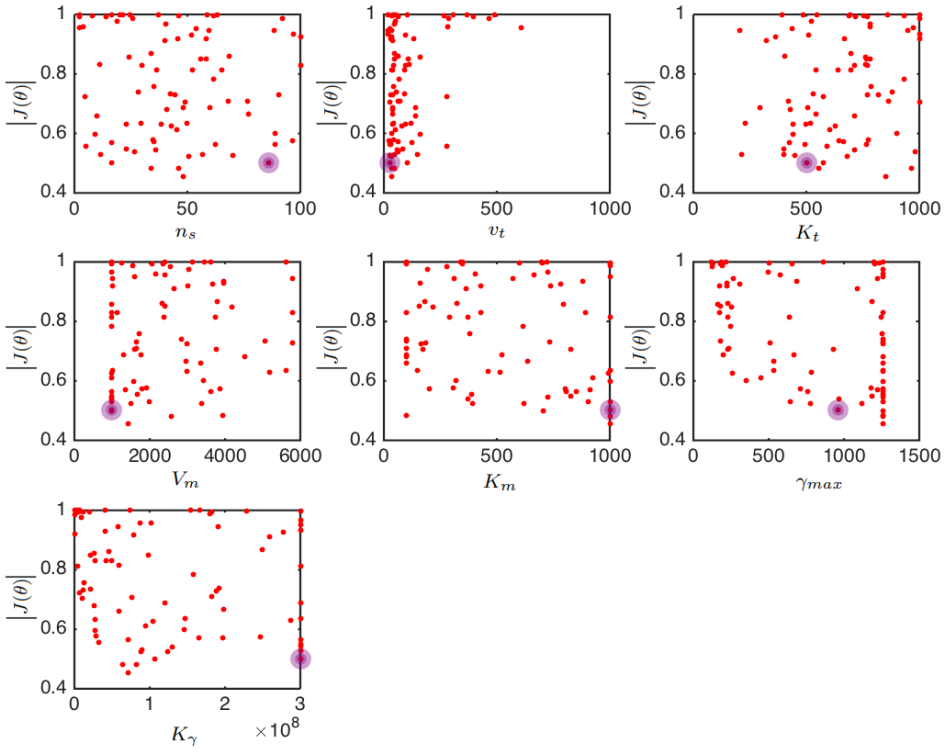

Figure 5.8: Values of the decision variables for the objectives J_1 (sensitivity), J_2 (precision), J_3 (ITAE) and J_4 (SME).

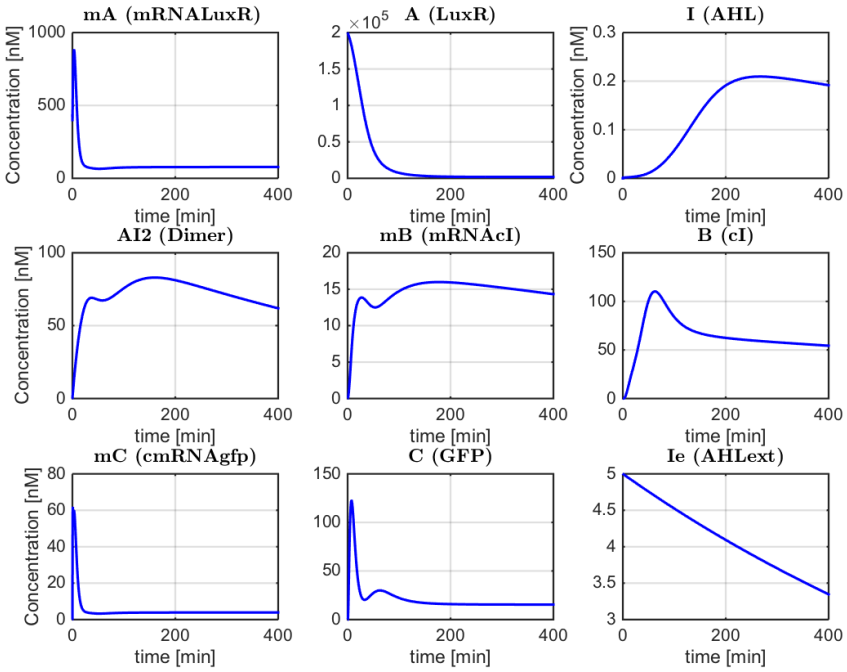

Figure 5.9: Temporal evolution of the biochemical species for the selected point (Four objectives case).

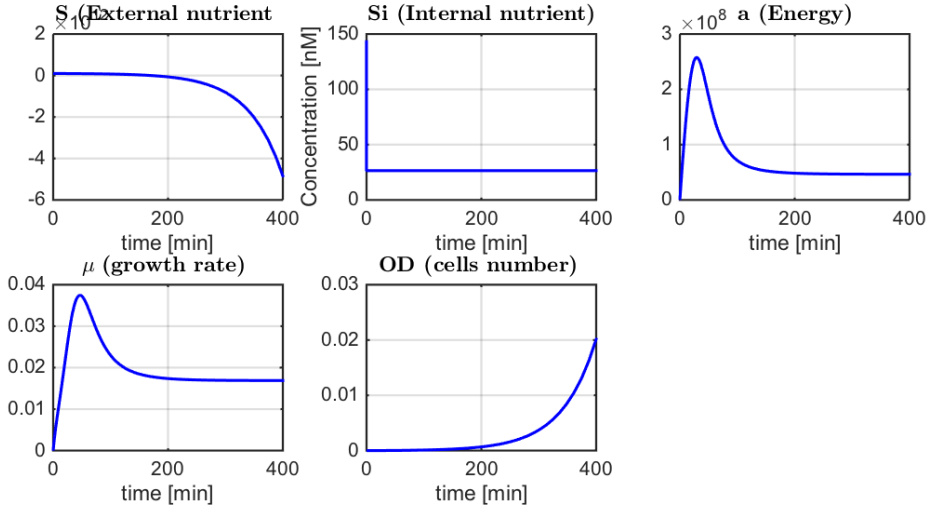

Figure 5.10: Temporal evolution of the growth related variables of metabolic model. In particular the second bottom plot is the expected cell growth

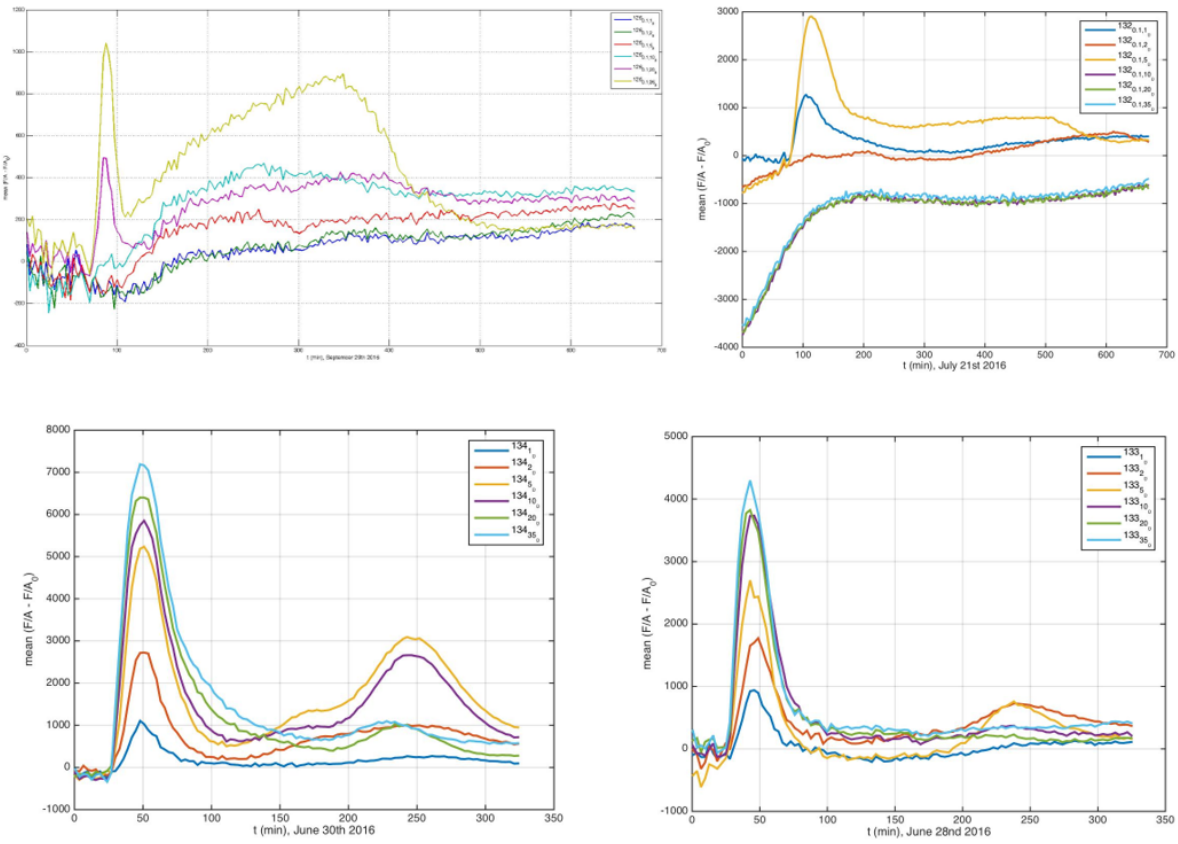

Figure 5.11: Experimental examples of the I1-FFL circuit implemented in the SB2CL Lab.

6

Conclusions and future work

A comprehensive review of several concepts has been done in Chapter 2, regarding the Central dogma of molecular biology and modeling of gene expression. The most common approach for modeling biological reactions relies on the law of mass action kinetics to derive a set of differential equations that characterize the evolution of reacting species concentrations (or number of molecules) over time.

Some obtained results are present as follows:

- The genetic synthetic circuit used in this work has relevance in biology and it has direct application in biotechnology and in Synthetic Biology design problems.
- We took a system of biochemical reactions modeled by Law of mass action kinetics for the synthetic gene network, and the corresponding set of deterministic ordinary differential equations (ODEs).
- We studied and used a metabolic model of the cell from a collaborator of the group SB2CL.
- We combined and integrated the model of the synthetic gene network with the metabolic model of the cell, in order to capture the metabolic burden effect in synthetic gene networks.
- The new combined model includes the proteins and mRNA of the I1-FFL circuit, the proteins and mRNA of the metabolic model of the cell, and the new ribosomal complexes of the I1-FFL circuit proteins.
- We assessed the necessity of tuning the combined model parameters, as the behaviour of the I1-FFL circuit was lost when we combined the models.
- We performed a multi-objective optimization and designed the proper objectives indices to tune the model parameters taking into account the desired behaviour (adaptation) and the experimental information.

- We obtained a model of the synthetic gene network that includes the effects of the metabolic burden.

The model that we obtained together with its tuned parameters is ready to be used in further applications such as model identification with experimental data and to analyze the effect of the over-expression of synthetic proteins in the growth rate of the microorganisms.

Acknowledgements

Research in this area is partially supported by program *Becas en el Exterior 2011* of Secretaría Nacional de Educación Superior, Ciencia, Tecnología e Innovación (SENESCYT) by the government of Ecuador.

Bibliography

[1] A new graphical visualization of n-dimensional pareto front for decision-making in multiobjective optimization. *Information Sciences*, 178(20):3908 – 3924, 2008.

[2] A. Ahmet and D. Arnosti. Mathematical modeling of gene expression: a guide for the perplexed biologist. *Nature Reviews Genetics*, 2011.

[3] U. Alon. *An Introduction To systems biology. Design Principles of Biological Circuits*. Chapman & Hall/ CRC Mathematical and computational Biology Series, London, UK, 2007. ISBN 978-1-58488-642-6.

[4] J. Ang, B. Ingalls, and D. McMillen. Probing the input-output behavior of biochemical and genetic systems: System identification methods from control theory. *Computer Methods*, (Part C):279–317, 2010.

[5] J. R. Banga. Optimization in computational systems biology. *BMC Systems Biology*, 2:47, 2008.

[6] Y. Boada, G. Reynoso-Meza, J. Picó, and A. Vignoni. Multi-objective optimization framework to obtain model-based guidelines for tuning biological synthetic devices: an adaptive network case. *BMC Systems Biology*, 10(1):27, Mar 2016. ISSN 1752-0509. doi: 10.1186/s12918-016-0269-0. URL `https://doi.org/10.1186/s12918-016-0269-0`.

[7] K. Deb, S. Bandaru, D. Greiner, A. Gaspar-Cunha, and C. C. Tutum. An integrated approach to automated innovization for discovering useful design principles: Case studies from engineering. *Applied Soft Computing*, 15(0):42 – 56, 2014.

[8] C. A. Mattson and A. Messac. Pareto frontier based concept selection under uncertainty, with visualization. *Optimization and Engineering*, 6(1):85–115, 2005.

[9] K. Miettinen. *Nonlinear Multiobjective Optimization*, volume 12. Kluwer Academic Publishers, Boston, MA, 1999.

[10] G. Reynoso-Meza, J. Sanchis, X. Blasco, and M. Martínez. Design of continuous controllers using a multiobjective differential evolution algorithm with spherical pruning. *Applications of Evolutionary Computation*, pages 532–541, 2010.

[11] G. Reynoso-Meza, J. Sanchis, X. Blasco, and J. M. Herrero. Multiobjective evolutionary algortihms for multivariable PI controller tuning. *Expert Systems with Applications*, 39:7895 – 7907, 2012.

[12] G. Reynoso-Meza, S. García-Nieto, J. Sanchis, and X. Blasco. Controller tuning using multiobjective optimization algorithms: a global tuning framework. *IEEE Transactions on Control Systems Technology*, 21(2):445–458, 2013.

[13] G. Reynoso-Meza, X. Blasco, J. Sanchis, and M. Martínez. Controller tuning using evolutionary multi-objective optimisation: current trends and applications. *Control Engineering Practice*, 28:58–73, 2014.

[14] M. Scott and B. Ingalls. Using the linear noise approximation to characterize molecular noise in reaction pathways. Department of Applied Mathematics, University of Waterloo,Waterloo-Ontario, Canada, 2013.

[15] G.-B. Stan. Modelling in biology. Lecture Notes in Modelling in Biology Course 2010-2012, Centre for Synthetic Biology and Innovation, Imperial College London, 2011.

[16] A. Y. Weiße, D. A. Oyarzún, V. Danos, and P. S. Swain. Mechanistic links between cellular trade-offs , gene expression , and growth. *Proccedings of the National Academy of Sciences*, page 21, 2015.

[17] J. P. Yadira Boada, Alejandro Vignoni. Reducción de modelos en biología sintética: Modelado y simulación de ruido y variabilidad en expresión genética. *XXXIV Jornadas de Automática 2013*, 2013.

Appendix A

Matlab Code

A.1 Mathematical Model

```
1
%% Reduced model adaptation  (LuxR, cI, GFP) + metabolic burden model
3
function [dxdt] = model_mburden_adap_v3(t,x,param)
5
%% parameters for adaptation model
7
    % dmA         % mA degradation rate  [1/min]
9   % dmB         % mB degradation rate  [1/min]
    % dmC         % mC degradation rate  [1/min]
11
    % kpA         % mA translation rate          [1/min]
13  % kpB         % mB translation rate          [1/min]
    % kpC         % mC translation rate          [1/min]
15
    % dA          % A degradation rate           [1/min]
17  % dB          % B degradation rate           [1/min]
    % dC          % C degradation rate           [1/min]
19
    % kd          %inducer diffudion rate       [1/min]
21
    % k2          % (AI) association rate        [1/min]
23  % k3          %(AI)2 association rate        [1/min]

25  % k_2         % (AI)2 dissociation rate      [1/min]
    % k_3         % (AI)2 dissociation rate      [1/min]
27
    % gamma1      % gB promoter Hill constant    [nM]
29  % gamma2      % gC promoter coefficient      [nM]
```

```
    % gamma3      % gC promoter coefficient        [adim]
    % gamma4      % gC promoter coefficient        [adim]
    % gamma5      % gC promoter coefficient        [n/M]

    % beta1       % gC promoter basal expression coefficient          [adim]
    % beta2       % gC promoter basal expression coefficient           [n/M]

    % dI          % Inducer(AHL) degradation rate                       [1/min]
    % dIe         % Inducer (AHL extracellular Inducer) degradation rate [1/min]

    % dAI         % (AI)(Monomer) degradation rate    [1/min]
    % dAI2        % (AI)2 (Dimer) degradation rate    [1/min]

    % KmACgA      % KmA transcription rate,CgA(GA copy numer)
                   %Promoter strength and plasmid origin of replication   [1/min]
    % KmBCgB      %KmB transcription rate,CgB(GB copy numer)
                   % Promoter strength and plasmid origin of replication  [1/min]
    % KmCCgC      %KmC transcription rate,CgA(GB copy numer)
                   % Promoter strength and plasmid origin of replication  [1/min]

    % mu=0.028;  %Specific growth rate  [1/min]
    %
    %Ncells=%2.4e8*0.18;      % [cells]   % number of cells in a 180 L culture with
    OD = 0.3 placed
    %Vcell=1e-15;             % [L]       %  typical volume of an E. coli
    %Vmedium=180;             % [L]       % (culture medium)


%% parameters for adaptation metabolic burden model

        %s               external nutrient
        %ns              nutrient efficiency
        %Kt              nutrient import half-maximal thresholds
        %Km              enzymatic half-maximal thresholds
        %Kq              q-autoinhibition half-maximal thresholds
        %vt_max          max.nutrient import rate
        %vm_max          max.enzymatic metabolizes rate
        %nr              ribosome length (in amino acids)
        %nq              length of q protein(in amino acids)
        %nt              length of et enzyme (in amino acids)
        %nm               length of em enzyme (in amino acids)
        %gamma_max        max. translation elongation rate
        %Kgamma           translation elongation threshold
        %ku               mRNA-ribosome unbinding rate
        %kb               mRNA-ribosome binding rate
        %dm               mRNA- degrdation rate
```

```
75          %wr_max           max. ribosome transcription rate
            %wt_max           max. enzyme that transport transcription rate
77          %wm_max           max. enzyme that metabolism transcription rate
            %wq_max           max. house-keeping protein transcription rate
79          %hq               house-keeping autoinhibition Hill coef.
            %Kq               house-keeping threshold
81          %teta_r           ribosome transcription threshold
            %teta_nr          non-ribosomal transcription threshold
83          %Mp               total cell mass
            %nA,nB,nC         length of LUXR,cI,C proteins(in amino acids)
85

87  %% adaptation model  states

89          % states

91          % x(1)-> mA      % mRNA_gA->mRNA_A->mRNA_LuxR     [nM]
            % x(2)-> A       % LuxR Protein                   [nM]
93          % x(3)-> I       % AHL  Inducer                   [nM]
            % x(4)-> AI2     % Dimer                          [nM]
95          % x(5)-> mB      % mRNA_gB->mRNA_B->mRNA_cI       [nM]
            % x(6)-> B       % cI   Protein                   [nM]
97          % x(7)-> mC      % mRNA_gC->mRNA_C->GFP           [nM]
            % x(8)-> C       % C    Protein                  [nM]
99          % x(9)-> Ie      % Extracellular Inducer AHL      [nM]

101 %% metabolic burden model  states

103
            % x(10)-> mr           mRNA-free ribosomes
105         % x(11)-> Cr           ribosome-bound mRNA free ribosomes
            % x(12)-> CA           ribosomal complex protein LuxR-A
107         % x(13)-> CB           ribosomal complex protein cI -B
            % x(14)-> CC           ribosomal complex protein GFP-C
109         % x(15)-> r            finite ribosomes
            % x(16)-> a            generic form of energy  ATP,NADPH
111         % x(17)-> et           enzyme that transport s into de cell
            % x(18)-> em           enzyme that metabolizes si into a
113         % x(19)-> mt           mRNA-enzyme that transport
            % x(20)-> mm           mRNA-enzyme that metabolizes
115         % x(21)-> Ct           ribosome-bound mRNA enzyme that transport
            % x(22)-> Cm           ribosome-bound mRNA enzyme that metabolizes
117         % x(23)-> si           nutrient internalized
            % x(24)-> mu           growth rate
119         % x(25)-> Mp           MAsa tottal cells
            % x(26)-> mq           mRNA-house-keeping protein
```

```
121         % x(27)-> Cq              ribosome-bound mRNA house-keeping proteins
            % x(28)-> q               house-keeping protein
123         % x(29)->s                external nutrient
             % x(30)->N                cells number
125
    %%%%%%
127
            % vA             translation rate protein LuxR-A
129         % vB             translation rate protein cI -B
            % vC             translation rate protein GFP-C
131         % vr             translation rate of ribosomes
            % Vimp           import rate
133         % Vcat           metabolism rate
            % q              house-keeping protein
135         % gamma_a        translation elongation rate
            % vt             translation rate of transport enzyme
137         % vm             translation rate of metamolism enzyme
            % vq             translation rate of q
139         % wr             transcription rate free ribosomes
            % wt             transcription rate enzyme that transport
141         % wm             transcription rate enzyme that metabolizes
            % wq             transcription rate house-keeping protein
143         % mq             mRNA-house-keeping protein
            % Cq             ribosome-bound mRNA house-keeping protein
145


147
     %% Synthetic gene circuit + metabolic burden
149
     % nutrient import and metabolism
151


153
              %gamma_a=(gamma_max*a)/(Kgamma+a)    %translational   %efficiency,
155                                                %rate of traslational elongation
               gamma_a=(param.gamma_max*x(16))/(param.Kgamma+x(16));
157
              %vt=Ct*gamma_a/nt          %translation rate of transport enzyme
159              vt=x(21)*gamma_a/param.nx;

161           %vm=Cm*gamma_a/nm         %translation rate of metabgolism enzyme
               vm=x(22)*gamma_a/param.nx;
163


165 % growth rate
```

```
167     %mu=(gamma_a/M)*Rt                  Rt=traslating ribosomes(numer,no masss)
        %mu=(gamma_a/M)*(Cr+Ct+Cm+Cq+CA+CB+CC);
169
        mu=((gamma_a)/param.Mp)*...
171             (x(11)+x(12)+x(13)+x(14)+x(27)+x(21)+x(22));
173
175      %translation of enzymes
177             %et= vt-mu*et;                          %transport enzyme
                    dxdt(17,1)=vt-mu*x(17);
179
                %em= vm-mu*em;                          %metabolism enzyme
181                 dxdt(18,1)= vm-mu*x(18);
183
                %Vimp=et*(vt_max*S0)/(Kt+S0)            %import rate
185                 Vimp= x(17)*(param.vt_max*x(29))/(param.Kt+x(29));
187             %Vcat=em*(vm_max*Si)/(Km+Si)        %metabolism rate
                    Vcat=x(18)*(param.vm_max*x(23))/(param.Km+x(23));
189
                %Si=Vimp-Vcat-mu*si                 %nutrient internalized
191
                dxdt(23,1)=Vimp-Vcat-mu*x(23);
193
195             %S=kin-Vimp.param.Ncells-dsS                    %nutrient external
197             dxdt(29,1)=-Vimp*x(30);
                    if (x(29)<=0 && dxdt(29,1)<0), dxdt(29,1)=0; end
199
    %translation rates
201
            %vr=Cr*gamma_a/nr
203           vr=x(11)*gamma_a/param.nr;
205           %vA= CA*gamma_a/nA     % gamma_a/nA=KpA
207             vA=x(12)*gamma_a/param.nA;
209           %vB= CB*gamma_a/nB     % gamma_a/nB=KpB
211             vB=x(13)*gamma_a/param.nB;
```

```
213             %vC= CC*gamma_a/nC      % gamma_a/nC=KpC

215                 vC=x(14)*gamma_a/param.nC;

217             %vq=Cq*gamma_a/nq

219                vq=x(27)*gamma_a/param.nx;

221
          %translation non_growth protein
223
              %q=vq-mu*q;
225           dxdt(28,1)=vq-x(24)*x(28);

227
  %transcription rates
229
              %wr= wr_max*a/(teta_r+a);
231           wr=param.wr_max*x(16)/(param.teta_r+x(16));

233           %wA= wA_max*a/(teta_nr+a);
              %wA=param.wA_max*x(16)/(param.teta_nr+x(16));
235           wA=param.KmACgA*x(16)/(param.teta_nr+x(16));

237           %wB= wB_max*a/(teta_nr+a);
              %wB=param.wB_max*x(16)/(param.teta_nr+x(16));
239           wB=param.KmBCgB*x(16)/(param.teta_nr+x(16));

241           %wC= wC_max*a/(teta_nr+a);
              %wC=param.wC_max*x(16)/(param.teta_nr+x(16));
243           wC=param.KmCCgC*x(16)/(param.teta_nr+x(16));

245           %wt= wt_max*a/(teta_nr+a);
              wt=param.wx_max*x(16)/(param.teta_nr+x(16));
247
              %wm= wm_max*a/(teta_nr+a);
249            wm=param.wx_max*x(16)/(param.teta_nr+x(16));

251            %wq= (wq_max*a/(teta_nr+a))*(1/(1+q/Kq)^hq));
              wq=(param.wq_max*x(16)/(param.teta_nr+x(16)))*...
253           (1/(1+(x(28)/param.Kq)^param.hq));

255
  %trancription of ribosomas
257           %mr=wr-(mu+dm)*mr+vr-kb*r*mr+ku*Cr          %free mRNA
                dxdt(10,1)=wr-(mu+param.dm)*x(10)+...
```

```
259                     vr-param.kb*x(15)*x(10)+param.ku*x(11);

261             %Cr=-mu*Cr+kb*r*mr-ku*Cr-vr;            %ribosomal-bound complex mRNA
                dxdt(11,1)=-mu*x(11)+param.kb*x(15)*x(10)-param.ku*x(11)-vr;
263
    %trancription of enzymes
265
            %mt=wt-(mu+dm)*mt+vt-kb*r*mt+ku*Ct      %free mRNA
267             dxdt(19,1)=wt-(mu+param.dm)*x(19)+...
                    vt-param.kb*x(15)*x(19)+param.ku*x(21);
269         %Ct=-mu*Ct+Kb*r*mt-Ku*Ct-vt;          %ribosomal-bound complex mRNA
              dxdt(21,1)=-mu*x(21)+param.kb*x(15)*x(19)-param.ku*x(21)-vt;
271
            %mm=wm-(mu+dm)*mm+vm-kb*r*mm+ku*Cm       %free mRNA
273             dxdt(20,1)=wm-(mu+param.dm)*x(20)+...
                    vm-param.kb*x(15)*x(20)+param.ku*x(22);
275
            %Cm=-mu*Cm+Kb*r*mm-Ku*Cm-vm;          %ribosomal-bound complex mRNA
277           dxdt(22,1)=-mu*x(22)+param.kb*x(15)*x(20)-param.ku*x(22)-vm;

279
            %mq=wq-(mu+dm)*mq+vq-kb*r*mq+ku*Cq
281             dxdt(26,1)=wq-(mu+param.dm)*x(26)+...
                    vq-param.kb*x(15)*x(26)+param.ku*x(27);
283
          %Cq=-mu*Cq+Kb*r*mq-Ku*Cq-vq;
285         dxdt(27,1)=-mu*x(27)+param.kb*x(15)*x(26)-param.ku*x(27)-vq;

287

289 %transcription of genes A,B;C

291       %CA = -mu*CA+Kb*r*mA-Ku*CA-vA;
            dxdt(12,1)=-mu*x(12)+param.kbA*x(15)*x(1)-param.kuA*x(12)-vA;
293
          %CB = -mu*CB+Kb*r*mB-Ku*CB-vB;
295         dxdt(13,1)=-mu*x(13)+param.kbB*x(15)*x(5)-param.kuB*x(13)-vB;

297       %CC = -mu*CC+Kb*r*mC-Ku*CC-vC;
          dxdt(14,1)=-mu*x(14)+param.kbC*x(15)*x(7)-param.kuC*x(14)-vC;
299

301 %synthetic gene circuit + metabolic burden

303       %x1 = mRNA_A-> mRNA_LuxR       mA    % free mRNA
```

```
305         %dxdt(1,1)=param.KmACgA - param.dmA*x(1); %-param.mu;

307             dxdt(1,1)=wA - (mu+param.dmA)*x(1) + vA...
                        - param.kbA*x(15)*x(1) + param.kuA*x(12);
309

311         %x2 = A--> LuxR              A protein

313             % Monomer
                c1 = param.dAI + param.k_2;%-param.mu;
315             c2 = c1^2 + 8*param.k3*(param.k2*x(2)*x(3) + 2*param.k_3*x(4));
                c3 = 4*param.k3;
317             M = (1/c3)*(-c1 + sqrt(c2));

319         %dxdt(2,1)= param.kpA*x(1) - param.dA*x(2) - param.k2*x(2)*x(3)...
            %+param.k_2*M;
321
             dxdt(2,1)= vA - (mu+param.dA)*x(2) - param.k2*x(2)*x(3)...
323                         +param.k_2*M;

325         %x3 = AHL

327           dxdt(3,1) = -param.k2*x(2)*x(3) + param.k_2*M + param.kd*x(9)...
                        - x(3)*(param.kd + param.dI)- mu*x(3);
329
            %x4 = (AI)2
331
              dxdt(4,1) = param.k3*M^2 - param.k_3*x(4)- param.dAI2*x(4)...
333                         -mu*x(4);


335
            %x5 = mRNA_B--> mRNA_cI          mB    % free mRNA
337
                %dxdt(5,1) = param.KmBCgB*( x(4)/(param.gamma1 + x(4)))...
339                          %- param.dmB*x(5); %- param.mu*x(5);

341               dxdt(5,1) = wB*(x(4)/(param.gamma1 + x(4)))...
                            - (param.dmB+mu)*x(5) + vB - param.kbB*x(15)*x(5)...
343                          + param.kuB*x(13);

345         %x6 = B--> cI              B protein

347         %dxdt(6,1) = param.kpB*x(5) - param.dB*x(6);

349          dxdt(6,1) = vB - (mu+param.dB)*x(6);
```

```
351
        %x7 = mRNA_C-> GFP          mC    % free mRN
353
             c4 = x(4)+ param.beta1*param.gamma4*x(6) ...
355               + param.beta2*param.gamma5*x(4)*x(6);

357          c5 = param.gamma2+ param.gamma3*x(4)+param.gamma4*x(6) ...
                  + param.gamma5*x(4)*x(6);
359

361        %dxdt(7,1) = param.KmCCgC*(c4/c5)-param.dmC*x(7);%- param.mu*x(7);

363         dxdt(7,1) = wC*(c4/c5)-(param.dmC+mu)*x(7) ...
                        +vC-param.kbC*x(15)*x(7)+param.kuC*x(14);
365

367        %x8 = C                      C protein

369       %dxdt(8,1) = param.kpC*x(7) - param.dC*x(8);

371        dxdt(8,1) = vC - (mu+param.dC)*x(8);

373        %x9 = Ie

375
             Kcells= (param.Vcell*x(30))/param.Vmedium;
377
             dxdt(9,1) = -(Kcells)*param.kd*x(9)+(Kcells)*param.kd*x(3) ...
379                        - param.dIe*x(9);
%                        if (t>=100 && t<103)
381 %              dxdt(9,1) = param.Ie;
%          end
383
  % Finite Ribosomes-translation of ribosomes (free ribosomas dynamics)
385
      % r= 2*vr-mu*r+vt+vm+vq+vA+vB+vC-Kb*r*(mr*mt*mm*mq+mA+mB+mC)+...
387               %Ku*(Cr+Ct+Cm+Cq+CA+CB+CC);

389        dxdt(15,1)=2*vr-mu*x(15)+vt+vm+vq+vA+vB+vC-...
             param.kb*x(15)*(x(10)+x(26)+x(20)+x(19))-...
391          param.kbA*x(15)*x(1)-param.kbB*x(15)*x(5)-param.kbC*x(15)*x(7)+...
             param.ku*(x(27)+x(21)+x(22)+x(11))+...
393           param.kuA*x(12)+param.kuB*x(13)+ param.kuC*x(14);

395 % Finite Energy
```

```
397         % a= ns*Vcat-mu*a-nr*vr-nt*vt-nm*vm-nq*vq-nA*vA-nB*vB-nC*vC;

399             dxdt(16,1)=param.ns*Vcat-mu*x(16)-param.nr*vr-param.nx*vt -...
                           param.nx*vm-param.nx*vq-param.nA*vA-param.nB*vB -...
401                        param.nC*vC;

403 %  Finite Proteome
    %
405 %            %Mp= nr*r+nt*et+nm*em+nq*q+nA*A+nB*B+nC*C+...
    %                 %nr*(Ct+Cm+Cq+Cr+CA+CB+CC);
407 %              Mp=param.nr*x(15)+param.nt*x(17)+param.nm*x(18)+...
    %                 param.nq*x(28)+param.nA*x(2)+param.nB*x(6)+param.nC*x(8)+...
409 %                 +param.nr*(x(11)+x(12)+x(13)+x(14)+x(27)+x(21)+x(22))
          dxdt(25,1)=0;
411       dxdt(24,1)=0;


413
 %cells number
415        %N=mu*N
      dxdt(30,1)=mu*x(30);
417
     %x31 is used to calculate J1 (protein C)
419     dxdt(31,1)=abs(vC - (mu+param.dC)*x(8));

421     %x32 is used to calculate J3 (protein C)
        dxdt(32,1)=t*(abs(vC - (mu+param.dC)*x(8)));
423



425
end
```

archivos_m/model_mburden_adap_v3.m

A.2 Simulation Configuration

```
2 %%
  clc;clf;clearvars;close all;
4
  %%% parameters for the model
6     param.dmA= log(2)/2;          % [1/min]                    % mA degradation rate
      param.dmB= log(2)/2;          % [1/min]                    % mB degradation rate
8     param.dmC= log(2)/2;          % [1/min]                    % mC degradation rate

10    param.kpA= 50;              % [1/min]                    % mA translation rate
```

```
param.kpB= 40;              % [1/min]  range [1 200]     % mB translation rate
param.kpC= 15;              % [1/min]  range [1 200]     % mC translation rate

param.dA= 0.01;             % [1/min]                    % A degradation rate
param.dB= log(2)/4;           % [1/min]  range [0.01 0.3]   % B degradation rate
param.dC= log(2)/4;             % [1/min]  range [0.01 0.3]   % C degradation
rate

param.kd= 2;                % [1/min]                    % inducer diffusion
rate

param.k2= 0.06;             % [1/min]                    % (AI) association
rate
param.k3= 0.01;             % [1/min]                    % (AI)2 association
rate

param.k_2= 0.2;             % [1/min]                    % (AI) dissociation
rate
param.k_3= 0.9;              % [1/min]                   % (AI)2 dissociation
rate

param.gamma1= 50;       % [nM]      range [50 200]     % gB promoter Hill
constant
param.gamma2= 0.02;         % [nM]                     % gC promoter
coefficient
param.gamma3= 0.1;          % [adim]   range [1e-4 0.5]   % gC promoter
coefficient
param.gamma4= 1.42;           % [adim]   range [5e-4 5]     % gC promoter
coefficient
param.gamma5= 70;         % [n/M]     range [1 100]      % gC promoter
coefficient

param.beta1= 0.05;          % [adim]                   % gC promoter basal
expression coefficient
param.beta2= 0.05;          % [n/M]                    % gC promoter basal
expression coefficient

param.dI=  0.001;           % [1/min]                  % Inducer(AHL) degradation
 rate
param.dIe= 0.001;         % [1/min]                  % Inducer (AHL
extracellular Inducer) degradation rate

param.dAI=  0.05;           % [1/min]                  % (AI)(Monomer)
degradation rate
param.dAI2= 0.005;          % [1/min]                  % (AI)2 (Dimer)
degradation rate
```

```
40
    param.KmACgA= 30;          % [1/min]     range [1 200]   %Promoter strength and
    plasmid origin of replication
42  param.KmBCgB= 10;          % [1/min]     range [1 200]   %Promoter strength and
    plasmid origin of replication
    param.KmCCgC= 30;          % [1/min]     range [1 200]   %Promoter strength and
    plasmid origin of replication
44
    param.Ncells=2.4e8*0.18;       % [cells]                 % number of cells in a 180
     L culture with ODOD(optical density) = 0.3 placed
46  param.Vcell=1e-15;             % [L]                     %  typical volume of an E.
     coli
    param.Vmedium=180e-6;          % [L]                     % (culture medium)
48
    % nA=6.022e23;                   % [1/mols]              %avogadro number
50   %param.mu=0.017;               % [1/min]               %Specific growth rate


52
%??????????????????????????????????????????????????????????????????????????????
54 %     1mol of moleculas are 6.022e23 (avogadro number moleculas
   %     factor=1e15/(param.Vcell*nA);
56 %??????????????????????????????????????????????????????????????????????????????


58
 %Parameters Metabolic Burden Model

60
%??????????????????????????????????????????????????????????????????????????????
62    %Here considerer all varaiables in molecules pere cell. assume a fixed
      %volume of 1?m^3=1e-15L to convert to numbers of molecules.
64    %This in original paper (WeiBe at 2015)
%??????????????????????????????????????????????????????????????????????????????
66
    %param.S=1e4;           %[molecs]              external nutrient
68  param.dm= 0.1;          %[1/min]               mRNA- degrdation rate
    param.ns=85.9;           %[adim]                 nutrient efficiency
70  param.vt_max=25.8;      %[1/min]               max.nutrient import rate
    param.Kt=501.2;         %[molecs]              nutrient import half-maximal
    thresholds
72  param.vm_max=1000;      %[1/min]               max.enzymatic metabolizes rate
    param.Km=1000;          %[molecs/cell]         enzymatic half-maximal thresholds
74  param.nr=7459;          %[aa/molecs]           ribosome length (in amino acids)
    param.nx=300;           %[aa/molecs]           length of q,et,em  protein(in
    amino acids)
76  param.nA=756/3;           %[aa/molecs]           length of A   (in amino acids)
    param.nB=775/3;           %[aa/molecs]           length of B    (in amino acids)
78  param.nC=720/3;           %[aa/molecs]           length of C   (in amino acids)
```

```
    param.gamma_max=959;  %[aa/min molecs]       max. translation elongation rate
80  param.Kgamma=3e8;     %[molecs/cell]         translation elongation threshold

82  param.ku=1;           %[1/min]               mRNA-ribosome unbinding rate
    param.kb=1;           %[aa/min molecs]       mRNA-ribosome binding rate
84
    param.kuA=1;          %[1/min]               mRNA-ribosome unbinding rate
86  param.kbA=1;          %[aa/min molecs]       mRNA-ribosome binding rate
    param.kuB=1;          %[1/min]               mRNA-ribosome unbinding rate
88  param.kbB=1;          %[aa/min molecs]       mRNA-ribosome binding rate
    param.kuC=1;          %[1/min]               mRNA-ribosome unbinding rate
90  param.kbC=1;          %[aa/min molecs]       mRNA-ribosome binding rate

92  param.wr_max=930;
    param.wx_max=4.14;    %[molecs/min cell]     max. enzyme that transport & max.
     enzyme that metabolism transcription rate
94  param.wq_max=948.93;  %[molecs/min cell]     max. house-keeping protein
    transcription rate

96% param.wA_max=1;       %[1/min]               max. A transcrption rate
%   param.wB_max=1;       %[1/min]               max. B transcrption rate
98% param.wC_max=1;       %[1/min]               max. C transcrption rate

100 param.Kq=152219;      %[molecs/cell]         q-autoinhibition half-maximal
    thresholds
    param.hq=4;           %[adim]                house-keeping autoinhibition Hill
     coef.
102 param.teta_r=426.87;  %[molecs/cell]         ribosome transcription threshold
    param.teta_nr=4.38;   %[molecs/cell]         non-ribosomal transcription
    threshold
104 param.Mp= 1e8;        %[aa]                  total cell mass

106% param.kin=1;
%   param.ds=1;
108
%% Initial conditions
110
    mA =431.253835618893;%32.7594914873419;   %61.9833444332460
112 A = 199734.733848284; %4075.00573700777
    I = 1.69038634676277e-10;
114 AI2=9.686790813919655e-12;
    mB  =6.40021353492325e-11;
116 B =5.08348701409222e-09;
    mC=8.00022307927807e-06;
118 C=0.000654301947591275;
    %param.Ie=5;%5.4198e11;
```

```
120     Ie=5;
        mr=40.1152061492874;%51.1884334330967;
122     Cr=762.022877743964;%646.868152594654;
        CA=2252.44338219681;%164.922608608980; %106.073868024521
124     CB=3.44373756246120e-10;%0;
        CC=4.02677030917859e-05;%0;
126     r =21.4428193682806;%14.1360010184276;
        a =11.2359053341826;%9.6;
128     et=2000;%6812.52991481445;%6062.75109887421;
        em=6812.52991481427;%6062.75109887421;
130     mt=10.9698347280545;%13.4068591000857;
        mm=10.9698347280574;%13.4068591000857;
132     Ct=65.1133820317549;%54.8838800677396;
        Cm=65.1133820317723;%54.8838800677396;
134     Si=143.082263755735;%143.082262120182;
        mu1=0;
136     Mp=0;
        mq=6.85608213753247;%505.391263549140;
138     Cq=40.7027152097119;%2068.93;
        q=668423.568083377;%228544.315679755;
140     S=1e12;                     %[molecs]              external nutrient
        N=1000; %param.Ncells/3;  %Equivlent to an intial OD=0.1
142     sensi=0;
        J3_0=0;
144
        X0= [mA;A;I;AI2;mB;B;mC;C;Ie;mr;Cr;CA;CB;CC;r;a;et;em;mt;mm;Ct;Cm;Si;mu1;Mp;mq;
        Cq;q;S;N;sensi;J3_0];
146


148 %% Simulation

150     %set an error
        options=odeset('AbsTol',1e-6,'RelTol',1e-4);
152     tfinal=400;
        tspan = [0, tfinal];
154
        %call the solver
156     %tic
        [t,X] = ode15s(@(t,x)model_mburden_adap_v3(t,x,param),tspan,X0,options);
158     %toc


160
    %% Figures
162
        figure1 = figure('Color',[1 1 1]);
164
```

```
        subplot1 = subplot(3,3,1,'Parent',figure1,'FontName','Arial');
166         box(subplot1,'on');
            grid(subplot1,'on');
168         hold(subplot1,'all');
            plot(t,X(:,1),'b','LineWidth',1.5);
170         title('\textbf{mA (mRNALuxR)} ','Interpreter','latex');
           %ylabel('{Concentration [nM]}','Interpreter','latex');
172        %axis([0 500 0 0.13]);
            ylabel('{Concentration [nM]}');
174         xlabel('time [min]');

176     subplot2 = subplot(3,3,2,'Parent',figure1,'FontName','Arial');
            box(subplot2,'on');
178         grid(subplot2,'on');
            hold(subplot2,'all');
180         plot(t,X(:,2),'b','LineWidth',1.5);
            title('\textbf{A (LuxR)}','Interpreter','latex');
182        %ylabel('${\pi}$','Interpreter','latex');
           %axis([0 500 0 0.0020]);
184         xlabel('time [min]');
        subplot3 = subplot(3,3,3,'Parent',figure1,'FontName','Arial');
186         box(subplot3,'on');
            grid(subplot3,'on');
188         hold(subplot3,'all');
            plot(t,X(:,3),'b','LineWidth',1.5);
190        %title('${\sigma}$(Cs)','Interpreter','latex');
            title('\textbf{I (AHL)}','Interpreter','latex');
192         ylabel('{Concentration [nM]}');
           %ylabel('${\sigma}$','Interpreter','latex');
194        %axis([0 500 0 0.35]);
            xlabel('time [min]');
196    subplot4 = subplot(3,3,4,'Parent',figure1,'FontName','Arial');
            box(subplot4,'on');
198         grid(subplot4,'on');
            hold(subplot4,'all');
200           plot(t,X(:,4),'b','LineWidth',1.5);
           %title('${\sigma}$(Cs)','Interpreter','latex');
202         title('\textbf{AI2 (Dimer)}','Interpreter','latex');
           %axis([0 500 0 0.35]);
204         xlabel('time [min]');
             ylabel('{Concentration [nM]}');
206
         subplot5 = subplot(3,3,5,'Parent',figure1,'FontName','Arial');
208         box(subplot5,'on');
            grid(subplot5,'on');
210         hold(subplot5,'all');
```

```
            plot(t,X(:,5),'b','LineWidth',1.5);
212         %title('${\sigma}$(Cs)','Interpreter','latex');
            title('\textbf{mB (mRNAcI)}','Interpreter','latex');
214         xlabel('time [min]');
            %axis([0 500 0 0.35]);
216     subplot6 = subplot(3,3,6,'Parent',figure1,'FontName','Arial');
            box(subplot6,'on');
218         grid(subplot6,'on');
            hold(subplot6,'all');
220         plot(t,X(:,6),'b','LineWidth',1.5);
            %title('${\sigma}$(Cs)','Interpreter','latex');
222         %title('B (cI)','Interpreter','latex','FontSize',14,'FontWeight','Bold
    ','FontName','Arial');
            title('\textbf{B (cI)}','Interpreter','latex');
224         %axis([0 500 0 0.35]);
            xlabel('time [min]');
226     subplot7 = subplot(3,3,7,'Parent',figure1,'FontName','Arial');
            box(subplot7,'on');
228         grid(subplot7,'on');
            hold(subplot7,'all');
230         plot(t,X(:,7),'b','LineWidth',1.5);
            %title('${\sigma}$(Cs)','Interpreter','latex');
232         title('\textbf{mC (cmRNAgfp)}','Interpreter','latex');
            %axis([0 500 0 0.35]);
234         ylabel('{Concentration [nM]}');
            xlabel('time [min]');
236
        subplot8 = subplot(3,3,8,'Parent',figure1,'FontName','Arial');
238         box(subplot8,'on');
            grid(subplot8,'on');
240         hold(subplot8,'all');
            plot(t,X(:,8),'b','LineWidth',1.5);
242         %title('${\sigma}$(Cs)','Interpreter','latex');
            title('\textbf{C (GFP)}','Interpreter','latex');
244         %axis([0 500 0 0.35]);
            xlabel('time [min]');
246     subplot9 = subplot(3,3,9,'Parent',figure1,'FontName','Arial');
            box(subplot9,'on');
248         grid(subplot9,'on');
            hold(subplot9,'all');
250         plot(t,X(:,9),'b','LineWidth',1.5);
            %title('${\sigma}$(Cs)','Interpreter','latex');
252         title('\textbf{Ie (AHLext)}','Interpreter','latex');
            %axis([0 500 0 0.35]);
254         xlabel('time [min]');
```

```
256   figure2 = figure('Color',[1 1 1]);

258       subplot1 = subplot(2,2,1,'Parent',figure2,'FontName','Arial');
              box(subplot1,'on');
260           grid(subplot1,'on');
              hold(subplot1,'all');
262           plot(t,X(:,10),'b','LineWidth',1.5);
              title('\textbf{mr (mRNA ribosomes)} ','Interpreter','latex');
264           %ylabel('{Concentration [nM]}','Interpreter','latex');
              %axis([0 500 0 0.13]);
266            xlabel('time [min]');

268       subplot2 = subplot(2,2,2,'Parent',figure2,'FontName','Arial');
              box(subplot2,'on');
270           grid(subplot2,'on');
              hold(subplot2,'all');
272           plot(t,X(:,15),'b','LineWidth',1.5);
              %title('${\sigma}$(Cs)','Interpreter','latex');
274           %title('B (cI)','Interpreter','latex','FontSize',14,'FontWeight','Bold
    ','FontName','Arial');
              title('\textbf{r (free ribosomes dynamics)}','Interpreter','latex');
276           %axis([0 500 0 0.35]);
              xlabel('time [min]');
278
       subplot3 = subplot(2,1,2,'Parent',figure2,'FontName','Arial');
280           box(subplot3,'on');
              grid(subplot3,'on');
282           hold(subplot3,'all');
              plot(t,X(:,11),'r','LineWidth',1.5);
284           plot(t,X(:,21),'k','LineWidth',1.5);
              plot(t,X(:,22),'c','LineWidth',1.5);
286           plot(t,X(:,12),'m','LineWidth',1.5);
              plot(t,X(:,13),'g','LineWidth',1.5);
288           plot(t,X(:,14),'b','LineWidth',1.5);
              plot(t,X(:,27),'--g','LineWidth',1.5);
290            legend('Cr','Ct','Cm','cA-LuxR','cB-cI','cC-GFP','Cq');
              title('\textbf{Cx-(Ribosomal Complexs)}','Interpreter','latex');
292           %ylabel('${\pi}$','Interpreter','latex');
              %axis([0 500 0 0.0020]);
294           xlabel('time [min]');

296   figure3 = figure('Color',[1 1 1]);

298       subplot1 = subplot(3,3,1,'Parent',figure3,'FontName','Arial');
              box(subplot1,'on');
300           grid(subplot1,'on');
```

```
            hold(subplot1,'all');
302         plot(t,X(:,29),'b','LineWidth',1.5);
            title('\textbf{S (External nutrient} ','Interpreter','latex');
304         %ylabel('{Concentration [nM]}','Interpreter','latex');
            %axis([0 500 0 0.13]);
306         xlabel('time [min]');

308      subplot2 = subplot(3,3,2,'Parent',figure3,'FontName','Arial');
            box(subplot2,'on');
310         grid(subplot2,'on');
            hold(subplot2,'all');
312         plot(t,X(:,23),'b','LineWidth',1.5);
            %title('${\sigma}$(Cs)','Interpreter','latex');
314         title('\textbf{Si (Internal nutrient)}','Interpreter','latex');
            ylabel('{Concentration [nM]}');
316         %axis([0 500 0 0.35]);
             xlabel('time [min]');
318
         subplot3 = subplot(3,3,3,'Parent',figure3,'FontName','Arial');
320         box(subplot3,'on');
            grid(subplot3,'on');
322         hold(subplot3,'all');
             plot(t,X(:,16),'b','LineWidth',1.5);
324         %title('${\sigma}$(Cs)','Interpreter','latex');
            title('\textbf{a (Energy)}','Interpreter','latex');
326         %axis([0 500 0 0.35]);
             xlabel('time [min]');
328
%          subplot3 = subplot(3,3,6,'Parent',figure3,'FontName','Arial');
330 %           box(subplot3,'on');
%           grid(subplot3,'on');
332 %           hold(subplot3,'all');
%           plot(t,X(:,25),'r','LineWidth',1.5);
334 %           title('\textbf{Mp}','Interpreter','latex');
%           %ylabel('{Concentration [nM]}','Interpreter','latex');
336 %           xlabel('time [min]');

338      subplot4 = subplot(3,3,4,'Parent',figure3,'FontName','Arial');

340      gamma_a=(param.gamma_max.*X(:,16))./(param.Kgamma+X(:,16));

342      mu=((gamma_a)./param.Mp).*...
            (X(:,11)+X(:,12)+X(:,13)+X(:,14)+X(:,27)+X(:,21)+X(:,22));
344
            box(subplot4,'on');
346         grid(subplot4,'on');
```

```
            hold(subplot4,'all');
348         plot(t,mu,'b','LineWidth',1.5);
            %title('${\sigma}$(Cs)','Interpreter','latex');
350         title('${\mu}$ \textbf{(growth rate)} ','Interpreter','latex');
            %axis([0 500 0 0.35]);
352          xlabel('time [min]');

354       subplot5 = subplot(3,3,5,'Parent',figure3,'FontName','Arial');
            box(subplot5,'on');
356         grid(subplot5,'on');
            hold(subplot5,'all');
358         plot(t,X(:,30)./1.44e8,'b','LineWidth',1.5);
            title('\textbf{OD (cells number)} ','Interpreter','latex');
360         %ylabel('{Concentration [nM]}','Interpreter','latex');
            %axis([0 500 0 0.13]);
362          xlabel('time [min]');

364   figure4 = figure('Color',[1 1 1]);

366          gamma_a= (param.gamma_max*X(:,16))./(param.Kgamma+X(:,16));
             vr=X(:,11).*gamma_a./param.nx;
368          vt=X(:,21).*gamma_a./param.nx;
             vm=X(:,22).*gamma_a./param.nx;
370          vA=X(:,12).*gamma_a./param.nA;
             vB=X(:,13).*gamma_a./param.nB;
372          vC=X(:,14).*gamma_a./param.nC;
             vq=X(:,27).*gamma_a./param.nx;
374
        subplot1 = subplot(2,1,1,'Parent',figure4,'FontName','Arial');
376          box(subplot1,'on');
             grid(subplot1,'on');
378          hold(subplot1,'all');
             plot(t,vr(:,1),'r','LineWidth',1.5);
380          plot(t,vt(:,1),'k','LineWidth',1.5);
             plot(t,vm(:,1),'c','LineWidth',1.5);
382          plot(t,vA(:,1),'m','LineWidth',1.5);
             plot(t,vB(:,1),'g','LineWidth',1.5);
384          plot(t,vC(:,1),'b','LineWidth',1.5);
             plot(t,vq(:,1),'k','LineWidth',1.5);
386          legend('vr','vt','vm','vA','vB','vC','vq');
             %title('${\sigma}$(Cs)','Interpreter','latex');
388          title('\textbf{Vx (translation rates)}','Interpreter','latex');
             ylabel('{Concentration [nM]}');
390          %axis([0 500 0 0.35]);
              xlabel('time [min]');
392
```

```
    subplot2 = subplot(2,1,2,'Parent',figure4,'FontName','Arial');

        wr=param.wr_max.*X(:,16)./(param.teta_r+ X(:,16));
        wt=param.wx_max.*X(:,16)./(param.teta_nr+X(:,16));
        wm=param.wx_max.*X(:,16)./(param.teta_nr+X(:,16));
        wA=param.KmACgA.*X(:,16)./(param.teta_nr+X(:,16));
        wB=param.KmBCgB.*X(:,16)./(param.teta_nr+X(:,16));
        wC=param.KmCCgC.*X(:,16)./(param.teta_nr+X(:,16));
        wq=(param.wq_max.*X(:,16)./(param.teta_nr+X(:,16))).*...
        (1./(1+(X(:,28)./param.Kq).^param.hq));

        box(subplot2,'on');
        grid(subplot2,'on');
        hold(subplot2,'all');
        plot(t,wr(:,1),'r','LineWidth',1.5);
        plot(t,wt(:,1),'k','LineWidth',1.5);
        plot(t,wm(:,1),'c','LineWidth',1.5);
        plot(t,wA(:,1),'m','LineWidth',1.5);
        plot(t,wB(:,1),'g','LineWidth',1.5);
        plot(t,wC(:,1),'b','LineWidth',1.5);
        plot(t,wq(:,1),'k','LineWidth',1.5);
        legend('wr','wt','wm','wA','wB','wC','wq')
        %title('${\sigma}$(Cs)','Interpreter','latex');
        title('\textbf{Wx (transcription rates)}','Interpreter','latex');
        ylabel('{Concentration [nM]}');
        %axis([0 500 0 0.35]);
        xlabel('time [min]');

figure5 = figure('Color',[1 1 1]);

        Vimp= X(:,17).*(param.vt_max*X(:,29))./(param.Kt+X(:,29));
        Vcat= X(:,18).*(param.vm_max*X(:,23))./(param.Km+X(:,23));

    subplot1 = subplot(2,2,1,'Parent',figure5,'FontName','Arial');
        box(subplot1,'on');
        grid(subplot1,'on');
        hold(subplot1,'all');
        plot(t,X(:,17),'b','LineWidth',1.5);
        title('\textbf{et (transporter enzyme)} ','Interpreter','latex');
        %ylabel('{Concentration [nM]}','Interpreter','latex');
        %axis([0 500 0 0.13]);
        xlabel('time [min]');

    subplot2 = subplot(2,2,2,'Parent',figure5,'FontName','Arial');
        box(subplot2,'on');
        grid(subplot2,'on');
```

```
            hold(subplot2,'all');
            plot(t,X(:,18),'b','LineWidth',1.5);
            title('\textbf{em (metabolic enzyme)}','Interpreter','latex');
            %ylabel('${\pi}$','Interpreter','latex');
            %axis([0 500 0 0.0020]);
             xlabel('time [min]');

        subplot3 = subplot(2,2,3,'Parent',figure5,'FontName','Arial');
            box(subplot3,'on');
            grid(subplot3,'on');
            hold(subplot3,'all');
            plot(t,Vimp,'b','LineWidth',1.5);
            %title('${\sigma}$(Cs)','Interpreter','latex');
            title('\textbf{Vimp (nutrient import)}','Interpreter','latex');
            %ylabel('{Concentration [mM]}');
            %ylabel('${\sigma}$','Interpreter','latex');
            %axis([0 500 0 0.35]);
             xlabel('time [min]');

      subplot4 = subplot(2,2,4,'Parent',figure5,'FontName','Arial');
            box(subplot4,'on');
            grid(subplot4,'on');
            hold(subplot4,'all');
            plot(t,Vcat,'b','LineWidth',1.5);
            title('\textbf{Vcat (metabolism)}','Interpreter','latex');
            %title('${\sigma}$(Cs)','Interpreter','latex');
           % title('\textbf{Ie (Vcat)}','Interpreter','latex');
            %axis([0 500 0 0.35]);
             xlabel('time [min]');

figure(figure1)
%close(figure2)
%close(figure3)
%close(figure4)
%close(figure5)
```

archivos_m/sim_model_mburden_adap_v3.m