UNIVERSIDADE DE SÃO PAULO
INSTITUTO DE FÍSICA DE SÃO CARLOS
PROGRAMA INTERUNIDADES DE PÓS-GRADUAÇÃO EM BIOINFORMÁTICA

PAULO E. P. BURKE

SIMULATION OF BIOCHEMICAL SYSTEMS USING CONSTRAINT-BASED METHODS AND COMPLEX NETWORKS

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001

São Carlos - SP
March 2021

PAULO EDUARDO PINTO BURKE

Simulation of Biochemical Systems Using Constraint-Based Methods and Complex Networks

Versão Original

Tese de Doutorado apresentada ao Programa Interunidades de Pós-Graduação em Bionformática da Universidade de São Paulo para obtenção do título de Doutor em Ciências.

Área de Concentração: Bioinformática

Orientador: Prof. Dr. Luciano da Fontoura Costa

São Carlos - SP
Março 2021

FICHA CATALOGRÁFICA

Burke, Paulo Eduardo Pinto
B959 Simulation of biochemical systems using constraint-based methods and complex networks / Paulo Eduardo Pinto Burke, orientador Prof. Luciano da Fontoura Costa.
São Carlos : 2021.
89 p.

Tese (Doutorado) - Universidade de São Paulo
Orientador: Prof. Dr. Luciano da Fontoura Costa
Programa Interunidades de Pós-Graduação em Bioinformática
Área de concentração: Bioinformática

1. Biochemical networks. 2. Process integration. 3. Stochastic simulation. 4. Whole-cell models. I.Costa, Luciano da Fontoura, orientador II.Universidade de São Paulo. III.Título.

CDD: 572.8

Elaborada pelo Serviço de Informação e Biblioteca Carlos Benjamin de Lyra do IME-USP, pela Bibliotecária Maria Lucia Ribeiro CRB-8/2766

ABSTRACT

Computational models of biomolecular systems have been employed to advance knowledge in many research areas. They have particular impact in modern areas such as Synthetic Biology, Bioengineering, and Precision Medicine. Current technologies can determine with high precision and throughput the molecules that compose cells and, to a great extent, the interactions between them. These interactions are usually grouped into cellular processes by their function and many are the available models that can represent and simulate them to a certain level of accuracy. Despite being commonly represented as separated systems, they are in fact interconnected. The simple fact that they may share common molecules makes their dynamics dependent on each other. Given the current high availability of data and computational power, models to represent whole-cells are being considered. However, current approaches to model and simulate cellular processes are challenging to be integrated given the high heterogeneity of methods employed. Thus, a more homogeneous approach to represent and simulate could make easier this integration. In this work, we propose a framework to model cellular processes by means of their underlying biochemical reactions as well as a simulation method that sources from this kind of representation. To investigate the capabilities of the modeling framework, we used the organism *Mycoplasma genitalium* as a case study aiming at representing all the molecules and interactions known to compose this organism by means of a single biochemical network. Among the results obtained from this model, we have that the obtained topology presents a good agreement with the literature, as well as good accuracy on the prediction of essential genes of the organism by employing cascade failure analysis. Additionally, we investigated the characteristics and capabilities of the so proposed simulation algorithm, called CBSA. It is shown to be able to perform efficiently discrete-stochastic evaluations of the dynamics of large sets of interactions. It is also able to be computed in parallel computing architectures such as GP-GPUs. We illustrate this by simulating several theoretical models as well as a challenging real biochemical system. Despite the advances reported in this work, much remains to be done in order to perform simulations at a whole-cell scale by using the proposed methods. Nevertheless, we point out possible future developments aiming at the ultimate goal of performing simulations of whole-cells.

RESUMO

Modelos computacionais de sistemas biomoleculares têm sido utilizados para avançar o conhecimento em muitas áreas do conhecimento. Eles têm um impacto particular em áreas modernas como a Biologia Sintética, Bioengenharia e Medicina de Precisão. As tecnologias atuais podem determinar com alta precisão e eficiência as moléculas que compõem células e, em grande parte, as interações entre elas. Essas interações são geralmente agrupadas em processos celulares por suas funções e muitos modelos que podem representá-los e simulá-los com um certo nível de precisão estão disponíveis na literatura. Apesar de serem comumente representados como sistemas separados, eles estão na verdade interconectados. O simples fato de que eles podem compartilhar espécies moleculares torna suas dinâmicas dependentes umas das outras. Dada a alta disponibilidade atual de dados e poder computacional, modelos para representar células completas estão sendo construídos. No entanto, a integração das abordagens atuais para modelar e simular processos celulares é dificultada devido à alta heterogeneidade dos métodos empregados. Assim, uma abordagem mais homogênea para representar e simular processos celulares poderia facilitar essa integração. Neste trabalho, propomos um framework para modelar processos celulares por meio de suas reações bioquímicas subjacentes, bem como um método de simulação que utiliza esse tipo de representação como base. Para investigar as capacidades do framework de modelagem, utilizamos o organismo *Mycoplasma genitalium* como estudo de caso com o objetivo de representar todas as moléculas e interações conhecidas que compoem este organismo por meio de uma única rede bioquímica. Entre os resultados obtidos com este modelo estão as semelhanças encontradas entre sua topologia e as descritas na literatura, bem como a predição de genes essenciais do organismo por meio de uma análise de falha em cascata. Adicionalmente, investigamos as características e capacidades do algoritmo de simulação proposto, denominado CBSA. Foi demonstrado que este algoritmo é capaz de calcular simulações estocásticas discretas de grandes conjuntos de interações de forma eficiente. Ele também pode ser calculado em arquiteturas de computação paralela, como GP-GPUs. Ilustramos isso simulando vários modelos teóricos, bem como um desafiador sistema bioquímico real. Apesar dos avanços relatados neste trabalho, muito ainda precisa ser feito para realizar simulações em escala de célula completa utilizando os métodos propostos. No entanto, apontamos possíveis desenvolvimentos futuros deste trabalho visando o objetivo final de realizar simulações de células completas.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

Humans always tried to understand and explain how living systems work. It helps us not only to answer our deepest philosophical questions but also provides means to modify and enhance our quality of life. Since the discovery of the smallest living unit, the cell, many researchers have focused on describing its components and functioning. However, such tasks have shown not to be trivial, even for the simplest forms of life [1].

Many advances have been made in order to provide precise descriptions of cellular contents. All cells are now known to be made of basic molecules, such as nucleic acids, amino acids, lipids, sugars, and so on, that are combined to form larger structures [2]. The different combinations of these basic building blocks lead to almost infinite possibilities, each organism having a similar but unique combination. The similarities between these structures allow us to develop techniques to evaluate the differences between them. For example, all living cells have a similar way to combine nucleic acids forming what is called a DNA molecule. It is a polymer composed of specific sequences of adenine (A), thymine (T), cytosine (C), and guanine (G) nucleic acids. The sequences are characteristic to each organism and are known to provide the basis for the formation of most other complex structures inside cells [2]. Although the basic knowledge about cell composition is already well established, the current challenge lays in obtaining precisely which structures compose each organism. Also, the structures that can be found inside a cell can vary in time.

The set of molecules that can be found inside a cell at a given time (e. g.DNA, RNA, and proteins) is a result of interactions between the molecules available in a previous moment. The interaction between molecules is what drives the dynamics of an intracellular environment but evaluating them is even more challenging than describing the molecules themselves. To reduce the complexity of the task, the usual approach is to divide these interactions into groups according to their function in the cell. These groups are called cellular processes [2]. For example, the set of reactions that involves enzymes to transform small molecules is called metabolism. The interactions between proteins and the DNA to produce RNA are grouped into the transcription process. Many are the processes already described inside cells and the understanding of their dynamics is fundamental to the understanding of an entire organism.

One approach to understanding any natural phenomenon is to make models of it. A model can be understood as a simplified representation of the real system that can explain its functioning to a given acceptable accuracy [3]. To understand cellular processes, several respective models have been proposed with ever-increasingly precision and detail. Representations of such models can be obtained by different means, such as networks [4], mathematical equations [5], and even circuits [6], also considering different levels of abstraction. Once having a representation of a given system, we can explore its dynamics through simulations [7]. It is a present

challenge to have these representations of different cellular processes integrated in such a way that the behavior of an entire cell could be investigated as a whole [8, 9]. In Figure 1.1 we illustrate the current problem of producing compatible models of different cellular processes built on different sources of data, having them integrated, and its dynamics evaluated as one unified model. Nevertheless, recent advances are providing means to the construction and simulation of integrated representations of cells [10]. These have been achieved by using hybrid approaches to interchange information between processes represented and simulated using different approaches. These large-scale cellular models are currently called whole-cell models.

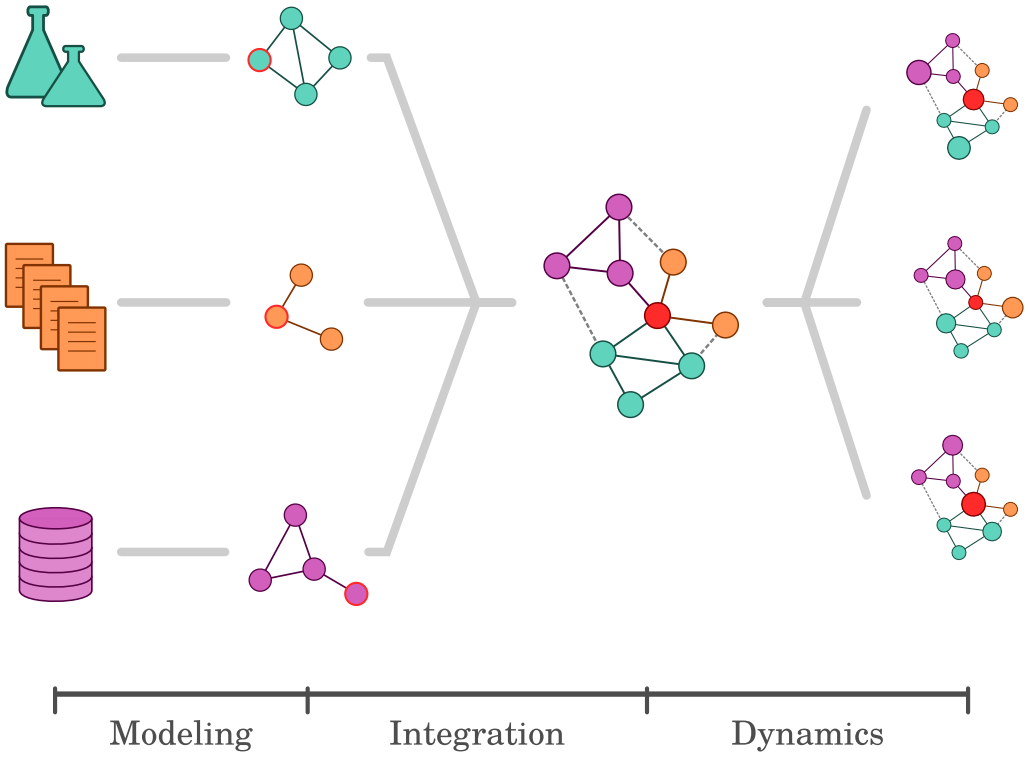

Figure 1.1: The scheme of an ideal workflow for modeling data from different processes and sources with a common approach, integrate them by their shared components, and evaluate the dynamics of the system as one.

Even though these cellular processes are generally represented as separated systems at different levels of abstraction, they all can be ultimately reduced to their underlying biochemical interactions. These interactions, even if considered by us as part of different cellular processes, at some point will have overlapping reactants and products, therefore, interconnecting them. This common ground could be used as a means to naturally represent and integrate such systems. Several methodologies have been developed in this direction but mainly focused on metabolism and signaling [11]. Some of them, such as rule-based modeling [12], are proving to be powerful tools to represent the highly combinatorial nature of biomolecular systems. Additionally, biochemical networks can be stored using community standards, such as SBML [13], expanding the usability of them. This last feature has been shown to be a barrier for current hybrid approaches on whole-cell models [14].

The methods employed to simulate biochemical systems are highly dependent on how they are modeled in the first place. On hybrid approaches, the diversity of representation leads to a necessary heterogeneity of simulation methods. This increases the interfacing complexity between models that might lead to some integration challenges, such as different time scales and competition for resources. If all cellular processes could be represented using the same approach, e. g.biochemical networks, they could be ideally simulated using the same methodology, thus overcoming some of the integration issues.

Many are the simulation methods that can source from biochemical networks [15]. However, their particular assumptions may limit their applications. For instance, ODEs can be derived from biochemical networks but are suitable only for systems where the number of copies of molecular species is high enough and the stochastic fluctuations are negligible [16]. This is because the molecular counts are represented as continuous variables of their concentrations, thus being not suitable for describing systems with low molecular counts where stochasticity plays an important role. On the other hand, methods such as the Stochastic Simulation Algorithm (SSA) can account for low molecular counts and their associated stochastic behavior, but this comes with a high computational cost to evaluate the dynamics of larger biochemical systems [17]. Other approaches based on constraint-modeling, such as the Dynamic Flux Balance Analysis (dFBA) [18, 19], can perform efficient and parameterless simulations of large systems but are subject to strong prior assumptions about the system's functioning.

The emergence of detailed computational models of whole-cells is believed to mark the beginning of a new era of computer-aided biological research [20]. Despite the promising potentials of initial whole-cell models, much needs to be improved for these models to become tools in everyday lab routines. One important step in this direction would be the development of more homogeneous approaches to model and simulate cellular processes, thus being the focus of the present work.

1.1 MOTIVATION

Computational models of biological cells have been produced for decades [21]. The level of detail and predictive capabilities are ever-growing along with the availability of experimental data and advances in computational power [22]. In 2012, Karr *et al.* published a comprehensive whole-cell computational model of a simple bacteria that was capable of predicting its phenotype from given alterations in genotype [10]. The model accounted for all the 525 genes of the *Mycoplasma genitalium* organism as well as for all known cellular processes. By adopting a hybrid approach, cellular processes were separately modeled and simulated using the respective more appropriate methods available in the literature and further integrated by exchanging information from time to time. They were able to predict several aspects of the organism such as growth rate and concentrations of metabolites, proteins, and RNAs. They also could predict with a fair accuracy the effects of gene deletions observed experimentally. Although *M. genitalium* is a relatively simple organism, yielding the smallest known genome in nature, the resulting computational model is already highly complex and its study can contribute to medical and synthetic biology applications. More recently, comprehensive whole-cell models of more

complex organisms, such as *Escherichia coli* [23] and *Sacharomices cerevisiae* [24], have also been published using similar approaches.

The mentioned whole-cell computational models are paving the way for computer-aided advances in synthetic biology, precision medicine, and bioengineering [25–27]. However, current modeling and simulation approaches are capable of providing only highly specialized models making it difficult to extend or adapt them to other organisms. In other words, if one wants to build a model of a different organism using current approaches, it must be done almost from scratch. One of the characteristics of these models that makes it so difficult to be adapted is that each cellular process is modeled and simulated by very distinct methodologies that might not be appropriate to every organism. Thus, we believe that more integrative and homogeneous approaches for both modeling and simulation of organisms could provide more scalable and transferable models.

1.2 OBJECTIVES

The main goal of this project is to investigate approaches for homogeneous modeling and simulation of biochemical systems in order to provide a common framework to integrate different cellular processes. This is an attempt to enhance current methodologies applied to the study and computer simulation of whole-cells. The main goal can be detailed into the following more specific objectives:

- *Homogeneous Modeling of Biochemical Interactions*: We aim at developing a common framework that could model different types of interactions between any type of molecules that can be found inside cells. For that purpose, we consider extending current methodologies used to represent metabolic networks to accommodate a broader range of molecular interactions.
- *Integration of Biochemical Systems*: Given that cellular processes are usually modeled as separate systems, we want to integrate them in a more natural way than current methodologies by having them modeled using a common framework.
- *Whole-Cell Scale Network Model of an Organism* To evaluate the capabilities of the so proposed framework, we want to apply it to real data aiming at obtaining a large scale biochemical network that can represent the landscape of molecular species and interactions of a given organism. To do so, we choose the *Mycoplasma genitalium* bacterium because of its relative simplicity and data availability.
- *Topological Analysis of a Whole-Cell Biochemical Network* The topological analysis of biological networks can unveil emergent properties of the respective systems they represent that would otherwise be unnoticed by analyzing separated parts. Similarly, we want to investigate the topology of a biochemical network of integrated cellular processes and compare it with current literature about separated processes.
- *In Silico Experiment Simulation* Gene knockout is an experimental technique often used to evaluate gene function and associated phenotypes of the organism.

It consists of removing or inactivating a gene thereby inhibiting its actions in the cell. We want to assess whether a whole-cell biochemical network of an organism constitutes a useful source to perform such experiments *in silico*. To do so, we perform cascading failure analysis for the removal of gene nodes in the network.

- *Network-Based Simulation of Biochemical Systems*: Simulate different cellular processes usually implicate in using different simulation methodologies. Given that one of the goals of this project is to develop a homogeneous integrative model of biochemical systems, here we aim at developing a network-based simulation method that can estimate the dynamics. We also consider important characteristics such as the discrete representation of molecular counts instead of concentrations and stochasticity. Given the complexity of simulating a whole-organism, here we test the proposed methodology using smaller theoretical models and some real subsystems.

1.3 MAIN CONTRIBUTIONS

We believe that the work developed in this thesis provided important contributions to the scientific community, with one of them being published in a peer-reviewed scientific journal [28]. The main contributions of this work are listed as follows:

- *Biochemical Modeling Framework*: We developed a framework to model and integrate biochemical reactions of several different cellular processes. It is an extension of the current methodology applied for metabolic networks also drawing concepts from rule-based modeling.
- *Mycoplasma genitalium Whole-Cell Biochemical Network*: Using the proposed framework, we modeled a whole-cell biochemical network of the *Mycoplasma genitalium* organism as a case study. The network comprises 15 cellular processes that were integrated into a single-component network. Topological analysis of the network showed agreement with the literature. By performing *in silico* gene knockouts in the model we could predict with a high accuracy essential genes for the organism comparing with experimental data [28].
- *Constraint-Based Simulation Algorithm*: A discrete-stochastic simulation algorithm for biochemical systems was proposed using concepts of constraint-based modeling. The algorithm, hence called Constraint-Based Simulation Algorithm (CBSA), could provide simulations very similar to the ones obtained using the SSA for some theoretical models while displaying superior computational efficiency. This algorithm was also designed to be computed using General-Purpose Graphics Computing Units (GP-GPUs) which arguably yield the current highest computational power available for some parallel algorithms [29].

1.4 THESIS ORGANIZATION

The two chapters that follow this introduction will describe some basic concepts involving the representation of chemical and biochemical interactions by means of

reaction equations and networks. In Chapter 4 we propose a modeling framework to homogenously represent cellular processes and its application in a case study is presented in Chapter 5. The next chapter discusses some approaches to simulate biochemical systems modeled as biochemical networks. There is also the proposal of a new simulation algorithm called Constraint-Based Simulation Algorithm (CBSA) in Chapter 7. Applications of the CBSA on theoretical and real biochemical systems are shown in Chapters 9 and 10 respectively. The last chapter contains some concluding remarks about this work as well as suggestions for further developments.

It is important to notice that Chapters 4 and 5 contain pieces of text from [28]. Chapters 6,7,9, and 10 contains pieces of text from [29]. All the texts used in this document were written by the author of this thesis regardless of the source.

2

SYSTEMS OF CHEMICAL EQUATIONS

In this chapter, we will introduce the basic concepts regarding representation of chemical interactions through chemical equations, the mathematical description of reaction velocities, and also introduce the concept of biochemical reactions.

2.1 CHEMICAL EQUATIONS

Chemical interactions can be expressed by means of chemical equations. They are symbolic representations of the relationship between reactants and products [30]. For example, equation {2.2} describes the methane combustion reaction with oxygen resulting in carbon dioxide and two water molecules.

$$CH_4 + 2\,O_2 \longrightarrow CO_2 + 2\,H_2O \qquad \{2.1\}$$

On the left side, the reactants are indicated with a plus symbol between them. The products are indicated on the right side. The number of each molecular species required in the reaction, also called stoichiometry, is indicated before each molecular species. In this particular reaction, two O_2 molecules are required and two water molecules are produced. For molecules that have their stoichiometry equals one, the number omitted. The arrow in the middle indicates that the reaction occurs in the direction pointed.

One important characteristic of chemical equations is that the mass-balance between reactants and products must always be respected. In other words, the sum of the masses of the reactants must be equal to the summed masses of the products. It is related to the conservation of the number of atoms. In the case of {2.2}, the one carbon, four oxygen, and four hydrogen from the reactants are conserved in the products but bound differently.

Some reactions can occur in both directions, called reversible reactions. They are usually represented using double harpoons, one to each direction. It is the case of the following reaction:

$$CH_4 + H_2O \rightleftharpoons CO + 3\,H_2 \qquad \{2.2\}$$

. This reaction could be also represented as two different reactions, each one representing one direction. When representing more than one reaction, we can say we have a chemical system such as represented in {2.3}.

$$\begin{aligned} CH_4 + H_2O &\longrightarrow CO + 3\,H_2 \\ CO + 3\,H_2 &\longrightarrow CH_4 + H_2O \end{aligned} \qquad \{2.3\}$$

Other different reactions can also be represented together, increasing the size of the chemical system indefinitely. For example, the following chemical system consists of two reversible chemical reactions forming a system of four reactions.

$$\begin{aligned} CH_4 + H_2O &\longrightarrow CO + 3\,H_2 \\ CO + 3\,H_2 &\longrightarrow CH_4 + H_2O \\ CO + H_2O &\longrightarrow CO_2 + H_2 \\ CO_2 + H_2 &\longrightarrow CO + H_2O \end{aligned} \tag{2.4}$$

2.2 REACTION RATES

Reactions can occur at different velocities depending on the environmental conditions. The first determinant aspect of reaction velocities is the concentration of their reactants. Usually, the higher the concentration of reactants, the faster is the reaction's velocity (or reaction's rate) due to the higher probability of collision between them in the environment. The second aspect is related to the likelihood that the collisions between the reactants will result in a reaction, usually indicated as a *rate constant* k [31]. Although it is called a constant, it is actually a function of some physical parameters such as temperature. This relationship of the rate constant and the temperature is usually described by the Arrhenius equation [32]. In conditions where the temperature is guaranteed to be constant, the value of k can mostly be considered constant as well.

To illustrate the relationship between rate constants and concentration of reactants, let us consider the following chemical reaction system:

$$aA + bB \underset{k_{-1}}{\overset{k}{\rightleftharpoons}} cC \tag{2.5}$$

where a molecules A react with b molecules B resulting into c molecules C with a rate constant k and C can be converted back into A and B molecules with a backwards rate constant k_{-1}. The mathematical relation that usually describes velocities of such reactions of the forward (v_f) and backward (v_b) direction can be written as follows:

$$\begin{aligned} v_f &= k[A]^a[B]^b \\ v_b &= k_{-1}[C]^c \end{aligned} \tag{2.1}$$

being k and k_{-1} in such units that the resulting units of v_f and v_b will be given in mols per liter per second (mol/(Ls)). From this equation, we can derive the variation of the concentrations in time for each molecule as the following system of differential equations:

$$-\frac{d[A]}{dt} = -\frac{d[B]}{dt} = \frac{d[C]}{dt} = v_f - v_b \tag{2.2}$$

. The same reaction rates can be written in terms of absolute number of molecules such as:

$$-\frac{dA}{dt} = -\frac{dB}{dt} = \frac{dC}{dt} = k\binom{A}{a}\binom{B}{b} - k_{-1}\binom{C}{c} \tag{2.3}$$

with appropriate units for k_1 and k_{-1}.

2.3 ENZYMATIC REACTIONS

The most common reactions found inside cells are those that are catalyzed by proteins called enzymes [2]. These reactions are also known as enzymatic or biochemical reactions for having the enzyme as the main player.

Enzymes are proteins capable of catalyzing very specific reactions in the cell by binding to their substrates in a particular site of their structures [2]. Despite the typical high specificity of the enzymes, there is a wide range of these molecules aiming at catalyzing several different reactions, such as combination, modification, and lysis of small molecules as well as catalyzing chemical modifications in other macromolecules, such as other proteins, RNAs, and DNAs. Once bound to the substrates, the enzyme reduces the energy required for the reaction to happen, further releasing the products. This mechanism can be described by the following chemical equations:

$$\mathrm{E} + \mathrm{S} \underset{k_2}{\overset{k_1}{\rightleftharpoons}} \mathrm{ES} \xrightarrow{k_3} \mathrm{E} + \mathrm{P} \tag{2.6}$$

where E is the enzyme, S is the substrate, ES is the enzyme-substrate complex, and P is the product. This mechanism can also be simply described as follows:

$$\mathrm{S} \xrightarrow{\mathrm{E}} \mathrm{P} \tag{2.7}$$

indicating that S is transformed into P by the enzyme E. Although the velocity of an enzymatic reaction could be estimated from the underlying mechanism described in {2.6}, the rate constants k_1 and k_2 are not easily measured. Instead, the most used approach to mathematically model the kinetics of enzymatic reactions is by using the Michaelis-Menten equation [2]. This equation is capable of correctly describing most of enzymatic reactions in the form as that represented in {2.7}. Thus, the velocity of an enzymatic reaction can be written as:

$$v = -\frac{d[S]}{dt} = \frac{d[P]}{dt} = \frac{k_{cat}[E][S]}{k_M + [S]} \tag{2.4}$$

where k_{cat} is the turnover capability of the enzyme (related to the k_3 in {2.6}) and k_M is known as the Michaelis constant. Given that $k_{cat}[E]$ is also referred as the maximum velocity of the reaction (v_{max}), k_M is numerically equal to the concentration of S when $v = v_{max}/2$. This constant k_M is much easier to be obtained experimentally than k_1 and k_2 from {2.6}.

3

COMPLEX NETWORKS AND BIOLOGY

In this chapter, we will introduce the basic concepts about how to represent natural systems using graphs, also called networks or complex networks, and their topological study. Particularly, we will explore the use of networks to model biomolecular interactions. It will be also introduced concepts about undirected, directed, and bipartite graphs, using examples of applications in protein interactions, genetic circuits, and metabolic networks respectively.

3.1 GRAPHS, NETWORKS, AND COMPLEX NETWORKS

Natural (and unnatural) systems can be studied by identifying their composing parts and how they interact. One approach to represent such systems is by modeling them using networks [33]. We can say that each part of the system is a node and then link the nodes according to the interactions between the parts they represent. For example, one could model a social system by representing people as nodes and connecting them following a given social relationship, such as friendships. The resulting network could be understood as a representation of friendship connections within a group of people. Such an approach can be extended to any system that can be divided into discrete parts and relationships.

In discrete mathematics, a set of vertices and edges connecting the vertices is called a graph. They do not necessarily need to represent something, but the study of the different structures that can be obtained using such a representation and their associated mathematical properties provides a mathematical foundation, called *graph theory*, that can be applied to many real problems [34]. Essentially, "graphs" and "networks" are synonyms. However, the latter is usually associated with representations of real systems while the former is more frequently used to refer to pure mathematical entities. Also, the terms "vertices" and "edges" are used in the context of graphs while the terms "nodes" and "links" are adopted when regarding networks.

When studying graphs or networks, one of the most important features to be analyzed is their topology [33]. It is related to the patterns of connections between the nodes. When representing real systems, the topology of a network can unveil many emergent properties of the system that would otherwise be impossible, or very difficult, to assess considering only the system's parts.

There are many ways to investigate the topology of a network. The most common and often effective way is to analyze the *degree distribution* of the nodes [33, 35]. The degree of a node is simply the number of connections it has with other nodes. Therefore, the distribution of nodes degree is the relative frequency (or probability) of each degree in the network. In Figure 3.1, three different networks are depicted with their associated degree distribution. The first one on the left has a regular pattern of connections, with almost all nodes presenting the same degree. The second one in the middle has its nodes connected in a uniformly random way [36].

The probability distribution $p(k)$ of finding a node in the network with a degree k is represented in the respective plot resembling a Gaussian distribution .

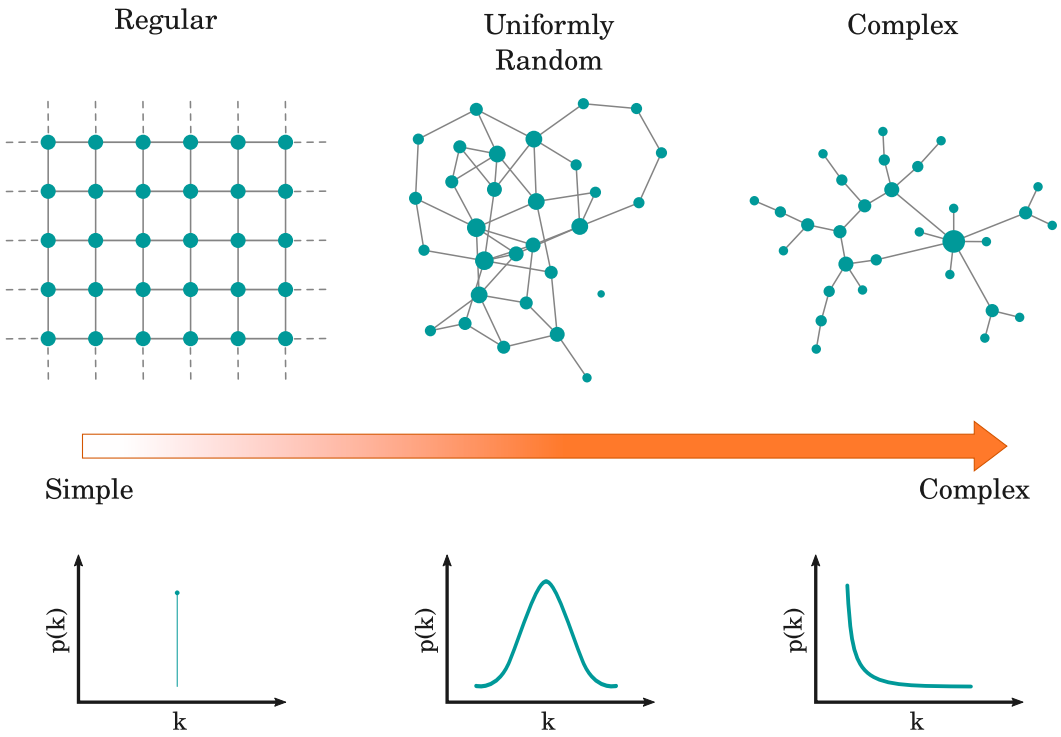

Figure 3.1: Regular, random, and complex network topologies and their respective node degree distributions.

The two networks mentioned in the last paragraph, having random or regular connection patterns, are considered simple networks. That is because the connecting pattern of any node is highly similar and predictable. On the other hand, some networks may present connecting patterns that are not trivial, being called *complex networks*. This is the case of the network depicted in the right on Figure 3.1 and is also one of the most common types of topology found when modeling real systems. The distribution of nodes' degree often follows a power-law-like shape, having few highly connected nodes, though typically more than in a uniformly random network, and several less connected ones [37]. Other characteristics, such as clusters, can also appear in complex networks as well as multi-modal degree distributions [38].

When modeling biological systems using networks, the connecting patterns obtained are usually non-trivial [39]. It has been suggested that network models of many different biological realms, from ecological to molecular systems, share similar topologies usually following power-law distributions of node degrees [40, 41]. One possible explanation about why this kind of topology is so ubiquitous is that the relationships between biological entities are shaped by evolution leading to robust structures that allow life to endure [42–44].

One important emergent characteristic of networks with power-law-like distributions of nodes degrees is that they are highly resistant to random failures [37]. They have many nodes with few connections and few nodes highly connected. If we randomly choose one node to fail in the system, there is a high probability that

it will be a node with few connections and will perhaps not having much impact on the whole structure. If we transfer this concept to a biological system, for example to a molecular system, random mutations occur in the DNA all the time possibly causing the malfunctioning of a gene product. If the molecular interactions follow such a topology, most of the genes will have fewer connections in the cell thus having a lower probability of causing significant damage to the whole system. Nevertheless, the few highly connected entities can serve as week spots. A malfunction of such entities can cause huge impacts in the molecular system, sometimes leading to diseases or death [45].

3.2 UNDIRECTED GRAPHS AND PROTEIN-PROTEIN INTERACTION NETWORKS

A graph G can be mathematically represented by a set of vertices and edges $G = v, e$ where $v = v_1, v_2, ..., v_n$ and $e = (v_a, v_b), ..., (v_c, v_d)$ with $a, b, c, d \in [1, n]$. Considering $(v_a, v_b) \equiv (v_b, v_a)$, we have an undirected network. It means that a relationship between two nodes is bilateral. One example of a system for which such type of network can be a suitable representation is in protein-protein interactions within a cell, such as complex formation [46]. If a protein A binds a protein B, it is the same to say that a protein B binds a protein A.

Another characteristic that can be incorporated into a network representation is a value associated with each edge, called weight, thus being mathematically expressed as $e = (v_a, v_b, w)$ [33]. In protein-protein interaction (PPI) networks, this weight w can for example represent the likelihood of that interaction to exist based on several experimental data sources. In Figure 3.2 we show a graphical representation of a PPI network of genes related to the p53 human gene as an undirected network with weights associated with the edges represented as the edge thickness.

The P53 gene is considered a tumor suppressor involved in several processes such as cell cycle regulation, apoptosis, and genomic stability [48–50]. Some mutations in its sequence can lead to damage to its protein product function potentially leading to diseases such as cancer [51]. In the context of PPI networks, it is usually a highly connected node [52] as observed in Figure 3.2. Considering that PPIs usually have degree distributions in a power-law form, the P53 can be considered a week spot in the network structure evidencing its role in diseases [41].

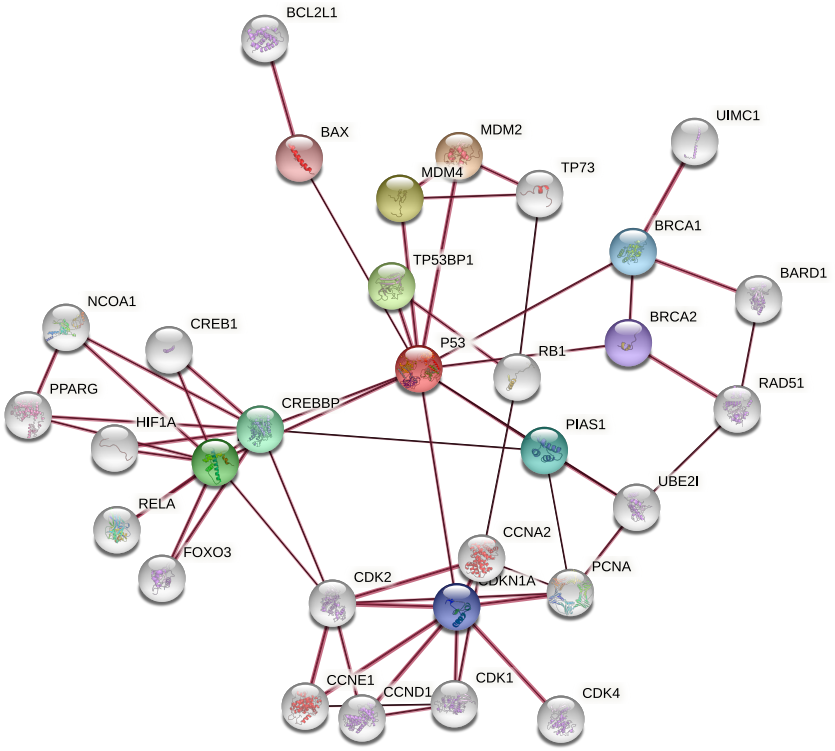

Figure 3.2: An example of an undirected weighted network where nodes and edges represent proteins and their physical associations respectively. The thickness of each edge is proportional to the confidence of that interaction obtained experimentally. The images inside each node display the tridimensional structure of the protein. This network was obtained from STRING [47] by searching for the human protein P53 and close interactions.

3.3 DIRECTED GRAPHS AND SIGNALING NETWORKS

There are cases in which relationships between two entities are not bilateral. In other words, we can mathematically express in a network that $(v_a, v_b) \neq (v_b, v_a)$. Networks with this kind of property are called directed networks and edges indicate which direction the interaction occurs [33]. For example, protein A can catalyze a chemical modification in protein B, but protein B can not catalyze a modification in A.

Directed networks are often used to model signaling processes inside cells [55–57]. Proteins can suffer modifications due to environmental changes such as temperature, osmotic stress, and the presence or absence of certain molecules. Once modified, they can trigger modifications in other proteins leading to a cascade of events that can ultimately lead to changes in gene expression as a response to that change in the environment. These signaling cascades often have determined directions to occur not being able to reverse the process by the same pathway of molecular modifications. Figure 3.3 illustrates a signaling network where the protein p53 is inserted (highlight in orange). The edges have arrows indicating the direction of the interaction. Some edges have perpendicular bars at the end indicating that this is an inhibitory interaction. Arrows and bars can indicate for example a positive or negative value respectively of the weight associated with the edge.

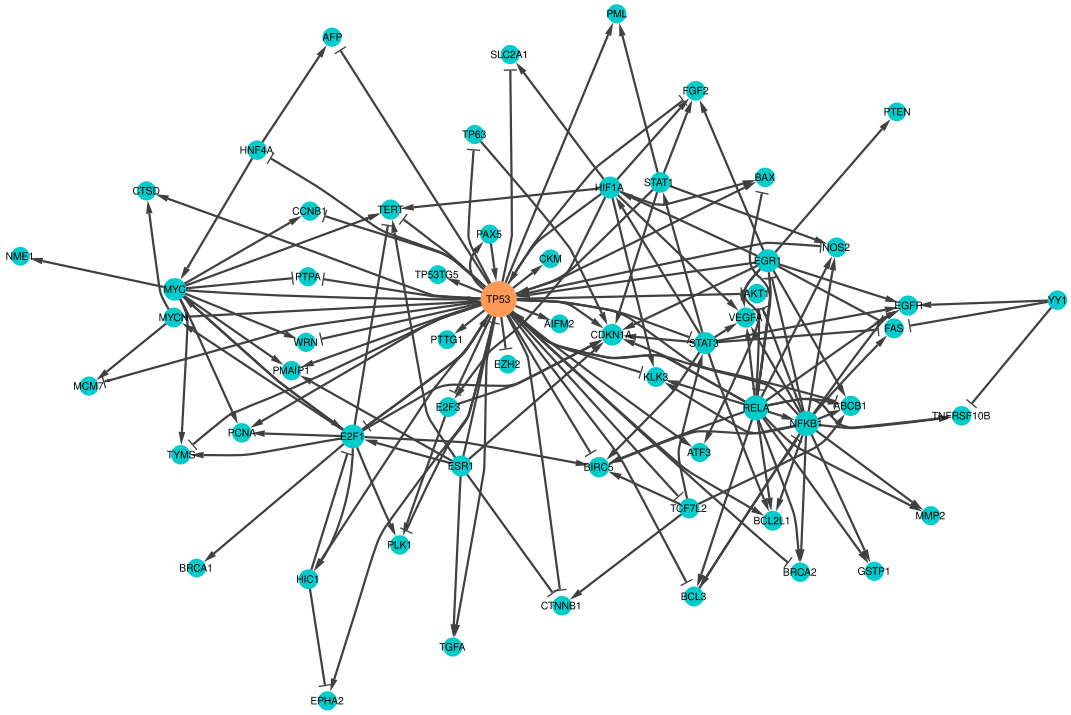

Figure 3.3: A signaling network where the p53 protein is inserted. It is represented by a directed network where arrows and bars indicate the direction of the interaction as well as if it is a stimulatory or inhibitory interaction respectively. This network was obtained using the OmniPath [53] plugin in Cytoscape 3.8.2 [54] by searching for the p53 protein. Node sizes are proportional to their degree.

3.4 BIPARTITE GRAPHS AND METABOLIC NETWORKS

Bipartite networks are networks with a particular structure where nodes can be divided into two groups that do not have connections between nodes of the same group [33]. They are commonly used to represent interactions between two different kinds of entities. It is the case of metabolic networks where we can have nodes representing metabolites and nodes representing reactions [40]. In this case, all the edges connect metabolites to reactions.

The relationships between metabolites and reactions are twofold: reactants and products. This can be represented by the directionality of the edges, becoming a bipartite-directed network. Figure 3.4 depicts a fraction of the central carbon metabolic network from *E. coli* with metabolites colored in blue and reactions in orange. Another characteristic that can be incorporated into this kind of network is the stoichiometry of the reactions as edge weights. For example, if a reaction has 2 ATP molecules as reactants, the edge going from the ATP node to the reaction node will have its weight equals to 2. In the same sense, if the reaction has 2 PI as products, the edge going from the reaction node to the PI node will have its weight equals to 2 as well.

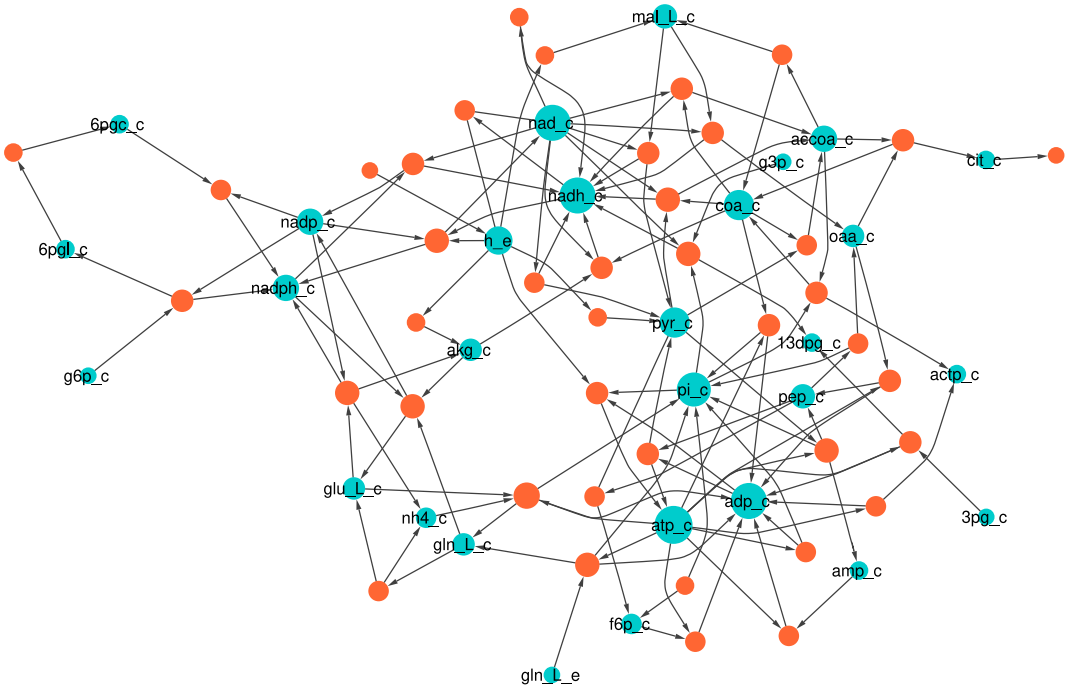

Figure 3.4: A fraction of the central carbon metabolic network from *E. coli* obtained at BiGG Models [11] and visualized using Cytoscape 3.8.2 [54]. Node sizes are proportional to their degree.

4

BIOCHEMICAL NETWORKS AND INTEGRATIVE MODELING

All cellular processes can be understood by means of their underlying biochemical reactions. By far, the most detailed cellular process in terms of specific biochemical reactions is the metabolism [58]. For many organisms, we already have very detailed and possibly complete metabolic maps [11]. Even though, metabolism is just a small set of reactions complementing other cellular processes. While the interface between metabolism and protein-protein interactions has already been tackled in the literature [59], other cellular processes such as protein and RNA synthesis, DNA replication, protein and RNA degradation, and cell cycle, are rarely represented at the biochemical level. In this chapter, we propose a modeling framework that aims at enabling the incorporation of a wide range of molecular species and interaction types to accommodate cellular processes such as the ones mentioned above. The methodology described here was published in [28].

4.1 BIOCHEMICAL REACTION MODELING FRAMEWORK

To better explain our modeling approach, a simple graphical notation will be used to depict molecules and interactions in the form of a network. Thus, using network science nomenclatures, representations of molecules and reactions will be called as *nodes*. Relationships between molecules and reactions will be called as *edges*. Figure 4.1 shows the graphical notation adopted in this work.

Here, molecule nodes can represent any physical entity within a cell such as metabolites, DNA regions, proteins, and RNAs. Each different state of a molecule (e. g., the active and inactive form of a protein) will be represented by different molecule nodes. Similarly, molecules in different cellular compartment locations (e. g., intra and extracellular glucose) are also represented by distinct nodes.

We can represent any interaction between molecules by a reaction node. In addition to biochemical reactions, such as in metabolism, more complex interactions such as gene transcription, protein synthesis, transport, protein complex formation, chromosome replication, and cell division can be incorporated into the model.

Reactions can be regulated and we can explicitly link the "modifiers" to their respective reaction nodes. To do so, we use a distinct type of edge, called *modifier edge*. In Figure 4.1a, the modifier edge is drawn as a circle-ended line connecting molecule "Enz" to the reaction. This connection means that the Enzyme is needed so that the reaction can occur, but it is neither consumed nor produced in the reaction. For example, other molecules such as transcription factors, genes, and mRNAs can act as modifiers, since their concentration does not change in some reactions.

To illustrate the modeling of some molecular interactions, Figure 4.1 shows some use cases derived from the generalizations we adopted. Example (a) depicts a biochemical reaction where an enzyme combines Met1 and Met2 into Met3. In (b), a given protein interacts with a ligand (e.g., in an allosteric site) producing the

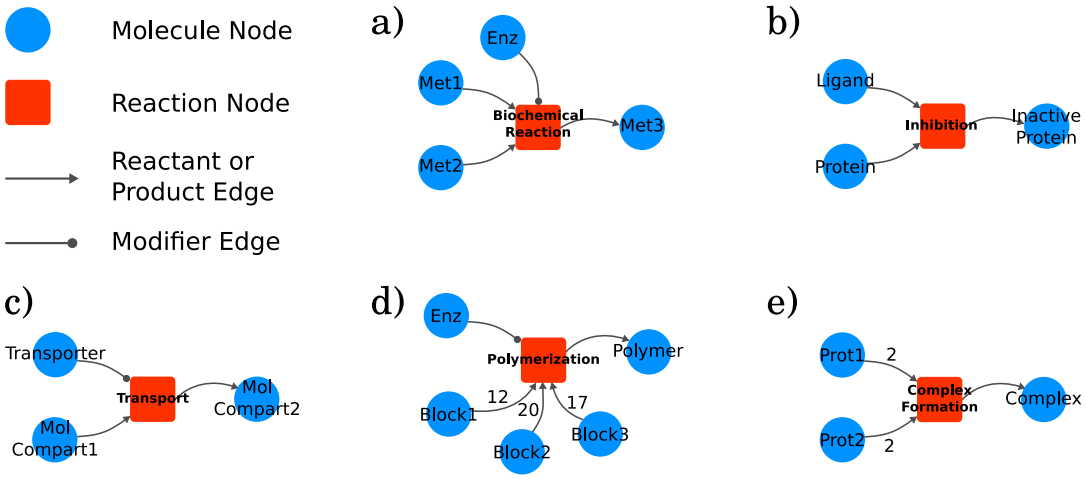

Figure 4.1: Graphical representation of the reaction modeling framework. Molecule nodes are represented by blue circles. Red squares represent reaction nodes. Arrow-ended edges indicate reactant or product relationships depending on direction. Circle-ended edges indicate modifiers to the reaction. Some modeling examples of distinct kinds of biochemical interactions are depicted as follows. (a) A biochemical reaction $\mathrm{Met}_1 + \mathrm{Met}_2 \xrightarrow{\mathrm{Enz}} \mathrm{Met}_3$; (b) inhibition of a protein by a ligand; (c) molecular transport from compartment 1 to compartment 2 by a transporter; (d) a polymerization reaction; (e) a tetrameric protein complex formation. All detailed network visualizations in this and further figures were generated using Cytoscape [54] unless stated otherwise.

protein's inhibited form. Transport through a cellular membrane can be approached as in example (c), where a given molecule is carried from extra to the intracellular environment by a transporter protein. Example (d) illustrates a polymerization process catalyzed by an enzyme where different stoichiometries of three basic building blocks are combined into a single polymer, such as in a protein synthesis reaction catalyzed by a ribosome. Protein complexation of four subunits (e.g., α and β hemoglobin subunits) is represented in example (e).

4.2 CELLULAR PROCESS MODELING AND INTEGRATION

Certain cellular processes make use of the same molecular machinery to produce different outputs given different inputs. It is the case of the synthesis of macromolecules such as proteins, RNAs, and DNAs. For instance, the synthesis of proteins involves a set of reactions that repeats for each mRNA given as an input. Most of the molecules involved in the process are the same for different mRNAs, sometimes only changing the stoichiometry of amino acids. For such processes, we propose the use of templates which are sets of reactions that can be replicated and adapted for different substrates.

In Figure 4.2, an example of a template for the transcription process is shown inside the red box. The template contains the reactions for the binding of an RNA Polymerase to a DNA region and the transition of the complex through the gene until completion of the transcription releasing an RNA molecule and the RNA Polymerase. The process is the same for the transcriptions of all genes, as the

template is able to be replicated only by changing the DNA region and the resulting RNA molecule. Particularities of each reaction, such as the need for transcription factors, can be incorporated as indicated by the dashed lines in the Figure 4.2. It is important to notice that the templates need to be made specifically to the subject organism and the modeling assumptions can vary from modeler to modeler.

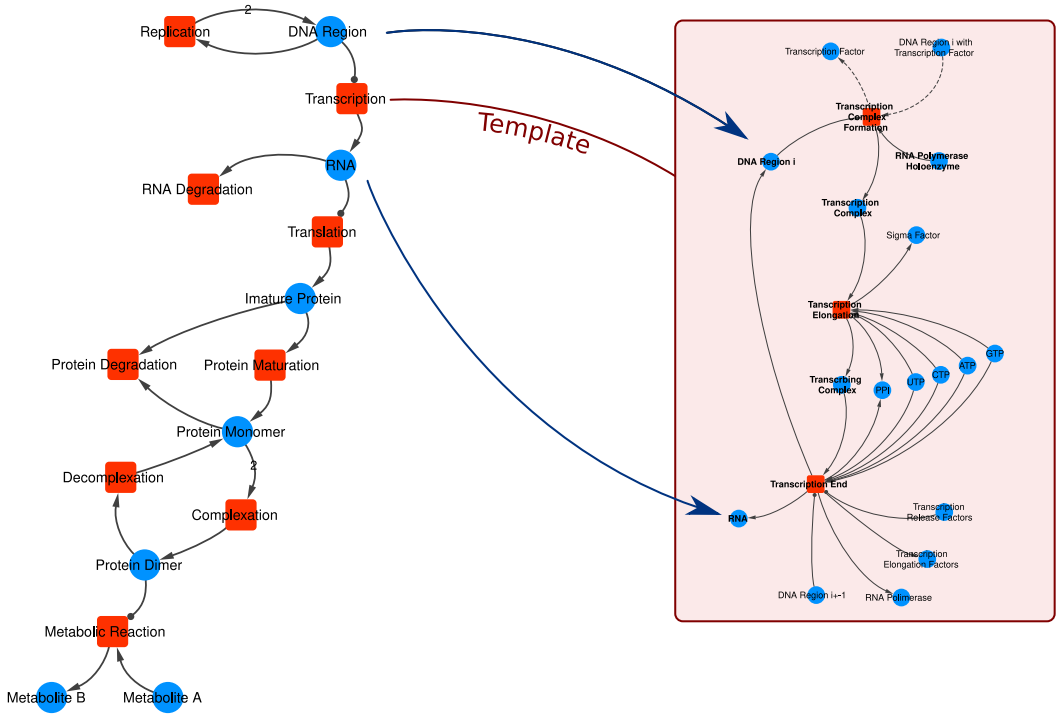

Figure 4.2: On the left, a simplified version of a full cycle of network building starting from a single metabolic reaction. For processes such as replication, transcription, translation, and degradation, templates are used to represent the process with adaptations for each specific case based on available data as shown on the right.

The integration of cellular processes modeled using the proposed framework should occur naturally if they share molecular species. As an example, a simplified representation of a full pathway from DNA to metabolite is depicted in Figure 4.2. Starting from the bottom, a homodimeric enzyme that catalyzes a given metabolic reaction has its biosynthesis pathway built upwards to the DNA level passing through protein complexation, protein maturation, protein synthesis, RNA synthesis, and DNA duplication. Degradation reactions for proteins and RNA are also shown.

5 MYCOPLASMA GENITALIUM CASE STUDY

In 2013, a public database called WholeCellKB was implemented aiming at gathering complete biological information about specific organisms [60]. The first organism deposited in this database was the pathogenic bacterium *Mycoplasma genitalium*. This organism yields the smallest genome known, with 580 kb and 525 genes. Because of its relative simplicity, *M. genitalium* has served as the model organism for breakthrough advances in synthetic biology [61] and whole-cell simulations [10]. The simplicity of this organism, allied to the structured data provided by the database, makes *M. genitalium* a particularly suitable model for the comprehensive integration of cell-scale biochemical interaction into a whole-cell biochemical network.

In this chapter, we use the *M. genitalium* organism as a case study to assess the capabilities of our proposed modeling framework. We use it to model the data available at WholeCellKB and produce a whole-cell biochemical network of the referred organism. Then, we characterize it topologically and perform *in silico* knockouts experiments and compare to experimental knockout data to validate the network. The methodology and results here described were published in [28].

5.1 NETWORK BUILDING PROCESS

To start building the biochemical network of the *M. genitalium* organism, we first queried all metabolic reactions from WholeCellKB and included them as reaction nodes in the model. Then, all metabolites which act as reactant or product, as well as enzymes, were incorporated in the network as molecule nodes and properly linked to their respective reactions. From this point, a recursive process starts by adding biosynthesis and degradation reactions for each molecule already in the network. For example, protein complexes are biosynthesized by macromolecular complexation reactions. Protein monomers are produced by translation and protein modification reactions. They also are degraded by proteolysis. For these newly incorporated reactions, the necessary molecules are added as molecule nodes and linked to them. This process repeats itself until all molecules have their respective biosynthesis and degradation reactions. At the end of this recursive process, it is expected the network to have all reactions from metabolism to DNA Replication, passing through all the central dogma of molecular biology (Fig. 5.1).

On top of the so obtained network, we queried the WholeCellKB for reactions that are still not included in the model, such as redundant reactions, and added them. To finalize the network, we manually included the "Cell Division Reaction" to which all necessary proteic complexes, such as FtsZ Ring, Chromosome Segregation proteins, and the duplicated Chromosome, are linked as modifiers. A detailed description can be found in the Supplementary Information.

In order to guarantee the coherence of the reactions in the network, we calculate the mass-balance for all reactions. Following the principle of mass conservation, the difference between the mass of reactants and products, weighted by their respective

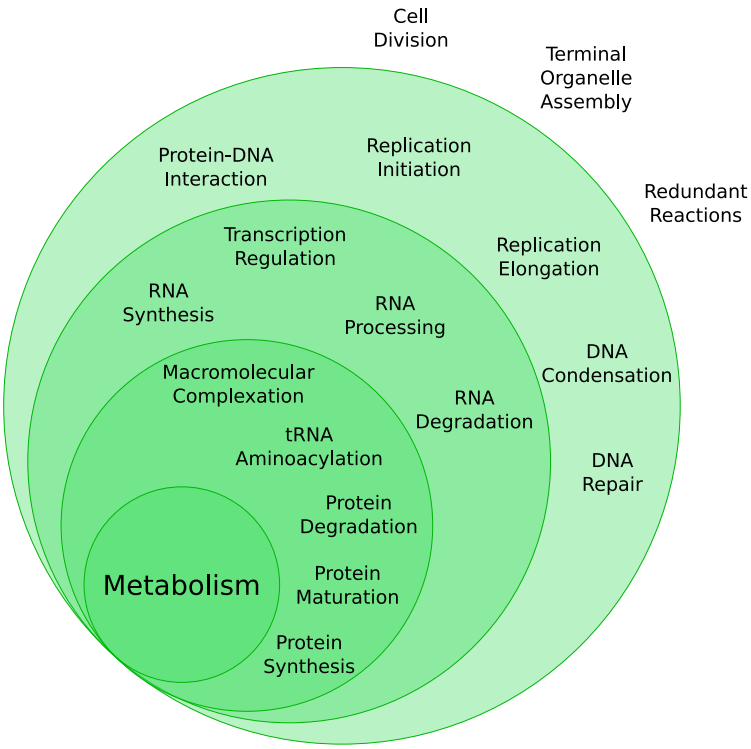

Figure 5.1: The hierarchical construction of the *Mycoplasma genitalium*'s whole-cell biochemical network. Starting from the metabolism several cycles of introducing necessary molecules to the reaction and their synthesis/degradation reactions. When the network construction reach the DNA level, the chromosome is divided in regions in order to better represent locus-specific protein interactions.

stoichiometry, should be zero. Although this approach can assure the mass-balance, literature evidence is still needed to ensure their correctness.

5.2 CHROMOSOME REPRESENTATION

Despite being the smallest known chromosome, *M. genitalium*'s still a large and lengthy circular molecule having 580kb. In order to better represent locus-specific protein interactions, we divided the chromosome into regions, each one being represented by a molecule node.

We used as reference the *M. genitalium* G37 genome [62], available at the NCBI database (NC_000908.2). In Figure 5.2 we can observe the circular chromosome representation as well as the genes distributed along with it. The transcription units (TUs) are the regions that are transcribed in RNA. One TU can encompass one or more genes, the last case also being called "Polycistronic RNAs". The RNAs from TUs with more than one gene can be further cleaved into separated RNA molecules, which is the case of tRNAs, or left as one molecule. In any case, each RNA molecule, polycistronic or not, is represented by a single molecule node.

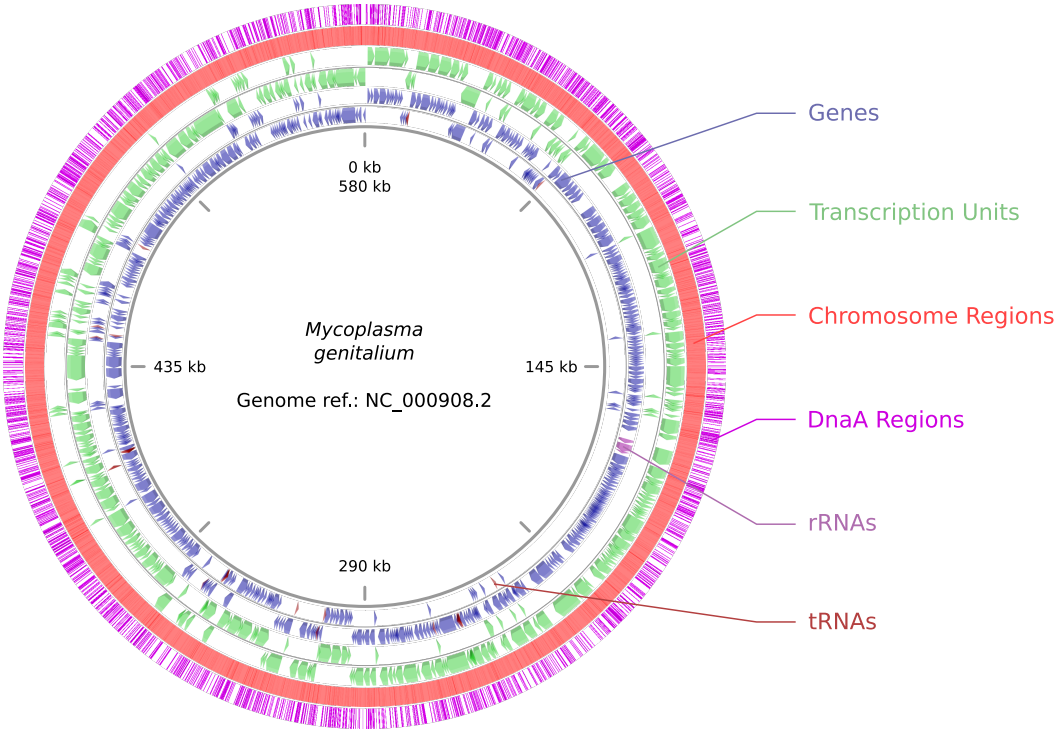

Figure 5.2: *M. genitalium* chromosome representation. Figure was partially generated using CGView [63]

Although intuitive approaches would be to divide the chromosome regions according to the TUs or even to split into regions of the same length, the more granular the division, more details can be incorporated in the model. The DnaA protein interacts with small $\sim$ 8 nucleotides long sequences repeated all over the chromosome. Figure 5.2 depicts the distribution of DnaA binding sites annotated in the WholeCellKB. The binding and polymerization of this protein at specific nucleotide sequences in the chromosome are the main mechanisms to control cellular replication and the binding sites present a more granular division of the chromosome. Thus, we adopted the DnaA binding sites as division points to define the chromosome regions, with the addition of the replication origin and terminus

sites. More specifically, each DnaA biding site is defined as a chromosome region, and each region between DnaA binding sites is also defined as a chromosome region.

5.3 MODELING CANONICAL PROCESSES

Although some processes, such as metabolism, have a straightforward modeling transition from WholeCellKB to the proposed framework, other processes require more attention. For instance, it is not usual to describe chromosome replication, gene transcription, protein synthesis, and some other processes as networks. Thus, we manually created templates based on literature for these processes. Particularities in the synthesis process of individual proteins, RNAs, and DNA are incorporated accordingly to data availability in WholeCellKB. The naming references in the computational model for the templates described in the following can be found in the Supplementary Information from [28].

Chromosome Replication

The chromosome's replication starts when the DnaA protein polymerizes in five specific DnaA binding regions near the replication origin and recruits all necessary molecular machinery to replicate the DNA. It is the formation of the two Replisomes at the origin of replication in the chromosome, which we call the Replication Complexes.

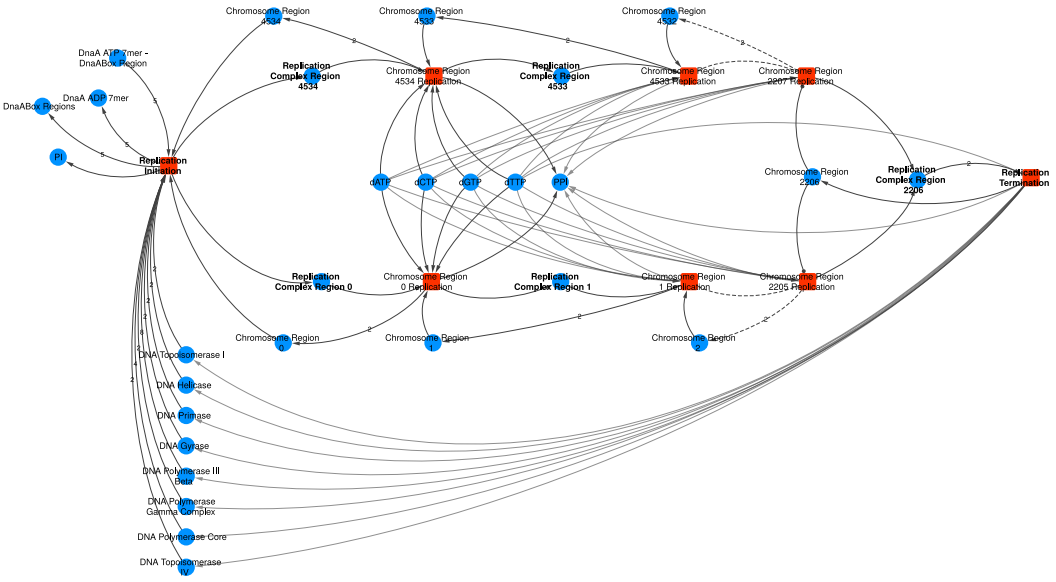

Figure 5.3: The template for Replication Initiation and Replication Elongation processes. The network depicted shows three steps of Replication Elongation going both chromosome directions. The dashed lines mask the rest of the process and go straight to the final reaction, where the terminus region (2206) is replicated and the Replisomes released. The actual molecules' IDs in the network are listed in the Supplementary Information. Although some stoichiometries are shown, most of them are hidden for better visualization.

Given that the chromosome is divided into regions, the formation of the Replication Complex also takes as a reactant a chromosome region. The two Replisomes, bound to the Chromosome Regions 0 and 4534, undergo their respective replication reactions where the free deoxy-nucleotides are consumed according to the regions' sequence. Each replication reaction produces two copies of the current Chromosome Region and consumes the next region, making the Replisome move through the DNA molecule (Figure 5.3). The two Replication Complexes move in opposite directions until they reach the replication terminus region, where the replication completes and the Replisomes' subunits are released. The collision of the Replisome with other Protein-DNA complexes, such as DnaA and RNA Polymerases, are handled separately.

RNA Synthesis

The RNA Synthesis is the process in which an RNA Polymerase makes an RNA molecule based on a Transcription Unit (TU), a region of the chromosome that may contain one or more genes. Given that a TU can be fragmented in several Chromosome Regions, as observed in Figure 5.2, the transcription process follows a similar approach to the Chromosome Replication, once the Transcription Complex moves through the DNA and its template is shown in Figure 5.4.

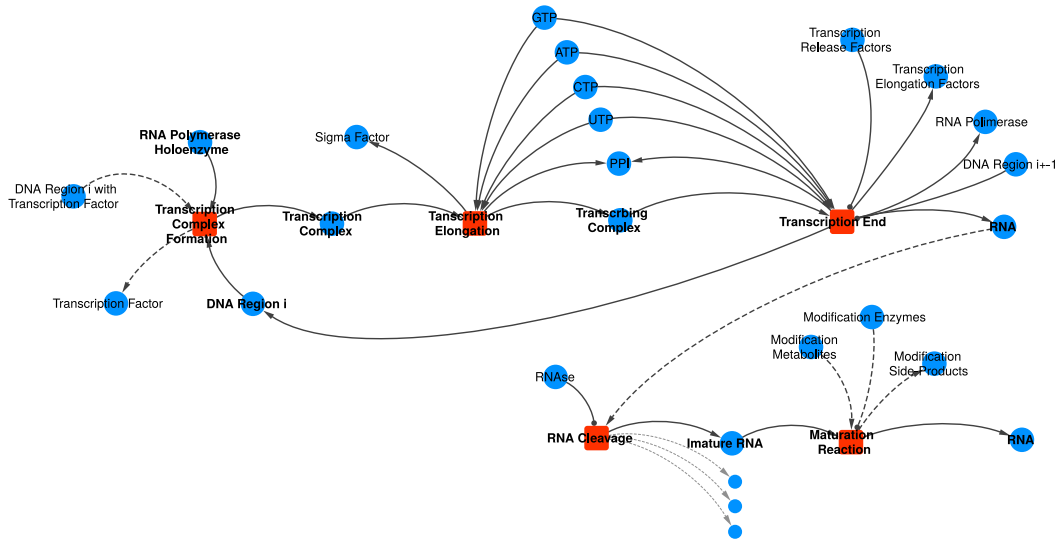

Figure 5.4: The template for the Translation process. Dashed lines indicate relationships that are not always required. For example, not all RNAs goes through RNA Cleavage and Maturation reactions, such as tRNAs. Most of the RNA modifications available in WholeCellKB were not included in the model due to systemic inconsistencies found in the database regarding the positions of the modifications.

The binding of the RNA Polymerase Holoenzyme to the beginning of a TU in the chromosome might require a Transcription Factor already bound to the respective Chromosome Region, as indicated with dashed links to the "Transcription Complex Formation" reaction in Figure 5.4. Then, given that the TU begins at the Chromosome Region *i*, the Transcription Complex, the example in Figure 5.4 depicts

a TU divided into only two Chromosome Regions, being the second one at the position $i+1$ or $i-1$ depending on which strand of the DNA the TU is found. TUs that are polycistronic and need to be cleaved to produce the individual functional molecules, the so transcribed RNA goes through further cleavage and maturation reactions.

Transcription Stall Reactions

A transcription reaction can be interrupted for several reasons. One of them is the collision with other molecules in the same region of a DNA strand. Here we modeled the stall reaction for transcribing complexes when a replication complex is in the next chromosome region. Once the transcription reaction can be interrupted at many chromosome regions, one incomplete RNA molecule is created for each reaction. The name of the molecule carries its sequence.

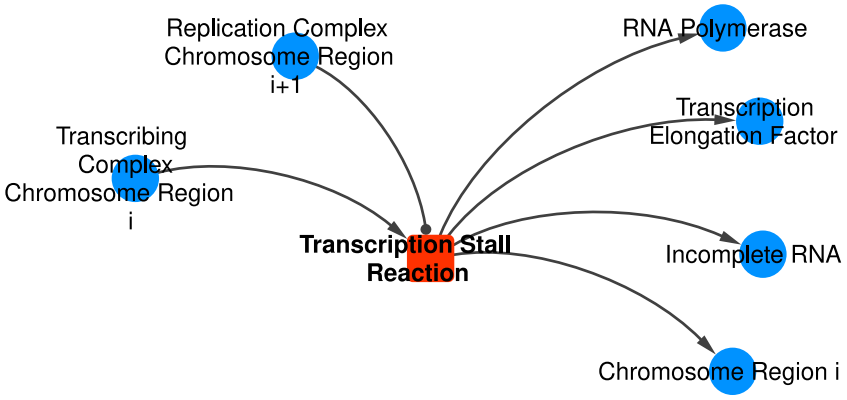

Figure 5.5: Transcription Stall Template.

RNA Degradation Reactions

The RNA degradation reaction template is depicted in Figure 5.6. The Peptidyl-tRNA Hydrolase is needed only in the case of aminoacylated tRNAs. Modifications in RNAs were not taken into account due to inconsistencies in WholeCellKB.

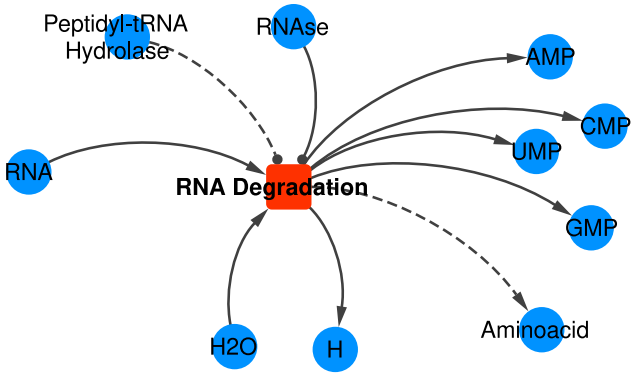

Figure 5.6: RNA Degradation Template.

Protein Synthesis

The template for the translation process is shown in Figure 5.7, including the translation complex formation, translation elongation, and protein maturation.

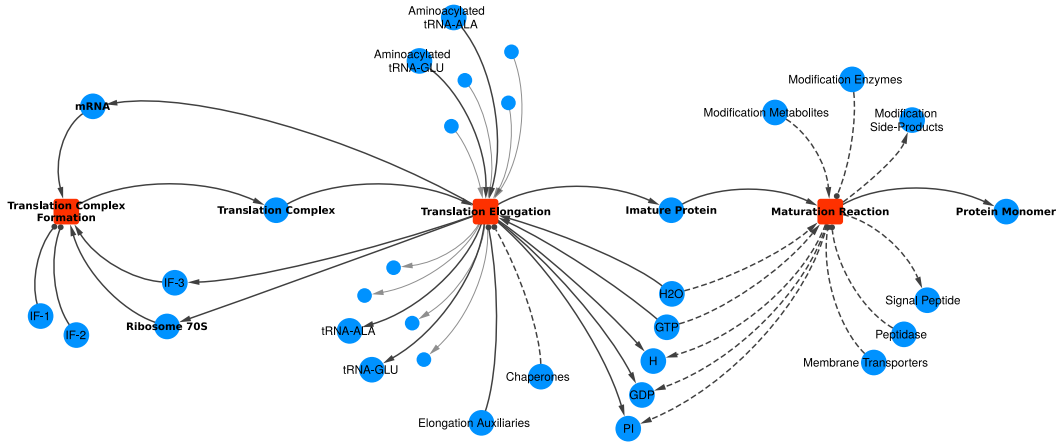

Figure 5.7: The template for Translation and Protein Maturation processes. Dashed lines indicate relationships that are not always required. For example, not all proteins require Chaperones. Membrane Transporters, Peptidase, and the production of the Signal Peptide are only required for secreted proteins. Only two of the twelve amino acids are shown to better visualization. The small blue circles illustrate the other amino acids and associated tRNAs.

The translation process starts with the complexation of the Ribosome 70S, Initiation Factor (IF) 3, and the mRNA with IF-1 and IF-2 as auxiliary molecules. Then, the translation complex proceeds to the elongation stage where aminoacylated tRNAs and energy molecules (GTPs) are consumed according to the mRNA sequence while the respective tRNAs without amino acids and the IF-3 are released. Elongation auxiliary proteins act as modifiers and chaperones might also be linked depending on the protein's annotations in WholeCellKB. Similarly, post-translational modifications can be incorporated in the Maturation Reaction, transforming an immature form of the protein into the functional one. Proteins that are secreted to the external environment have also linked to the Maturation Reaction the necessary membrane transporters, the peptidase to cleave the Signal Peptide, and the Signal Peptide itself.

Translation Stall Reactions

Just as transcription reactions, the translation process can be interrupted by several reasons too. However, when a transcription complex stalls, the incomplete protein needs to be tagged with a specific amino acid sequence in order to be rapidly degraded. Thus, this process is represented by two template reactions: the stall of the translation complex and the translation of the signal peptide. Once we do note represent intermediate molecules during the translation process, all stalled translation reactions will only produce the same incomplete peptide, which contains only the degradation signal sequence.

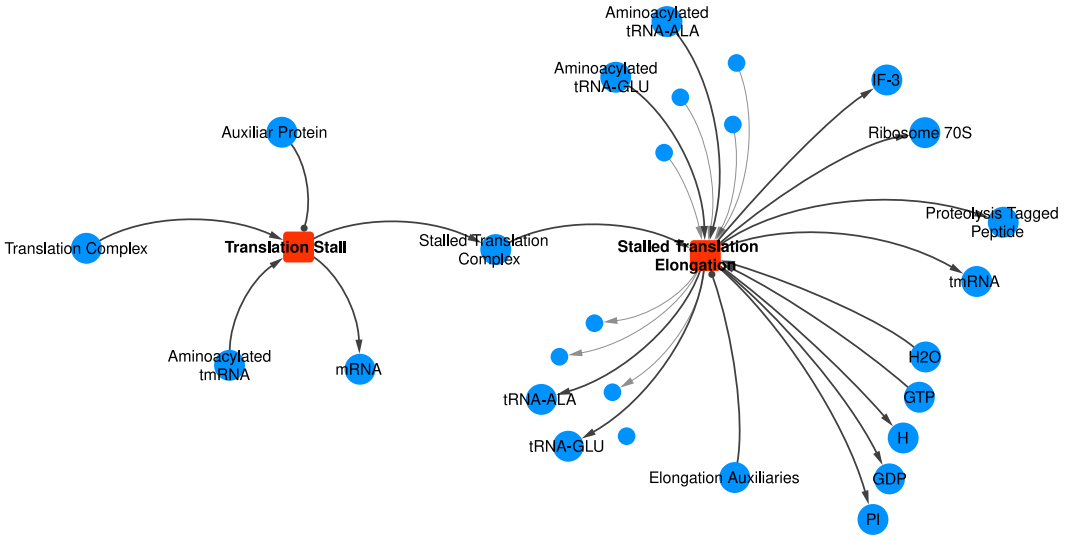

Figure 5.8: Translation Stall Template.

Protein Degradation Reactions

The proteins produced by the cell can be degraded in order to recycle amino acids, control proteins' concentration, remove defective proteins from the cytosol, and other reasons. Figure 5.9 shows the template for protein degradation reactions. According to the protein's location (cytosol or membrane), different proteases can be recruited for its degradation. Proteins tagged with the Proteolysis Peptide are degraded by the membrane protease.

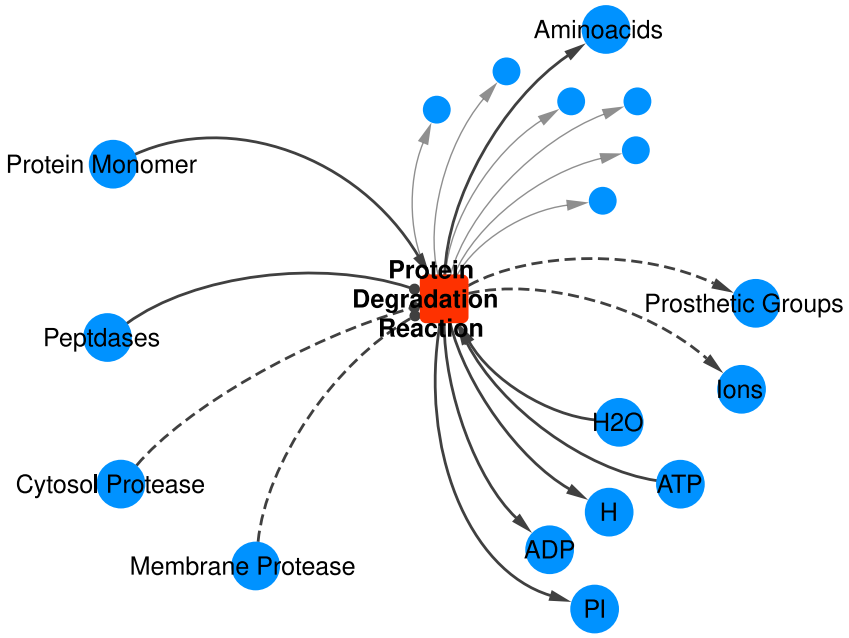

Figure 5.9: Protein Degradation Template.

Cell Division Reaction

The cell division is a biochemical and mechanical event involving several molecules and structures. In the *M. genitalium*'s whole-cell network it was modeled as a single reaction with all necessary molecules linked as modifiers. Figure 5.10 illustrates the reaction.

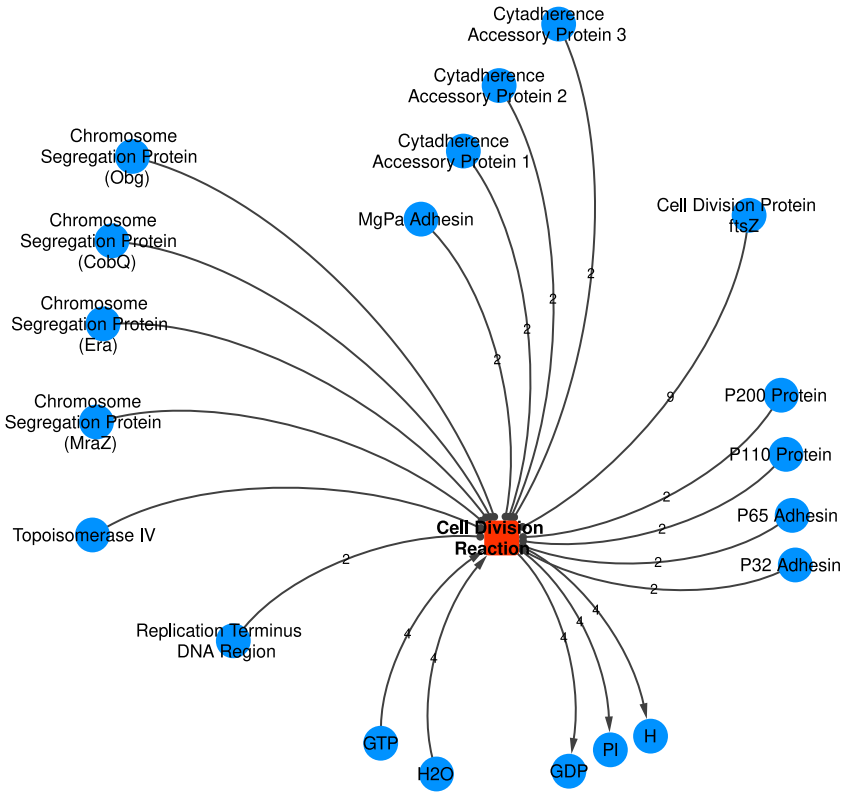

Figure 5.10: Cell Division Reaction illustration.

5.4 SOFTWARE STRUCTURE AND IMPLEMENTATION

The software called PiCell was developed to build the Whole-Cell Extended Biochemical Network of *Mycoplasma genitalium* but also being adaptable for other organisms. It is composed of three parts:

- Database Handler
- PiCell Core
- Network Constructor

that can be accessed by Python 3 scripts. The database handler is the interface between databases and the PiCell core. One handler should be implemented for each database to be used as a source of the model. The PiCell Core is responsible for organizing the data obtained from databases and create intermediate molecules and reactions in order to fulfill the central dogma of biology in the model. When all

necessary information os gathered in the PiCell Core, it can be exported as a single network model, with linked molecule and reaction nodes, following the framework proposed in this work. This model is then further submitted to validation and analyses. Figure 5.11 shows a schematic of the software implemented to build the *M. genitalium*'s network.

Database Handler

The necessary information for the model was acquired from the WholeCellKB through the WholeCellKB Handler, a piece of Python 3 code implemented specifically for this database. The data in the WholeCellKB database was available in several formats. The JSON format was chosen because of its easiness of access from Python. In addition to the JSON database file, the Handler can read two other files: one containing the database entries to be ignored, and another containing a name mapping to be applied in the database.

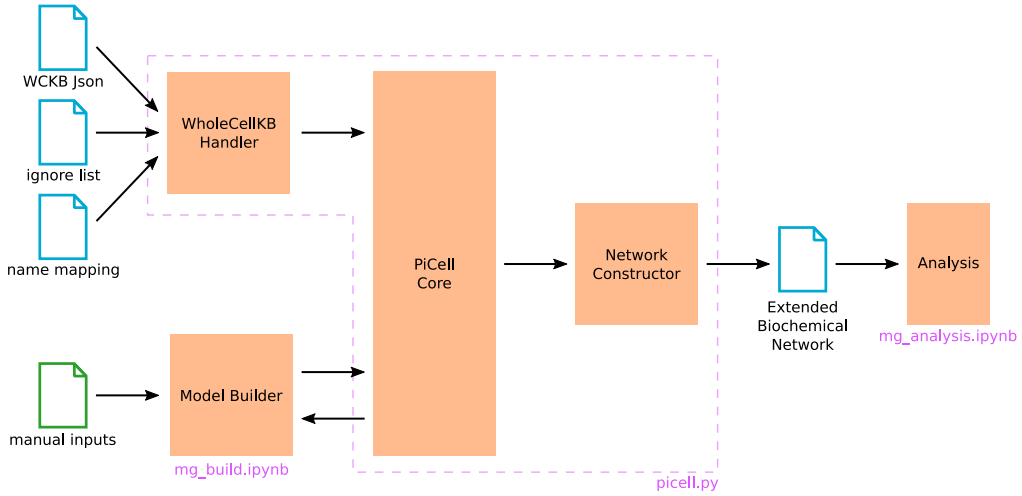

Figure 5.11: The schematic implementation of the PiCell, a software to build Whole-Cell Extended Biochemical Networks.

Model Builder

The control of the modeling is made through an IPython Notebook using the Jupyter interface[1]. Before acquiring the database's information, the model must be configured. Information about the canonical cellular processes must be provided in order to be constructed from the templates by the PiCell Core.

The genetic information about the organism must also be provided. In the case of *M. genitalium*, it was also available in the WholeCellKB. The chromosome sequence, chromosome features, genes, and transcription units are necessary to construct the canonical processes.

1 https://github.com/pauloburke/whole-cell-network

Molecules and reactions to be added in the model can be retrieved from the database or inserted manually. An example of the latter is the cell division reaction and its structure and components can be found in Figure 5.10. Reactions, such as metabolic and aminoacylation, were retrieved from the database, as well as the participant molecules.

PiCell Core

The PiCell Core is responsible for structuring the information acquired from databases and inserted manually in such a way that it can be more easily manipulated, checked for inconsistencies, and be further translated into an extended biochemical network.

CHROMOSOME REPRESENTATION The first function of the PiCell Core is to create a representation of the cell's chromosomes based on the genetic information provided. Each chromosome is divided into regions according to annotated regions and respecting a maximum region length. In the case of *M. genitalium*, the maximum region length was set a very high value so that all the regions' sizes are only constrained by the annotations in the genome. Transcription Units' starts and ends were not considered in this process.

RECURSIVE CREATION OF CANONICAL REACTIONS The second function of the PiCell Core is to generate missing canonical reactions for macromolecules in the model. This functionality is based on the premise that all macromolecules in the model must have at least one biosynthesis and one degradation reaction. Thus, this process can iterate from protein complexes needing their complexation reaction, up to the expression of their respective genes. For example, consider that a given metabolic reaction inserted in the model is catalyzed by a protein complex. The complex must be synthesized by a protein complexation reaction. The monomers required in this reaction must be synthesized by a translation reaction from their respective mRNA. The mRNA then needs to be synthesized by a transcription reaction from its respective DNA regions. Finally, DNA regions must be synthesized by their replication reactions. This cycle of reactions must be created for every macromolecule in the model. Similarly, the degradation reactions for each macromolecule is created. All reactions created by the PiCell Core are based on the templates described before. Particularities of each reaction created, such as specific chaperones in protein translation, are added in the reactions according to data availability in the database.

CONSISTENCY CHECKS Additionally to the premise presented in the last paragraph, the PiCell Core performs a mass-balance check in order to probe for inconsistencies in the reactions. All metabolites must have their composition formula described in the model. From their atomic composition, their mass is estimated. Given that all macromolecules are combinations of basic metabolites, the mass of all molecules can be estimated upwards. Then, to check the mass-balance consistency of any reaction, we simply calculate the mass of reactants minus the mass of products. The absolute value obtained must be less than one, the mass of a hydrogen

atom. It is important to notice that although this methodology adds an extra layer of confidence in the model, the correctness of all reactions still relies on the data sources.

BIOCHEMICAL NETWORK CONSTRUCTION After the model completion, it is ready to generate a working model following the extended biochemical network framework. For each reaction described in the PiCell core, a respective reaction is created in the network. The molecules are created respecting their location. If a given molecule can occur in more than one location, one molecule node is created for each location and linked to their respective reactions accordingly. Reversible reactions are represented by two reaction nodes, one for each direction. The final model can be exported in SBML, some network formats, and also as a networkx graph. More technical details can be found in the Supplementary Information of [28].

5.5 M. GENITALIUM WHOLE-CELL BIOCHEMICAL NETWORK

Based on WholeCellKB's [60] and genomic information, we built the whole-cell biochemical network of *M. genitalium*, comprising the molecular types and cellular processes indicated in Figures 5.13a and 5.13b.

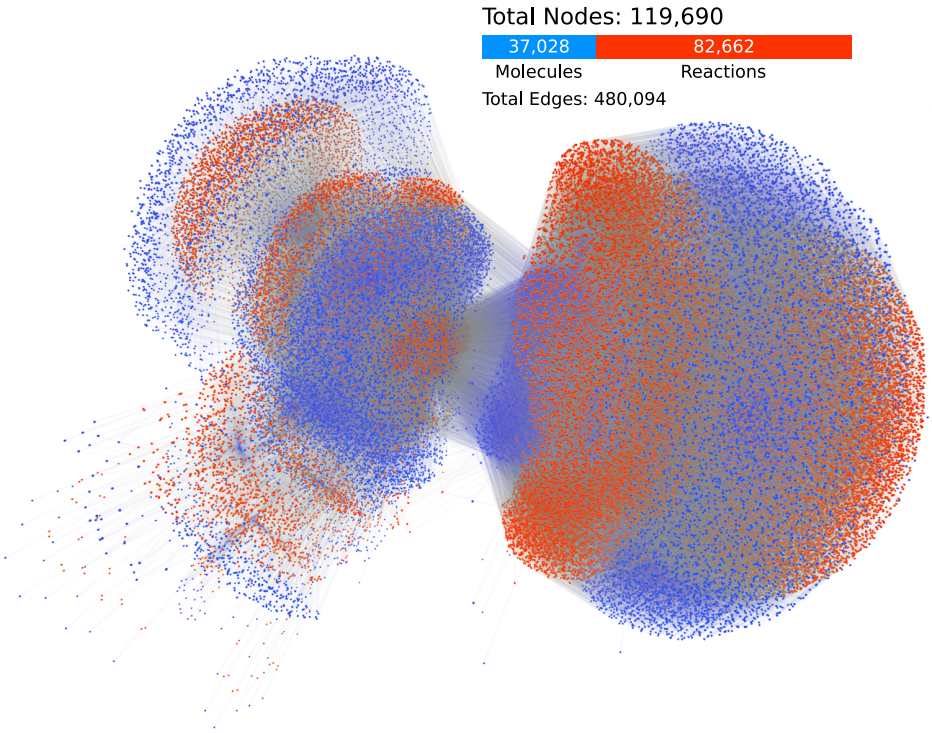

Figure 5.12: Graphical visualization of the *M. genitalium* whole-cell network. Edges with high associated stoichiometry (over 100) are hidden for better visualization. The big circular group on the right is mainly composed by DNA-Protein complexes and their respective formation reactions. Figure generated using unpublished software.

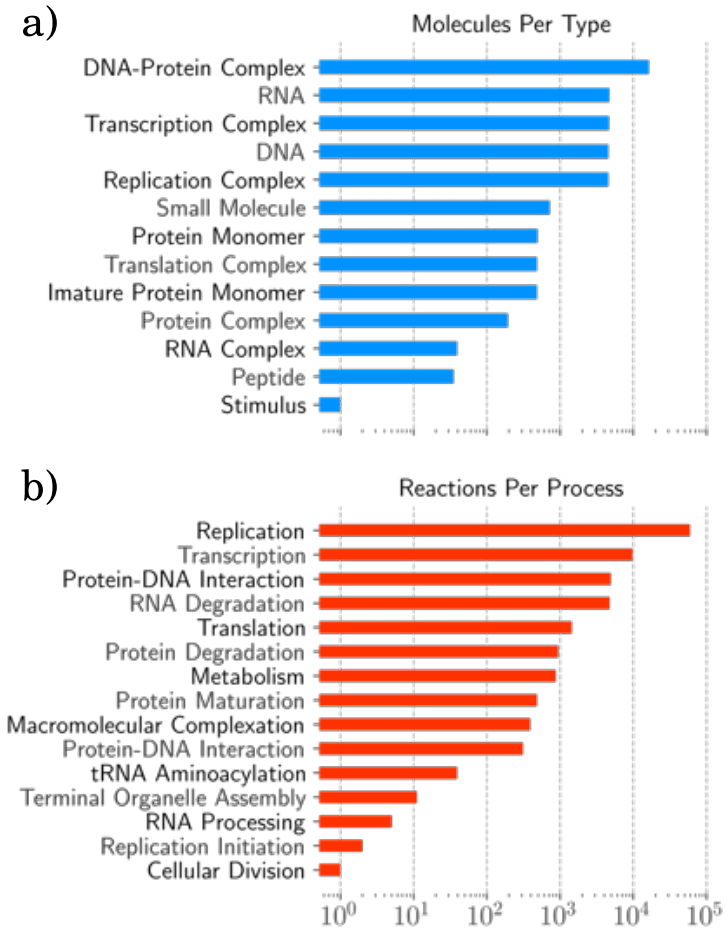

Figure 5.13: a) Number of molecules per functional group in logarithmic scale. b) Number of reactions per cellular process in logarithmic scale.

Topological Analysis

The generated model is a bipartite, weighted, and directed network containing a single connected component with 119,690 nodes and 480,094 links (Fig. 5.12). The nodes comprise 37,028 molecules and 82,662 reactions. The network is available in GML and SBML formats in a Github repository indicated in section "Availability of Data and Materials". We annotated molecule nodes with their functional group. Figure 5.13a shows its distribution among the nodes. We also annotated reactions according to the cellular process to which they belong. As shown in Fig. 5.13b, almost all known cellular processes can be found among the reaction nodes.

Figure 5.14 depicts the distribution of nodes within the five considered locations: cytosol (c), cellular membrane (m), Terminal Organelle Cytosol (tc), Terminal Organelle Membrane (tm), and extracellular environment (e). The 249 molecules located in the extracellular environment account for nutrients, side-products expelled from metabolism as well as secreted proteins.

Considering the *degree* of a node as the number of connections it owns, we analyzed the degree distributions for molecule and reaction nodes. For both cases, multi-modal distributions were obtained. It is known that metabolic and protein-

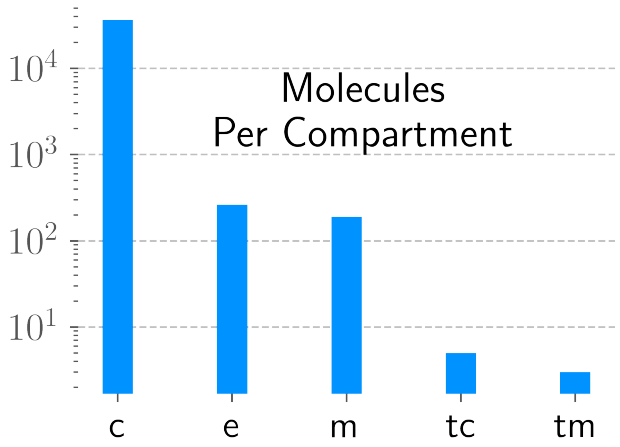

Figure 5.14: Number of molecules per compartment in logarithmic scale.

protein interaction networks often have power-law-like degree distributions [39–41, 64]. Thus, we analyzed the distribution of proteins and metabolites separately (Fig. 5.15a). We found that their distributions corroborate literature, but this assumption is not true for the whole system. The degree distribution for reactions showed different well-separated distributions that were found to be related to different processes (Fig. 5.15b). For instance, Protein Degradation reactions were found grouped in a Gaussian-like region. Other processes have reactions with a signature degree, such as DNA Replication and Ribosome Assembly, the former being accounted into the Macromolecular Complexation process.

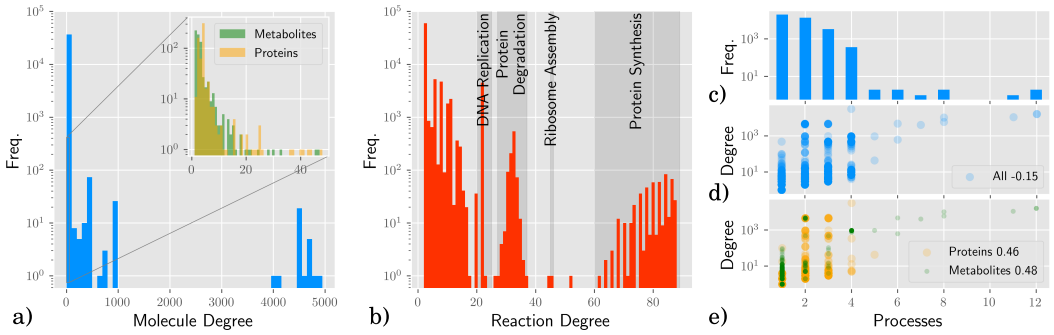

Figure 5.15: a) Degree distribution of molecule nodes. The subgraph in the figure is the same degree distribution but showing only the protein and metabolite subgroups of nodes. b) Degree distribution for reaction nodes where the degree is the number of connections solely. c) Distribution of number of processes a given molecule node participates. d) Spearman correlation between node degree and number of processes for all molecule nodes. e) Spearman correlation between node degree and number of processes for only proteins and metabolites.

Regarding the network completeness, 20% of the proteins monomers and protein complexes (a total of 137 molecules) have no interactions described in the WholeCellKB. These molecules are connected only to their biosynthesis and degradation reactions, participating in no other interactions. Many of these molecules with unknown function are putative membrane proteins. Because there is still no signaling

pathway reported in *M. genitalium*, these putative membrane proteins can be a starting point for the elucidation of signaling processes in this organism.

Network models of natural systems usually share some topological characteristics. The degree distribution through the nodes of a network is one of these shared features [37]. Somehow, it is believed that the processes that create these natural systems, particularly the biological systems, resulted in patterns of interaction between its elements which follows a power-law distribution [39–41, 64]. We found evidence that the degree distribution in our network resembles a power-law but only for selected molecule groups. It is the case of proteins and metabolites. The degree distribution when considering all the nodes is not trivial. Whilst having found such power-law-like structures inside the network, they represent a small fraction of all molecular entities represented in the model. The assumption of a cell's robustness explained by network topologies seems more limited in this perspective, even though the so obtained distributions rely on our modeling assumptions. Nonetheless, the properties of the power-law-like subsets could propagate to the rest of the network by cascading failures.

Interface Between Processes

Molecules shared by reactions from different processes are interfaces between them. It means that the concentration of such molecules can affect, and be affected by, the dynamics of more than one cellular process. We found that 46.8% of the molecule nodes participate in at least two cellular processes, making the bridges between them. Parting from the hypothesis that the more connected a molecule is in the network, the higher is the chance that it connects cellular processes, a positive correlation is expected between the node's degree and the number of processes that it participates. Figure 5.15d shows that it is not the case, where we obtained a low negative correlation between those measures. Nevertheless, we found a significant positive correlation when analyzing only proteins and metabolites (Fig. 5.15e).

Remarks on the Mycoplasma genitalium Whole-Cell Biochemical Network

We chose the pathogenic bacterium *M. genitalium* organism as a case study to illustrate the capabilities of the proposed framework because of its simplicity and data availability in the WholeCellKB database. Despite its simplicity, the so obtained network comprises thousands of biochemical interactions from which several cellular processes emerge.

This approach provides a means such as that all cellular processes are explicitly described as a unified network of biochemical interactions. To support this affirmation, we mention the fact that the whole-cell biochemical network of *M. genitalium* has only one component, which means that all biochemical interactions are in some level interconnected. Moreover, 46.8% of the molecules participate in reactions respective to more than one process, acting as the intrinsic interfaces between them. Despite the analyses performed in this work, we believe that more information about how processes are integrated could be extracted from this model.

Indications about the reliability of the obtained reactions can be derived from the WholeCellKB, where each reaction is assigned to its respective literature evidence.

The reactions in the network that were not explicitly described in the database, such as replication, transcription, and translation reactions, were derived from basic biological knowledge about these processes. Some of these reactions also incorporated particular molecules such as transcriptional factors and chaperones, which therefore have their literature evidence indicated in the database as well.

5.6 GENE ESSENTIALITY PREDICTION

Some specific gene deletions can trigger a deadly cascade of failures in cells. Other deletions might not cause such an impact, characterizing cellular robustness [44]. Cascading failure analysis has been used to evaluate robustness on several network-based systems [65–68]. Analogously, it has been successfully applied for the estimation of essential genes in metabolic networks. They removed reactions regulated or catalyzed by a given gene product, and the impact of cascade failure propagation on the network structure revealed a correlation with gene knockout lethality [69, 70].

The proposed framework and metabolic networks share many structural characteristics. Because of their similarity, the same cascading failure approach can be used to estimate the impact of gene deletion on a whole-cell biochemical network.

We adopted the same algorithm described by Mombach *et al.* [69] to perform the cascading failure analysis. As demonstrated in their work, cascade failures were started by removing reactions regulated by a given enzyme in order to quantify its essentiality. However, enzymes and other regulators are explicitly connected to their respective reactions in our framework. Thus, we use molecule nodes which represent the functional molecule of each gene as starting points of the cascade failure dynamics, instead of reaction nodes, as shown in Figure 5.16 and described in the Algorithm 1. For genes that are further translated into proteins, we selected the molecule nodes, which represent its protein monomers. For tRNAs and ribosomal RNAs, the selected nodes were the ones representing the RNA molecules themselves.

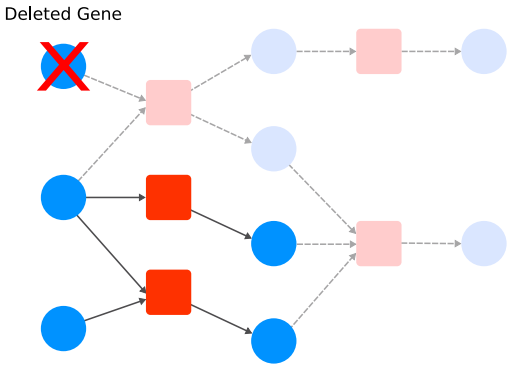

Figure 5.16: Example of cascading failure. The nodes with shaded color and dashed edges were removed after cascading failure by removing the initial gene molecule node.

Algorithm 1 Cascading Failure

```
1: procedure RECURSIVENODEREMOVAL(N,TARGET_REACTION)   ▷ N is the molecule node to remove
2:     R ← list of reactions where Nis reactant
3:     M ← empty list of molecules
4:     remove(N)
5:     while Length(R)> 0 do
6:         for all r in R do
7:             for all products of r do
8:                 if product.indegree = 0 then
9:                     append product to M
10:            remove(r)
11:        R ← empty list of reactions
12:        for all m in M do
13:            append reactions where m is reactant to R
14:            remove(m)
15:        M ← empty list of molecules
```

In order to observe the effect of gene deletion in the network, we checked if a specific critical reaction, the cellular division, is still present in the network after the gene deletion. In other words, we remove the node representing the respective gene and perform the cascading failure analysis. Then, the gene is classified as essential if the critical reaction is absent in the network. Otherwise, the gene is classified as non-essential. Additionally to single gene deletions, we performed double gene deletions to analyze relationships between pairs of genes.

We performed single and double gene deletion experiments. For single-gene deletions, each gene was removed following by its cascading failure. Table 5.1 shows the comparison with experimental data achieving 54% of exact matches with 1.7% of false positives.

Table 5.1: Validation of gene essentiality predictions against experimental data

	Match	**False Positive**	**False Negative**
Single	0.5409	0.0171	0.442
Double	0.56	0.0190	0.421
Only Essential	0.9524	0.0476	0.0

For double gene deletion, we simultaneously removed each pair of genes and performed the cascading failure. The double gene deletion slightly enhanced the classification, which resulted in 56% of correct matches. Among the genes classified as essential only in the double deletions, three distinct groups could be outlined as depicted in Figure 5.17. The first is composed of three genes (MG071, MG322, and MG323) responsible for ion transport across the membrane. The second group is composed of seven genes (MG020, MG046, MG183, MG208, MG239, MG324, and MG391) involved in the protein degradation process. The deletion of almost any combination of genes from these two groups, except for MG239 that is only essential

with MG071, indicates a non-viability of the cell. The third group is composed of two genes (MG013 and MG245) that are involved in the folate metabolism and are only classified as essential if both are simultaneously removed.

It is important to observe that genes with unknown functions, which compose 22.24% of all genes, were included in this analysis. These genes might have a considerable impact on the matches rate, once all were classified as non-essential. Nonetheless, considering only genes classified as essential by the combined approach, the correct matches increase to 95%, indicating the high accuracy of the model.

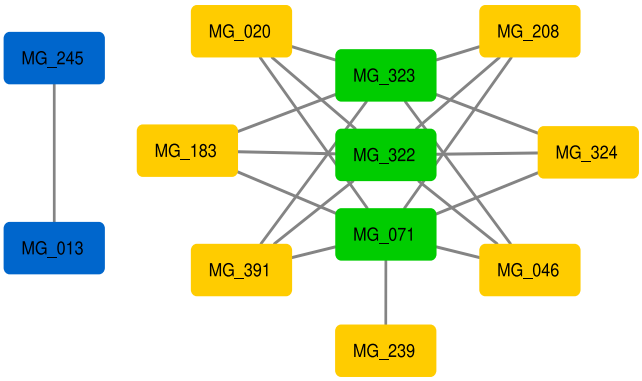

Figure 5.17: Genes classified as essential in double deletions. The connected pairs means that their conjunct deletion resulted in the deletion of the cell division reaction. They are grouped by colors respective to their functional groups: cross-membrane ion transporters (green), protein degradation (yellow), and folate metabolism (blue)

Network Rewiring

Randomization of network topologies is one of the most commonly applied methods to investigate how much information is encoded within a model [71–73]. Here we employ a randomization process that changes the source and target of each edge with a probability p. For each edge, a random number between 0 and 1 is generated from a uniform distribution. If this number is smaller or equal than the probability p, the source and target of this edge are assigned to new nodes randomly chosen. The new nodes are chosen among ones with the same type to keep the bipartite characteristic of the network. In order to validate the *M. genitalium* whole-cell biochemical network generated in this work, we simulated gene deletions by performing the removal of nodes that represent each gene product, followed by cascade failure, and then analyzed the damage caused on the network. The simulation results regarding the 525 genes of *M. genitalium* were compared to experimental gene essentiality classification by global transposon mutagenesis gene disruption available in the literature [61]. The experimental data classified 382 genes out of 525 as being essential to the organism so it can replicate itself.

In order to check the statistical relevance of the *M. genitalium* whole-cell biochemical network, we generated randomly rewired networks based on the original one by reconnecting each edge, preserving directionality, with a probability p. A total of 50 replicates were generated for each probability with p ranging from 0 to 1 with a

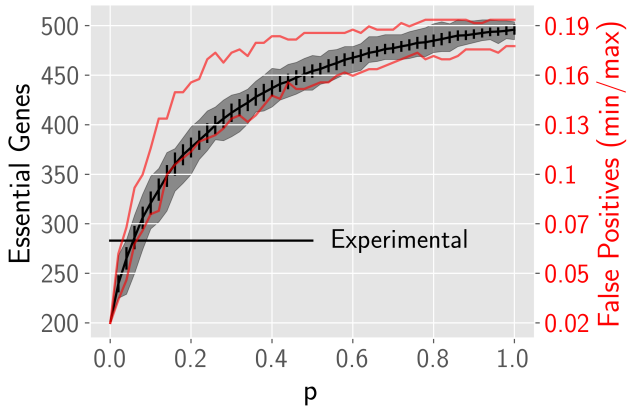

Figure 5.18: Gene essentiality prediction of randomly rewired networks with probability p from the original *M. genitalium* whole-cell biochemical network. Each rewiring rate accounts for 50 replicates.

0.02 step. In Figure 5.18 we can observe that the higher the network randomization level, the higher is the number of false positives in the essentiality classification.

5.7 RELATED MODELING METHODOLOGIES

Network-based models have been extensively used to represent cellular processes. Besides to metabolic and signaling networks, which was already tackled in this work, we can give as examples: a) genetic regulation network, representing positive and negative relationships between gene expressions [6, 74]; and b) protein-protein interaction (PPI) networks, indicating physical interactions between proteins [75–77]. Some of these networks, namely metabolic and signaling networks, are already representing their processes based on their underlying biochemical reactions. On the other hand, genetic regulation and PPI networks represent processes that are composed sometimes by several intermediate biochemical reactions, therefore providing a high-level representation.

If we consider a set of networks representing processes of the same organism by using the network models above mentioned, there is currently no means for a straightforward integration between them once they have very different modeling assumptions. Studies addressing the integration of cellular processes often maintain the respective underlying networks in separated layers [78–81].

No quantitative comparison can be established between the so constructed *M. genitalium* network and any other network model for two reasons: a) there is no other network model for this organism; b) no modeling approach have integrated cellular processes that could be topologically addressed as a single network. Nevertheless, the closest approach available in the literature is the one provided by *Reactome.org* [82]. They aim at representing several different cellular processes through biochemical reactions. However, biochemical pathways are organized in a nested form, thus not providing a single integrated network. Also, processing processes, such as transcription and translation, are not represented. Even so, *Reactome.org* represents an accessible, curated, and annotated biochemical content which can be further transcribed, with no much effort, into single whole-cell networks using our approach.

Regarding the network's construction process, our approach could further benefit from related methodologies for metabolic networks. For instance, orthology-based network reconstructions [83, 84] could be directly applied using the *M. genitalium* whole-cell biochemical network and other networks of model organisms that can emerge. The development of databases containing whole-cell biochemical networks would be increasingly useful for such reconstructions as new models are provided, also benefiting the study of non-model organisms.

6 SIMULATING BIOCHEMICAL SYSTEMS

Computational models of biochemical reactions have proven to be useful tools for a better understanding of molecular systems [85, 86], as well as in optimizing experimental research [87, 88]. As a consequence of ever increasing data availability, these models are significantly growing in size and complexity toward the scale of entire cells [23–25].

Simulations of such models can be carried out by applying among a choice of many available methods (e.g., Ordinary Differential Equations, Stochastic Simulation Algorithms, and Dynamic Flux Balance Analysis) [18, 89]. However, each simulation method has its advantages and limitations, which influence their choice concerning specific applications. In this chapter, we will discuss important characteristics to be considered when simulating biochemical systems at a cellular scale and introduce two simulation methodologies that consider these important characteristics. Part of the content presented in this chapter is also available in [29].

6.1 CELLULAR-SCALE BIOCHEMICAL SYSTEM CHARACTERISTICS

Complex reaction systems, such as biochemical interactions in a cellular environment, have some characteristics that are important to consider when modeling and simulating them. One of these characteristics is that complex biochemical systems often involve reactions taking place at significantly different time-scales. Also, the number of copies of molecular species can vary from a single to thousands of molecules [4, 90].

Another important characteristic is that the number of copies of a given molecular species at any time takes a discrete value. However, when the system involves a sufficiently high number of copies for all species involved, it is reasonable to treat this quantity as a continuous variable [16]. With this assumption, the most straightforward approach to simulate such systems is to solve a set of coupled, first-order, ordinary differential equations called Reaction-Rate Equations (RE) [91–93]. Other approaches have been developed for the case when the rate-constants are not available. It is the case of the Dynamic Flux Balance Analysis [18, 19]. Despite the computational efficiency of these methods for the case of many molecules, when it comes to smaller numbers, the results can be unsatisfactory in the sense that they will provide real-valued approximations of quantities that are in fact discrete-valued. Also, these methods provide deterministic solutions, while reactions are better described as stochastic processes [17, 94].

A physical consequence of a small number of copies of a given molecule in a biochemical system is that the reactions involving it will have their rate highly influenced by thermal fluctuations and molecular crowding. A better description of these reaction rate fluctuations can be achieved through stochastic formulations of the system. Then, the evolution of molecular counts would be a consequence of the reaction occurrences following a given probability rather than a rate. The

single-valued function that gives the probability of finding a particular set of molecular counts at a given time is called the Reaction Master Equation (RME) [95]. Its behavior in time can be formulated as a Markovian random walk in RME space. Although it is often impossible to solve an RME analytically, the most frequently used method being equivalent to numerically solving it is the Stochastic Simulation Algorithm (SSA), also known as the Gillespie Algorithm [96, 97]. Even though SSA can adequately account for the inherent discreteness and stochasticity of molecular interactions, it typically implies a high computational cost. Some faster methods were derived from approximations of SSA, such as the τ-Leaping method [98, 99].

So far, we pointed out some important characteristics shared by many complex biochemical systems, such as different scales of reaction rates and molecular concentrations, discrete-valued number of molecules, and random fluctuations in reaction rates. Therefore, an ideal simulation method for such systems should be able to encompass all these features while being computationally efficient.

6.2 STOCHASTIC SIMULATION ALGORITHM

Since its publication by Daniel Gillespie in [96], the Stochastic Simulation Algorithm (SSA) has been the standard approach to simulate stochastic systems. In the most used form of this algorithm, called the *Direct Method*, a well-mixed solution where M molecular species can react through R reaction channels is assumed. Given an initial state X_0 for the number of copies of each molecular species, we define a time-step τ where only one of the R possible reactions can occur. For each reaction channel μ in R, we must calculate its propensity such as:

$$a_\mu = h_\mu c_\mu \tag{6.1}$$

where h_μ is the number of possible combinations between the reactant molecules of μ in the current state of the system $X(t)$. The c_μ is a reaction constant that can be understood as the probability of that reaction occurring in a period. It is related to the rate constant of the reaction k_μ and the volume of the system V as follows:

$$c_\mu = \frac{k_\mu}{V^{I_\mu - 1}} \tag{6.2}$$

given that I_μ is the number of molecules reacting in r_μ. The simulation step is then performed by selecting a pair of μ and τ such as:

$$\begin{aligned} a_0 &= \sum_{i=1}^{R} a_i \\ \mu : \sum_{i=0}^{\mu-1} a_i < r_1 a_0 &< \sum_{i=0}^{\mu} a_i \\ \tau &= \frac{\ln(1/r_2)}{a_0} \end{aligned} \tag{6.3}$$

where r_1 and r_2 are random numbers drawn from a unitary uniform distribution. We then update the system state X by performing the reaction μ and then increase time t by τ. This procedure is repeated until a stopping criterion is reached.

Although SSA has been extensively tested and applied in the simulation of several stochastic systems, it tends to be a computationally expensive algorithm that may become prohibitive for very large systems [17]. There are some optimized formulations and approximation of the SSA aimed at improving computational performance [98]. In the following subsection, we present one of the most used approximation methods, the τ-Leaping method.

The τ-Leaping Method

The τ-Leaping method, also proposed by D. Gillespie [98], is an approximation of SSA which is considerably faster to compute. In this method, at each iteration, we leap in time by τ so that the changes in the propensity function a will be slight. Then, the number of occurrences k_μ in the time step τ for each reaction channel $\mu \in R$ will be sampled from the Poisson random distribution:

$$P(k; a_\mu, \tau) = \frac{e^{-a_\mu \tau}(a_\mu \tau)^k}{k!} \qquad (k = 0, 1, 2, ...). \tag{6.4}$$

6.3 FLUX BALANCE ANALYSIS

Flux Balance Analysis (FBA) is a widely used mathematical method to study cell's metabolism behavior. The first works describing this technique drawback to the '80s where they solved fluxes in metabolic pathways using linear programming (LP) [100]. In the last decades, FBA has been successfully applied to genome-scale metabolic networks being capable of accurately predict cell's behavior [18, 101–103].

The formulation of an FBA problem [104, 105] starts with a metabolic network. Such a network can be represented by a stoichiometric matrix S of size $m \times n$. Each row of the matrix S represents one of the m metabolites in the network. Each column represents one of the n reactions. The entries in matrix S account for the stoichiometry of each metabolite in each reaction. Negative values indicate metabolites consumed and, positive values, metabolites produced by a reaction. Metabolites that do not participate in a reaction have their values equals to zero in the respective column.

Let v be a vector of size n that represents the fluxes through reactions of the metabolic network and x is a vector of size m which represents the concentration of all metabolites. The steady-state of the system can be given as

$$\frac{dx}{dt} = 0 \tag{6.5}$$

which can be represented as a system of mass-balance equations

$$S \cdot v = 0 \tag{6.6}$$

.

In any realistic metabolic network, the number of reactions is greater than the number of metabolites ($m > n$). Thus, there are more variables than equations

allowing more than one solution to the equation system. The space of solutions can be reduced by adding boundary constraints for the fluxes

$$a_i < v_i < b_i \tag{6.7}$$

.

The constrained solution space is composed by all v that satisfies the Equations 6.6 and 6.7. However, we may be interested in solutions that maximize an objective, such as, achieve a maximum growth rate or maximize the production of a specific metabolite. That objective can be expressed as a linear combination of fluxes such

$$Z = c^\intercal v \tag{6.8}$$

where c is a vector of coefficients for each flux. So, FBA uses linear programming (LP) to find a v that maximize or minimize Z. Summarizing, the canonical FBA problem can be stated as

$$\begin{aligned} &\text{maximize or minimize} && Z = c^\intercal v \\ &\text{subjected to} && S \cdot v = 0, \\ & && a_i < v_i < b_i \end{aligned} \tag{6.9}$$

.

By the end of the optimization, the output is a flux vector v that maximizes or minimizes Z.

6.4 DYNAMIC FLUX BALANCE ANALYSIS

Flux Balance Analysis is capable of estimate qualitatively the behavior of metabolism with no parameter other than a stoichiometric matrix from a metabolic network. However, its original formulation, even being capable of predict metabolism's response to environmental change or genetic regulation, is not time-linked and can only predict instantaneous transitions. Also, the original FBA is not capable of predicting metabolites concentrations. Some efforts have been made to produce time-linked simulations of metabolism predicting transition phases due to environmental or genetic changes as also time-series of metabolite's concentration [106, 107].

In order to achieve flux profiles through time during transition phases between steady-states, there was proposed an FBA variation called *Dynamic Flux Balance Analysis* (DFBA) [18]. This framework was implemented in two different ways: dynamic optimization where requires the solving of a non-linear programming (NLP) problem and static optimization which requires the solving of multiple linear programming (LP) problems. The dynamic optimization solves an NLP problem for a particular spanning of time with a global objective function, returning temporal profiles of fluxes and metabolites concentrations as well. The static approach divides the spanning time of interest into smaller intervals and at the beginning of each interval, an instantaneous optimization problem is solved in order to predict the fluxes at the time point and then followed by integration over the interval to compute metabolites concentration over time [18]. The DFBA approach could

quantitatively predict the diauxic batch growth of *E. coli* when comparing with experimental data and showed that static optimization performed a more accurate prediction.

A similar approach called *time-linked FBA* (tFBA) extends DFBA by separating process reactant consumption and byproduct return to occur between time steps [107]. It adapts the model to allow a more confident data integration with long-term reactions executed in whole-cell models.

Another approach that enhances dynamic models of FBA is called *Integrated Dynamic Flux Balance* (idFBA) and it proposes an integrated dynamic analysis of biochemical networks bringing together signaling, metabolic, and transcriptional regulation processes [108]. For that, the signaling network is represented using stoichiometric formalism expanding the metabolic network represented by S and an incidence matrix $I_{m' \times t_N}$ where m' is the number of reactions of the expanded stoichiometric matrix S and t_N the number of time steps. Each element ij of the incidence matrix I has binary values representing whether a reaction i occurs at time j or not. This approach allows integration of slow reactions by setting the value of reaction i to 1 only at time steps which respect a delay τ_{delay}. The transcriptional regulation is represented by boolean formalism evaluating at each time step setting fluxes upper bound (b_i) to 0 of reactions regulated by a gene not expressed at this time step. Fluxes are estimated at each time step similar to the static optimization using LP described in [18]. This method showed good predictions when compared with kinetic models of the *S. cerevisiae* high-osmolarity glycerol pathway in response to osmotic stress [108].

7

CONSTRAINT-BASED SIMULATION ALGORITHM

In the last chapter, we discussed some important characteristics to be considered when aiming at simulating biochemical systems at a cell scale such as discreteness, stochasticity, and efficiency. We also described two methodologies, one that encompasses the discreteness and stochasticity but lacks efficiency, and another that is efficient but cannot account for stochasticity.

In this chapter, we propose a simulation method aimed at fulfilling all these important characteristics. We use constraint-based modeling to mathematically represent biochemical networks modeled using the framework proposed in Chapter 4. Additionally, we compare the proposed algorithm with the SSA using a theoretical biochemical model. The content presented in this chapter is also available in [29].

7.1 CONSTRAINT-BASED MODELING

Let $m = \{m_1, m_2, ..., m_M\}$ be a set of M molecular species and $r = \{r_1, r_2, ..., r_R\}$ be a set of R reactions where molecules in m play roles as reactants and/or products. The stoichiometry of each reaction, *i.e.* how much of each molecular species is consumed or produced, can be represented by means of a stoichiometric matrix $S_{M\times R}$. Every entry S_{ij} yields how much of the molecular species i is consumed (negatively signed) or produced (positively signed) by the reaction j.

The stoichiometric matrix S cannot yield information about the molecular species whose counts do not change by the end of the reaction. This is the case of catalysts, modifiers, or auto-catalytic reactions (e. g.A $\longrightarrow$ A+A). Thus, to represent such cases, we will define a regulation matrix $R^*_{N\times R}$ where $R^*_{i,j} = \gamma$ if γm_i molecules are needed in reaction r_j. Figure 7.1 illustrates how a biochemical network can be mathematically represented.

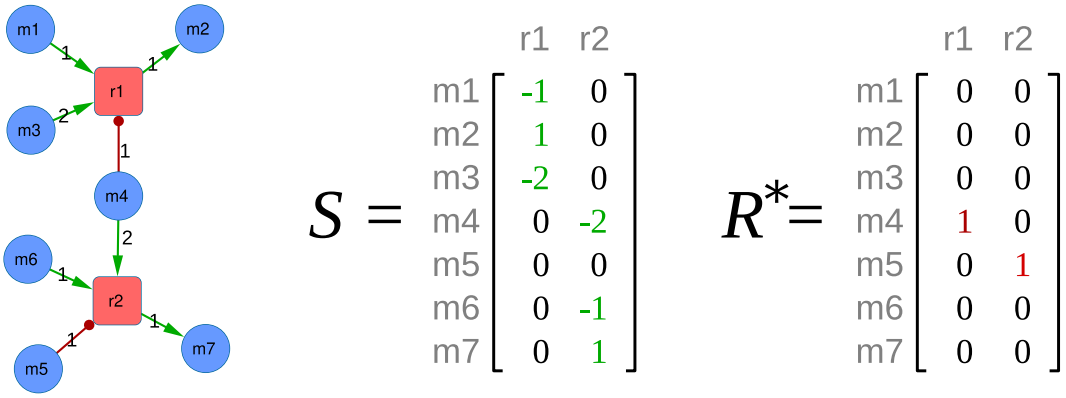

Figure 7.1: An example of how a biochemical network modeled using the framework presented in Chapter 4 can be represented by means of a stoichiometric and a regulatory matrix.

If we consider a vector x of length M, where x_i is the number of m_i molecules, homogeneously distributed in a given volume V, the variation of x along time can be written as:

$$\frac{dx}{dt} = S \cdot v, \tag{7.1}$$

where v is a vector of fluxes and has length R. The flux v_i can be understood as how many times the reaction i occurs in a given period. The latter will henceforth be considered as 1 second.

Although v is, for now, an unknown variable, its values are constrained by well-known physical properties such as mass balance. Therefore, we can set the following constraints to the solution space:

$$x \geq 0, \tag{7.2}$$

which assure that there are no negative number of molecules, and

$$Sv + x \geq 0, \tag{7.3}$$

asserting that a given flux is valid if, and only if, it does not result in any negative number of molecules. An example of solution space for two parallel reactions is depicted in Figure 7.2. The allowable space is upper-bounded the line that determines the mass-balance constraint.

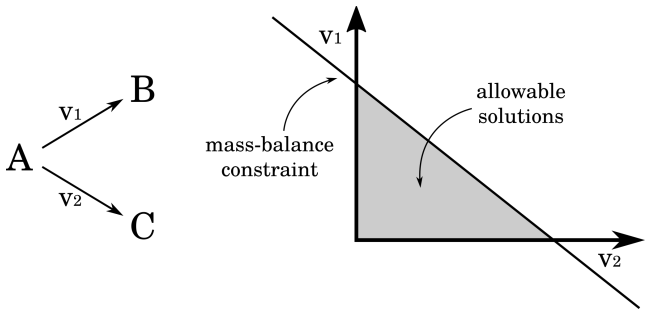

Figure 7.2: Example of a solution space for a system with two reactions. The flux for each reaction is represented in the axes. Both reactions share the same reactant A, thus, the combination of fluxes v_1 and v_2 must satisfy the mass-balance constraint.

7.2 SIMULATION STEPS

Given an initial condition, each iteration of the algorithm is divided into five steps: 1) calculate the flux of each reaction based on the current system state; 2) add noise; 3) check if the calculated fluxes satisfy the mass-balance constraints (Eq. 7.3); 4) if mass-balance constraint not satisfied, find a valid set of fluxes; and 5) update the molecular counts. Figure 7.3 illustrates the algorithm scheme. These steps must be repeated until a given stopping criterion is reached, such as a maximum time.

The following paragraphs describe in detail each step of the algorithm.

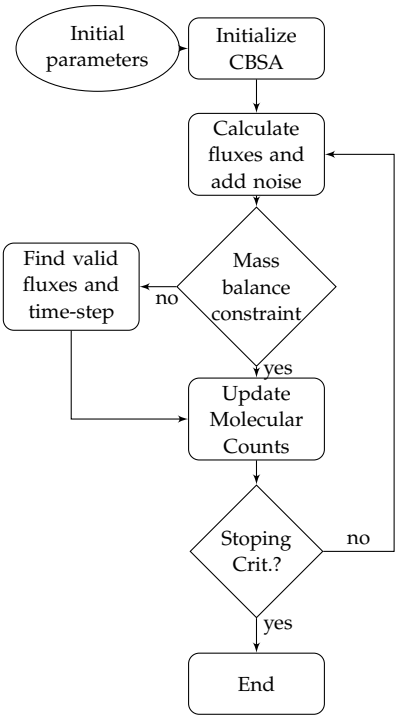

Figure 7.3: Flowchart of the Constraint-Based Simulation Algorithm.

INITIAL PARAMETERS The chemical system is defined by the stoichiometric matrix S, which describes the reactants and products of every reaction, and the regulation matrix R^* that indicates other molecules necessary to given reactions but without being consumed nor produced in the process. Each reaction must also have an associated constant c, which is the fraction of all combinations between its reactant molecules that would indeed result in a reaction by the time of one second. Also, the initial state of the system x_0 must be provided, defining the initial molecular count for every molecular species. Finally, we need to set a maximum time-step length Δt_{max}.

FLUX ESTIMATION The flux $v_i(t)$ of a reaction r_i in a given instant t can be understood as how many times the reaction r_i can occur along the next second. Therefore, the flux of a given reaction can be any function $g(x)$ multiplied by the total number of combinations between the reactants. For the sake of simplicity, we will consider $g(x)$ as a constant c_i for each reaction. Thus, following the same reasoning used in SSA to estimate a, we can write $v_i(t + \Delta t)$ as follows:

$$v_i(t + \Delta t) = c_i \prod_{j \forall S_{j,i} < 0} \frac{\binom{x_j(t)}{-S_{j,i}}}{-S_{j,i}!} \prod_{j \forall R^*_{j,i} > 0} \frac{\binom{x_j(t)}{R^*_{j,i}}}{R^*_{j,i}!} \Delta t \tag{7.4}$$

.

ADDING NOISE TO FLUXES In order to account for fluctuations which might influence the velocity of reactions, specially in the case of lower molecular counts, we will add noise to the calculated flux on each step using the chemical Langevin

Equation [98, 109]. This equation introduces a Gaussian white noise η in the flux v (Eq. 7.8) as:

$$\begin{aligned} v_i &= v'_i + \eta_i \\ \eta_i &= \sqrt{v'_i}\mathcal{N}(0,1) \end{aligned} \tag{7.5}$$

where $\mathcal{N}(0,1)$ is a random variable drawn from the normal distribution with mean zero, and standard deviation one. The variable v'_i is equal to $v_i + v_{dec_i}$ which will be introduced in the next paragraph.

DISCRETE FLUX CALCULATION In order to account for the inherent discreteness of the system, the flux v must be always integer. However, Eqs. 7.4 and 7.5 can output decimal results. In order to keep $v \in \mathbb{N}$, we will separate the integer from the decimal part and store the later in v_{dec} for the next iteration. Then, v_{dec} and v at each iteration can be calculated using the following operations:

$$\begin{aligned} v_{dec_i} &= v_i - \lfloor v_i \rfloor \\ v_i &= v_i - v_{dec_i} \end{aligned} \tag{7.6}$$

By storing the decimal part in v_{dec_i}, we assure that reactions in which $v_i < 1$ will occur at a delayed time when $v_i \geq 1$.

FINDING A VALID FLUX Although each v_i is mostly valid for every single reaction, this might not hold when considering the whole system. To illustrate such a case, consider a system in which a given molecular species m_k is a reactant for many different reactions. Once the estimation of each v_i is independent of the others, the total amount of m_k required for all reactions might exceed the pool x_k and violate the mass-balance constraint from Eq. 7.3. It may also happen in the case of $c_i >> 1$. To avoid such a condition, if v fails the mass-balance constraint, we may look for the maximal Δt that assure non-negative values on $x(t + \Delta t)$. We do that by multiplying the current Δt by a constant $0 < \alpha < 1$, recalculating the fluxes, updating the decimal part v_{dec}, and testing the mass-balance constraint until a valid solution is found, as shown in Eq. 7.7.

$$\begin{aligned} &while\ (x + S \cdot v < 0): \\ &\quad \Delta t = \alpha \Delta t \end{aligned} \tag{7.7}$$

.

In the worst-case scenario, all v_i will reach zero, and another calculation step begins restoring Δt to its initial value.

MOLECULAR COUNT UPDATE Having a discrete flux v passing the mass-balance constraint stated in Eq. 7.3, we simply update the molecular counts by solving Eq. 7.1 using Euler's method. The update equation becomes:

$$x(t + \Delta t) = x(t) + S \cdot v \tag{7.8}$$

.

7.3 ALGORITHM VALIDATION

In order to test the correctness of the solutions provided by the CBSA, we use here a reaction system that can demonstrate several of the important characteristics of real biochemical systems, such as different magnitudes of molecular counts and reaction velocities, and regulated reactions while demonstrating the method's capabilities and limitations. Other examples can be found in the Supplementary Information.

Let us consider the reaction system {7.8} where a Transcription Factor (Tf) triggers the production of a protein P when active. The Transcription Factor Tf is activated and deactivated spontaneously by a slow reaction. The activated Tf triggers a swift production of hundreds of protein P copies, which are spontaneously degraded. With the system starting with only one Tf molecule, it should transit between its activated and inactivated states, triggering bursts on the production of P (Fig. 7.4a).

$$\begin{aligned} \mathrm{Tf_{active}} &\underset{c_2}{\overset{c_1}{\rightleftharpoons}} \mathrm{Tf_{inactive}} \\ \mathrm{Tf_{active}} &\xrightarrow{c_3} \mathrm{Tf_{active}} + \mathrm{P} \\ \mathrm{P} &\xrightarrow{c_4} \varnothing \end{aligned} \qquad \{7.8\}$$

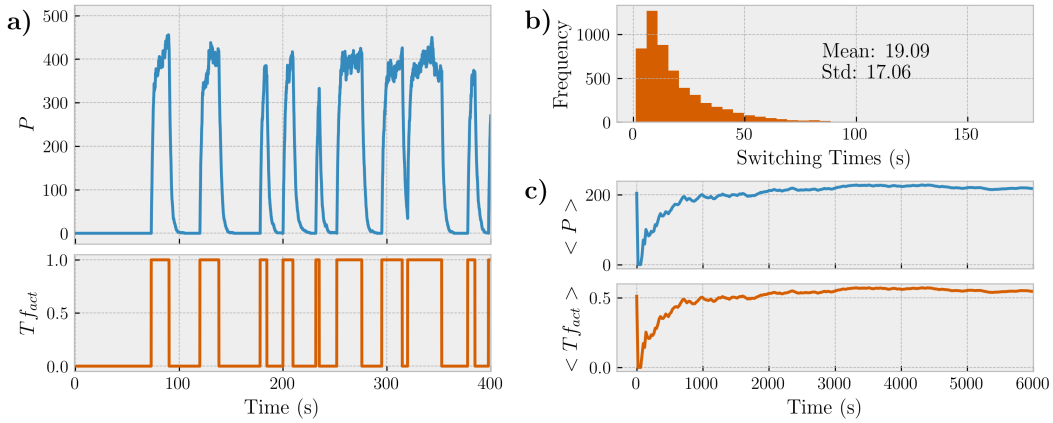

Figure 7.4: **(a)**Simulation of the chemical system described in {7.8} using the CBSA with c_1, c_2, c_3, and c_4 equals to 0.05, 0.05, 200, and 0.5 respectively. The initial state of x contains only one $Tf_{inactive}$ molecule. Every time that $Tf_{inactive}$ turns into Tf_{active}, a burst in P is observed. **(b)** With $c_1 = c_2 = 0.05$, the Tf should transit between states every 20 seconds in average. We can see the distribution of switching times with mean 19.09 and standard deviation 17.06 obtained from a simulation of 100,000 seconds. **(c)** The cumulative mean of Tf_{active}, named $< Tf_{act} >$, at a given time t corresponds to the mean value of Tf_{active} in the interval $[0, t]$. Once $c_1 = c_2$, they have the same transition probability, thus, $< Tf_{act} >$ should converge to 0.5 with $t \to \infty$. Likewise, the cumulative mean of P should converge to 200 once the number of P molecules peaks at around 400 when Tf is active, and it is approximately for half of the time. The parameter α was fixed as 0.5 for all CBSA simulations.

The simulation depicted in Figure 7.4a was performed using the CBSA. The biochemical system is represented employing the stoichiometric matrix S and the regulation matrix R^*. As S can only represent the reaction's final stoichiometry

without accounting for intermediate molecules, we must represent the catalytic Tf_{active} molecule in the regulation matrix R^* as follows:

$$S = \begin{matrix} \\ Tf_{active} \\ Tf_{inactive} \\ P \end{matrix} \overset{\begin{matrix} r_1 & r_2 & r_3 & r_4 \end{matrix}}{\begin{bmatrix} -1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}} \tag{7.9}$$

$$R^* = \begin{matrix} \\ Tf_{active} \\ Tf_{inactive} \\ P \end{matrix} \overset{\begin{matrix} r_1 & r_2 & r_3 & r_4 \end{matrix}}{\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}} \tag{7.10}$$

where r_1 to r_4 are the respective reaction channels with rate constants c_1 to c_4.

The times between activation and deactivation of Tf depend solely in the rate constants c_1 and c_2. Thus, we could estimate these times as $1/c_1$ and $1/c_2$. In the example depicted in Figure 7.4a we adopted $c_1 = c_2 = 0.05$ making the transition times between activate and inactivate states equals to 20 seconds. However, the system is stochastic and the observed transition times will compose a distribution rather than an exact value. In Figure 7.4b we can observe the distribution of the switching times between states of Tf with the mean close to the expected value of 20 seconds. Given any equal values for c_1 and c_2 and the initial condition with only one Tf molecule, we can estimate that the mean value for both activated and inactivated states should converge to 0.5 with $t \to \infty$. This behavior is indeed observed in Figure 7.4c.

Validation Respectively to the Stochastic Simulation Algorithm

The SSA is already a well-established method to perform stochastic simulations of chemical systems. Thus, we use it as a gold-standard method in order to evaluate the correctness of the simulations performed using the proposed Constraint-Based Simulation Algorithm.

Regarding the reaction system described in {7.8}, we already observed agreement between its simulations using CBSA and theoretical mean values expected, as shown in Figure 7.4. Now, we shall compare these results with trajectories obtained from simulations using the original Stochastic Simulation Algorithm and its approximated form, the τ-leaping. Though all methods are stochastic, it is not necessary to perform several runs of the same experiment to obtain statistics about the model because this is a particular oscillatory system and does not diverge at any point in time. Thus, we can make consistent statistics from the time-series with sufficiently long simulation time.

The first comparison we make regards the distribution of switching times between both states of Tf. In Figure 7.5a we can see that the distributions for the three considered methods have the same exponential-like shape with similar means and standard deviations. With the SSA and τ-leaping method, we obtained the mean

switching times closer to the theoretical value. It is also observed that the CBSA allowed a lower standard deviation than the other methods. We believe this is a consequence of the constraints imposed on the model after the addition of noise, which can reject nonviable solutions. In this sense, the constraints may introduce a bias in the distribution from which the random numbers are drawn mainly affecting the standard deviation of reaction times.

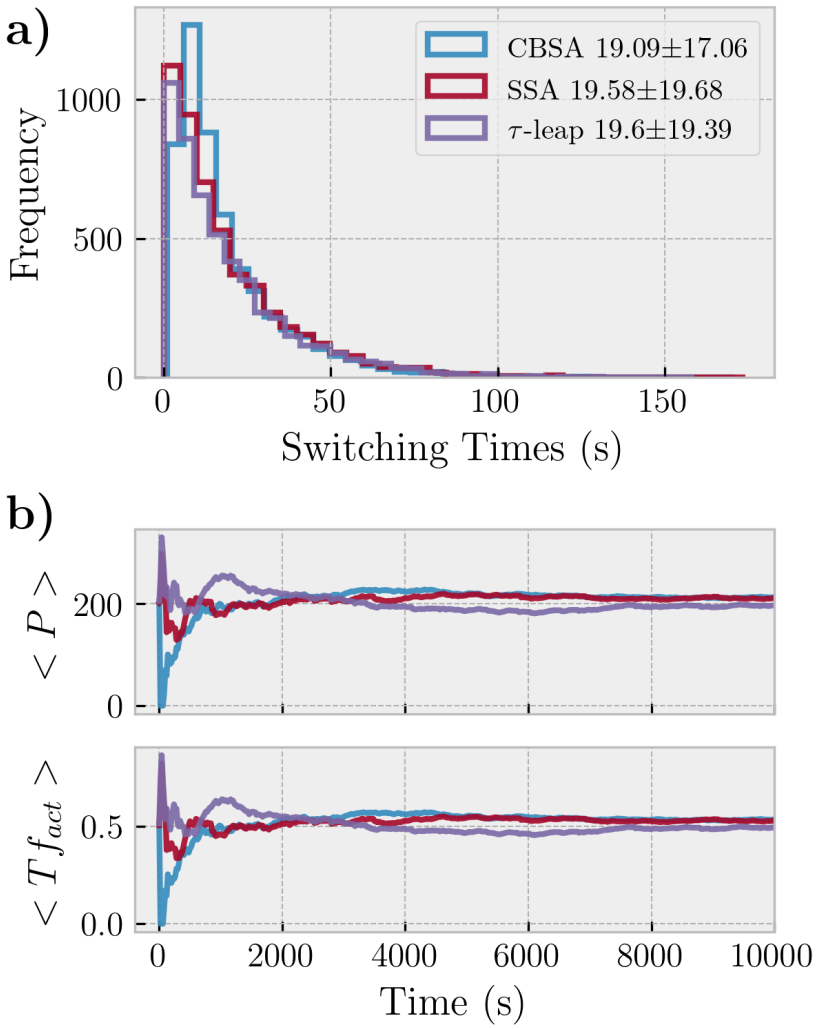

Figure 7.5: **(a)** Distribution of Tf switching times for the CBSA, SSA, and τ-leaping methods with a simulation time of 100,000 seconds. Means and standard deviations indicated in the legend. **(b)** The cumulative mean of P and Tf_{active} for the three methods.

A consequence of the lower standard deviation of reaction times can be observed in the time that the system takes to converge the mean number of molecules to the theoretical value. The mean values of both P and Tf_{active} converge faster to the expected values as depicted in Figure 7.5b. Another limitation of the methods is that it does not necessarily lead to a linear relationship between the time step adopted in the Euler integration and the error obtained when compared to the theoretical value. The main reason for this possible non-linear relationship between time step and error is that the noise η_i added to each flux v_i goes with the square-root of the

flux. So, when $v_i < 1$, the smaller the value that v_i takes, the greater is the noise η_i introduced. Thus, by reducing the integration step Δt we increase the chance o obtaining a flux $v_i < 1$ and the inverted noise proportion.

Despite the differences in the standard deviation of reaction velocities, all methods agreed regarding the qualitative behavior of the system and molecular counts. Other examples can be found in the Supplementary Information where similar conclusions are obtained. We also discuss other aspects of the algorithm such as the influence of Δt and α in the results, relative errors, and in the number of steps to achieve the simulation time.

8

HIGH PERFORMANCE SIMULATION

Parallel computing was a design principle when developing the Constraint-Based Simulation Algorithm aiming at providing a scalable and efficient method. All operations in the CBSA can be computed in parallel thanks to the relative independence of the operations through the vectors and matrices. Therefore, we could take advantage of the massive computing power of GP-GPUs to accelerate individual simulations of large chemical systems. In this chapter, we describe optimizations in the algorithm that makes use of sparse-matrix implementations. We also perform a benchmark comparing the simulation times of sequential and parallel implementations of the CBSA with other methods such as SSA and ODE solvers. The content presented in this chapter is also available in [29].

8.1 SPARSE-MATRIX IMPLEMENTATION

In large chemical systems, it is likely that the representation using a constraint-based model will result in very sparse S and $R*$ matrices, having a large amount of zero-valued entries. This means that the dot products involving these matrices will have to compute several pointless operations, leading to an unnecessary computational effort. To avoid such a situation and also reduce the memory usage of the system, we use a different representation of the S and $R*$ matrices which will be described in more detail below.

Each step of the CBSA has two cost-wise major operations: the calculation of fluxes v (Eq. 7.4) and the dot product $S \cdot v$ (Eq. 7.7 and 7.8). Both operations involve multiplications and/or sums through rows or columns. Thus, we want to reduce the number of operations by reducing the number of rows or columns of the matrices, removing as many zero-valued entries as possible. To do so, we will construct a substitution system for each of the two operations.

Expansion of x and v

The following substitution systems will only work after we expand the vectors x and v by inserting the values 0 and 1 at their respective initial positions, making their sizes $M + 1$ and $R + 1$ respectively.

V Substitution System

To calculate v in a more efficient way, let V_{STO} and V_{IDX} be two matrices of size $T \times R + 1$ where R is the number of reactions in the system and T is the maximum number of reactants plus modifiers which a reaction in the system can have.

Each column in matrix $\boldsymbol{V}_{\text{STO}}$, starting at the second column, will be filled with the stoichiometry of the reactants and modifiers of the corresponding reaction. The entries not used in each column, as well as those from the first column, will be filled with the value one.

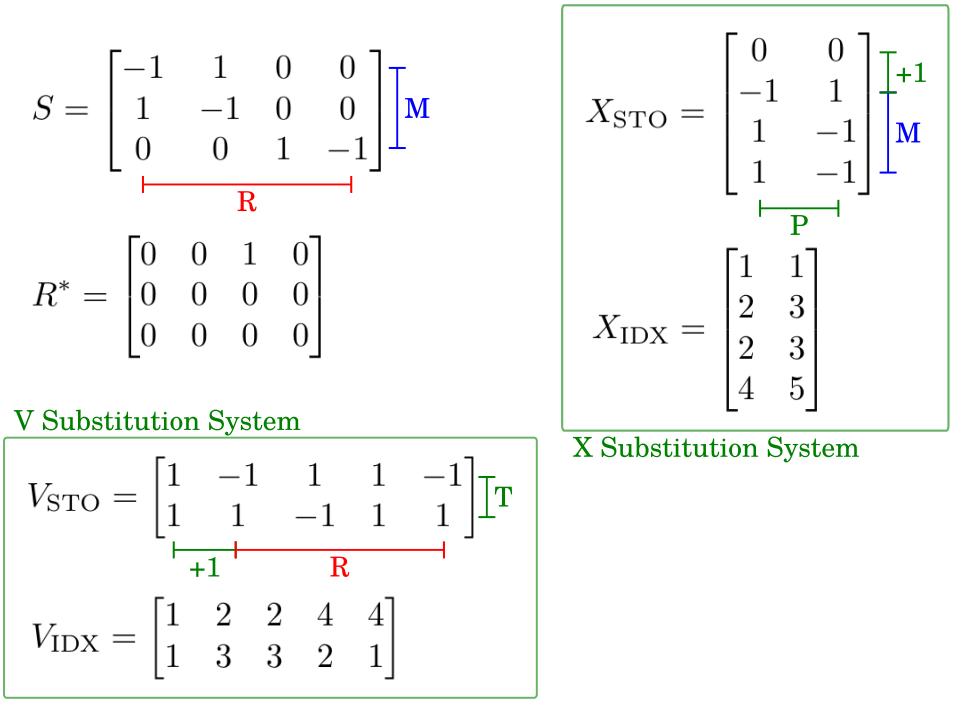

Figure 8.1: Example of the substitution system applied to the reaction system {7.8}. It is important to notice that the indexes in V_{IDX} and X_{IDX} are relative to the expanded vectors of v and x. Thus, the indexes are increased by one relatively to the matrix S.

Each column in matrix $\boldsymbol{V}_{IDX}$, starting at the second column, will be filled with the indexes of the reactants and modifiers of the correspondent reaction. The entries not used in each column, as well as those from the first column, will be filled with the value one (or zero depending on the initial index assumed).

Then, we can rewrite the Equation 7.4 as:

$$v_i(t+\Delta t) = c_i \prod_{j=1,2,\ldots T} \frac{\binom{x_{V_{IDXji}(t)}}{V_{STOji}}}{V_{STOji}!}\Delta t \tag{8.1}$$

.

The new equation changes the number of multiplication operations from $2MR$ to $(R+1)T$. Given that $2M > T$, the number of operations will necessarily be reduced. Also, the greater the difference between $2M$ and T, the greater the gain obtained by applying the V substitution system.

8.1.0.1 *X Substitution System*

Similar to the method presented in the last section, we will build a substitution system to calculate $S \cdot v$. Let X_{STO} and X_{IDX} be two matrices of size $M \times P + 1$ where M is the number of molecules in the system and P is the maximum number of reactions a molecule participates in the reaction system.

Each line in matrix $\boldsymbol{X}_{STO}$, starting at the second one, will be filled with the stoichiometry in reach reaction that the correspondent molecule participates. The

entries not used in each row, as well as those from the first row, will be filled with the value zero.

Each line in matrix $\boldsymbol{X}_{\text{IDX}}$, starting at the second row, will be filled with the indexes of the reactions that the correspondent molecule participates. The entries not used in each row, as well as those from the first row, will be filled with the value one (or zero depending on the initial index assumed).

Then, we can rewrite $\Delta x = S \cdot v$ as follows:

$$\Delta x_i = \sum_{j=1,2,...P} X_{\text{STO}ij} v_{X_{\text{IDX}ij}} \tag{8.2}$$

.

The new equation changes the number of multiplication and sum operations from MR to $(M+1)P$. Given that $R \geq P$, the greater the difference between R and P, the greater the gain obtained by applying the X substitution system.

8.2 COMPUTING TIME BENCHMARK

One of the main constraints of the Stochastic Simulation Algorithm is its computational cost. The algorithm performs only one reaction for each time step and, depending on the size of the system and the rate constants, it may demand a very large number of steps to achieve the desired simulation time. The τ-leaping approximation and some variations of it can reduce the number of steps in some cases, usually being adopted for bigger chemical systems.

Another limitation of SSA is that the way the algorithm is designed makes it difficult to be implemented for parallel environments. Although there are implementations of the SSA to be computed using many computer cores, or even using General-Purpose Graphics Processing Units (GP-GPUs) [110–112], they can only accelerate replicates of the same model, reducing the time to compute several trajectories, but still limited regarding the size of the chemical system. There are also implementations for computing the SSA in a discretized space [113] where the subsystems in each subspace can be performed in parallel.

One of the goals of the present simulation method is to be computationally scalable to a variety of chemical system sizes. By size, we mean both the total number of molecules and the number of reactions. We shall then compare, for some cases, the CBSA's computational expense against the SSA and some of its implementations, and also an ODE solver. [1] For the SSA, we use the implementations from three Python libraries: StochPy[2] [114], STEPS[3] [115], and GillesPy2[4] [116]. The ODE implementation is part of the latter library. We use pyOpenCL[5] to distribute computations of the CBSA on GPUs.

1 All simulations were performed using the same computational architecture: Linux Debian 10 on an Intel i7-5930K, 64GB of DDR4 RAM, and NVidia GTX 1650 4GB.
2 StochPy: http://stochpy.sourceforge.net/
3 StochPy: http://http://steps.sourceforge.net/
4 GillesPy2: https://pypi.org/project/gillespy2/
5 pyOpenCL: https://pypi.org/project/pyopencl/

Benchmark Model

We use as a benchmark model a diffusion process through a discrete toroidal square-lattice space of length L, as depicted in Fig. 8.2. In this model, reactions will represent transitions between neighbor sub-spaces with a diffusion coefficient k_{diff}. Therefore, a system with a space of length L yield $4L^2$ reactions and L^2 molecular species. The system will start with an initial amount of molecules in only one of the sub-spaces and they will diffuse with a rate k_d.

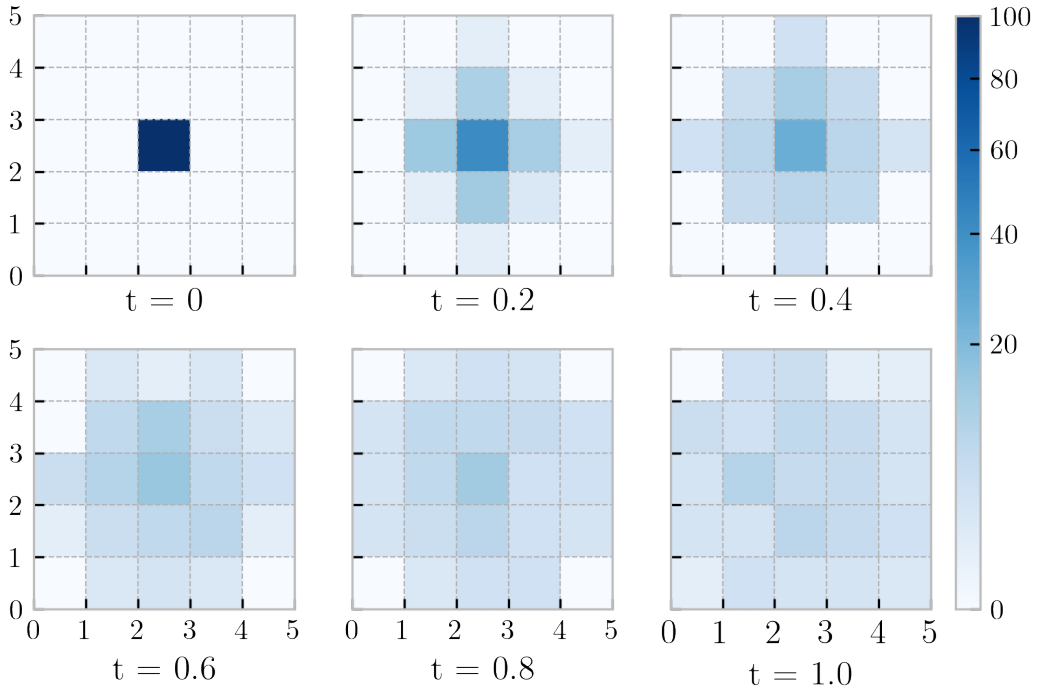

Figure 8.2: Benchmark using a diffusion model though a square-lattice discretized space with periodic boundary conditions. Simulation of the system with length $L = 5$ and 100 initial molecules using CBSA. Each plot depicts the distribution of the diffused molecules in the grid for the times $t = \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\}$.

Overall Performance Comparison

Given that the CBSA is designed to simulate large chemical systems, we first set two distinct scenarios using the diffusion model (a small and a large system) to compare the computational cost of the CBSA with other methods and implementations. The hypothesis is that the larger the system, the more impact the effect of matrix sparseness will have in reducing the CBSA computation time. For the small system case, we considered a system with $L = 2$, 10 initial molecules, and $k_d = 10.0$. For the large system we set $L = 16$, 10240 initial molecules, and $k_d = 1.0$. In Figure 8.3 we can see the mean computation time of 10 replicates for each of the considered methods in both small and large system cases. For the small system case, the sequential implementation of CBSA performed similarly to all SSA implementations except for STEPS. It is also observed that the τ-leaping implementations have

no improvement of performance for such cases. On the other hand, the GPU implementation of CBSA obtained the worst performance for the small system. The reason for such limited performance is that GPU computations have a computational overhead due to memory transfers between host and device and also a lower clock speed. However, when the system is large enough, the efficiency gained with the computation distributed through the hundreds of GPU's cores overcomes the memory transfer times making it marginal. This situation can be observed when considering the large system. Besides the higher efficiency of the sequential CBSA compared to the SSA methods, the gain in computation time is even better when using the GPU. For the large system, some methods were not able to conclude due to excessive use of the system's memory or reached the maximum time established.

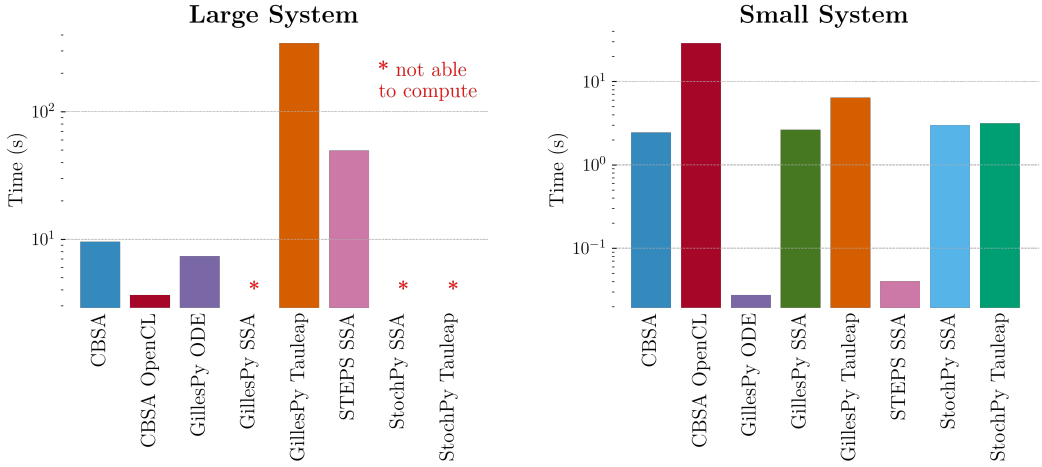

Figure 8.3: Simulation times in logarithmic scale for a small and a large system. The parallel implementation of the CBSA is denoted as "CBSA OpenCL". Considering that the CBSA is designed to simulate large systems, we selected two different scenarios where we estimate we could get the worst and the best performance of the algorithm when compared to the other considered methods. The small system has the parameters $L = 2$, 10 initial molecules, and $k_d = 10.0$. The large system has the parameters $L = 16$, 10240 initial molecules, and $k_d = 1.0$. In the case of the large system, the methods marked with a red star were not able to finish the simulations because of system memory shortage (specific to the machine used) or for reaching the maximum computation time of 10 minutes per replicate. All results shown are an average of 10 replicates.

Parameter Scaling Comparison

To better understand how each variable of the diffusion model affects computation times, we performed simulations varying the square-lattice length L, the initial number of molecules, the diffusion rate k_d, and the total simulation time. Regarding the system size, all the SSA implementations and the sequential CBSA showed an exponential growth in computing time as the value of L increased (Fig. 8.4a). On the other side, the GPU implementation of CBSA obtained a much lower growth rate in computational time as the size of the system grows, also performing better than all other methods when $L \geq 32$. Similar exponential growth in computation

time for all SSA implementations, except the StochPy Tauleap, was obtained when increasing the number of initial molecules in the system (Fig. 8.4b). However, the computational time is invariant to the number of molecular copies for both CBSA implementations as well as for the ODE solver.

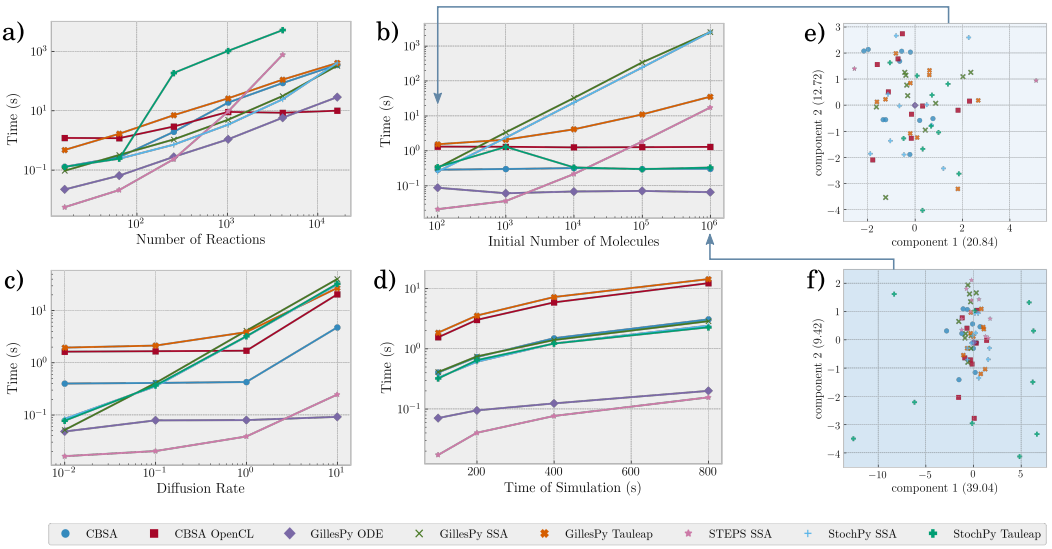

Figure 8.4: In **(a)**,**(b)**,**(c)**, and **(d)** the computing time for varying number of reactions, number of initial molecules, diffusion rate k_d, and total time of simulation are depicted respectively. The fixed parameters are $L = 4$, 100 initial molecules, $k_d = 0.1$, and total simulation time of 300 seconds. In **(e)** and **(f)** we plot the Principal Component Analysis for results obtained from simulations using 100 and 1,000,000 initial molecules in the system. Given that $L = 4$ used PCA to reduce the dimension of the vector of molecule counts x from 16 to 2 in order to be able to visualize the data. The variance explained in each axis is indicated in their respective labels.

It is expected in the SSA that the higher the reaction rates are, the more steps it takes to achieve a given simulation time. In Figure 8.4c we can observe that all SSA implementations present an exponential growth in the computation time as the diffusion constant is increased, with the τ-leaping implementations presenting a lower growth rate. The CBSA implementations only showed an increased computational time for the higher diffusion rate considered. This can happen when several reactions compete for the same reactant and the sum of their fluxes is higher than the available amount of the molecule. Thus, when several fast reactions share the same reactant, the CBSA needs to decrease the time step Δt implicating a higher number of steps to be computed in order to achieve a given simulation time. The variations in the time steps used in the CBSA are discussed in more detail in the next chapter. For all methods, the computing time has a linear relationship with the total time of the simulation (Fig. 8.4d).

Some Observations about the Simulation

It is worth mentioning that one of the reasons for the increased performance of STEPS' implementation in all the above-mentioned analyses is that it uses a limited

number of random numbers generated before the actual simulation. One effect of this implementation choice can be observed when analyzing the variance of the molecular counts at the end of the simulations. In Figures 8.4e and 8.4f we can see a Principal Component Analysis (PCA) where the final molecular counts of all simulations are projected in a lower dimension space for two different number of initial molecules in the system. Considering that the simulation using ODE can be equivalent in some cases to the mean of several runs of a stochastic method, we can observe that in both PCA plots that the solutions provided by CBSA and SSA are scattered around the single solution provided by the ODE solver, with a lower scattering when the number of molecules in the system is higher. However, in Figure 8.4e we can observe only 2 out of 10 points simulated by STEPS because they are overlapping. This result suggests that the limited number of random values has an impact on the stochasticity of the method, mostly when a lower number of molecules in the system is considered. It is also noticed that the solutions provided by the StochPy Tauleap are more scattered in Figure 8.4f than all other methods, which combined with the no variation in computation time might indicate an inconsistency in the solutions for a high number of molecules in this particular reaction system.

Although it is not the intent of this work to deliver a fully-featured software, both sequential and parallel implementations of CBSA, along with several other examples, are freely available at our GitHub repository[6].

6 CBSA: https://github.com/pauloburke/CBSA

9 THEORETICAL BIOCHEMICAL MODELS

To further investigate the correctness and characteristics of the simulation algorithm proposed in the last chapters, we present here more examples of biochemical systems. The content presented in this chapter is also available in [29].

9.1 TWO REACTIONS MODEL

Let us consider the set of two reactions described in {9.9}. The molecule A can be converted in a molecule B or C with reaction rates c_1 and c_2 respectively.

$$\begin{aligned} A &\xrightarrow{c_1} B \\ A &\xrightarrow{c_2} C \end{aligned} \qquad \{9.9\}$$

Figure 9.1 depicts several trajectories obtained by simulating this system using the CBSA. A very similar result is obtained using the SSA. The histogram under the plot shows the distribution of the time at which the number of molecules A reaches zero as it is consumed by both reactions. The histograms on the right show the distribution of the number of B and C molecules after 15 seconds.

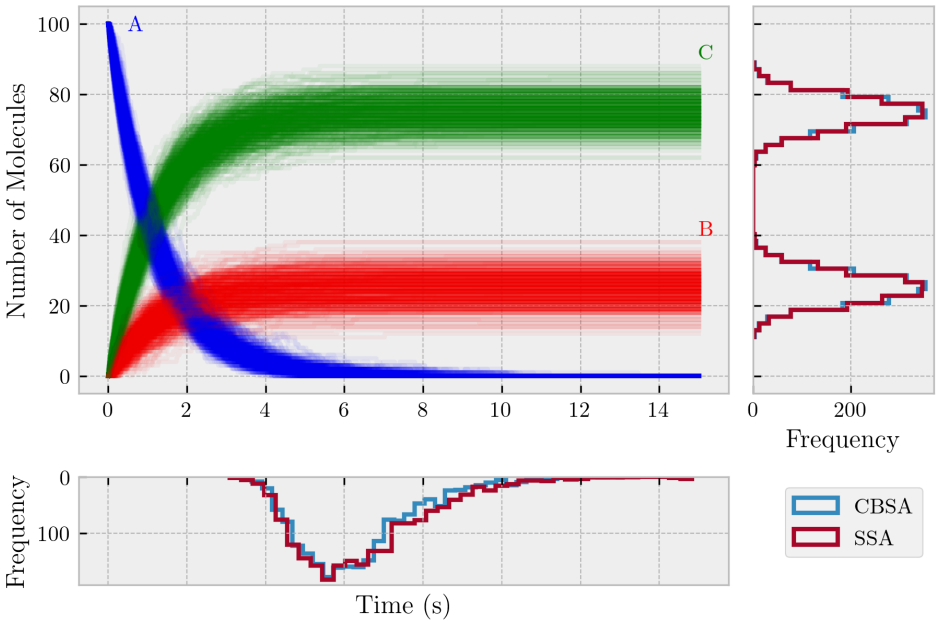

Figure 9.1: Simulation of the chemical system described in {9.9} using the CBSA.

9.2 N-REACTIONS MODEL

The CBSA first calculates the reaction fluxes as if they were independent. However, if two or more reactions share the same reactants, they can compete for them and their fluxes will no longer be dependent. Thus, if the amount requested of a given molecule by the reactions is bigger than the available amount, the set of calculated fluxes will not pass the mass-balance constraint. In such a case, the algorithm will search for a Δt that when multiplied by the fluxes will satisfy the constraint.

The search for valid fluxes is accomplished by multiplying the current Δt by a constant $0 < \alpha < 1$ until $v\Delta t$ meets the mass-balance constraint. The number of times that the multiplication by α is necessary in order to find a valid set of fluxes depends on the system and the system's current state, impacting the time demanded by each step in the simulation.

To better understand the impact of Δt and α on the simulations using the CBSA, consider the chemical system described in 9.10. The system contains $n + 1$ reactions where the first is a spontaneous creation of molecule A at a rate c_0 and the other n reactions will convert A into $B_1, B_2, ...B_n$ molecules with a rate c_1. For some combinations of n, c_0, c_1, and Δt the reactions will compete for A, leading the algorithm to search for a valid $v\Delta t$.

$$
\begin{aligned}
&\varnothing \xrightarrow{c_0} A \\
&A \xrightarrow{c_1} B_1 \\
&A \xrightarrow{c_1} B_2 \\
&\ldots \\
&A \xrightarrow{c_1} B_n
\end{aligned}
\qquad \{9.10\}
$$

We simulated the system described in {9.10} for different values of n. We also considered different values of Δt and α. Figure 9.2 shows some statistics from 50 simulation replicates for each combination of n, Δt, and α. Overall, we can observe that as n grows, the competition for A is intensified, implying a decreasing mean step length from the initial δt. It also implies a higher computational cost.

The reduction in the step length is due to the multiplications of Δt by α to find a valid solution. In Figure 9.3 we can see how many α-iterations the algorithm takes to find a valid solution given different initial Δt and α. As n grows, the number of α-iterations in each step of the algorithm increases, abbreviating the time-step taken. Therefore, more steps are required to achieve the desired stopping criterion.

Regarding the error in the simulation outputs, we compared the final number of $B_1, B_2, ...B_n$ molecules with an expected theoretical mean value. Except when $\Delta t = 1.0$, for almost all other combinations of parameters the error remained below 1%. To analyze the standard deviation of the final values, we compared them to the ones obtained with the SSA. Excepting when $\Delta t = 1.0$, all standard deviations obtained presented an error lower than 10%.

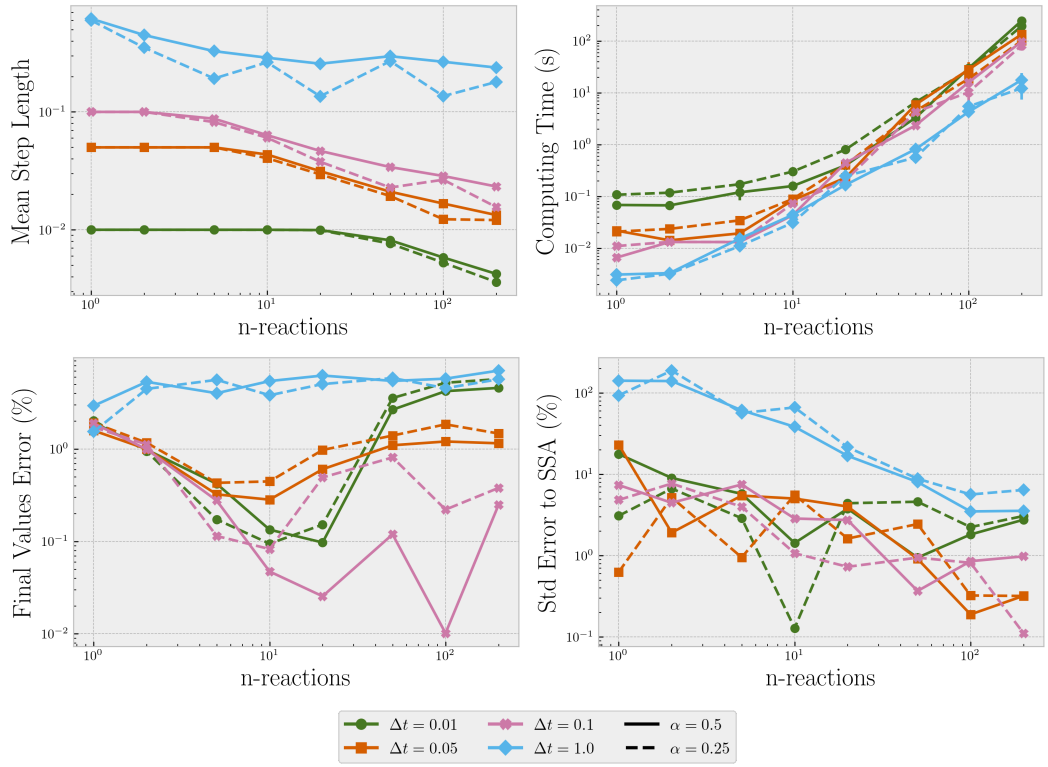

Figure 9.2: Analysis of mean step length, computing time, and relative errors from 50 simulation replicates of the system {9.10} for different number n of reactions. The stopping criterion was set as 10 seconds. The errors for both final values and standard deviations were calculated using the root-mean-square error (RMSE).

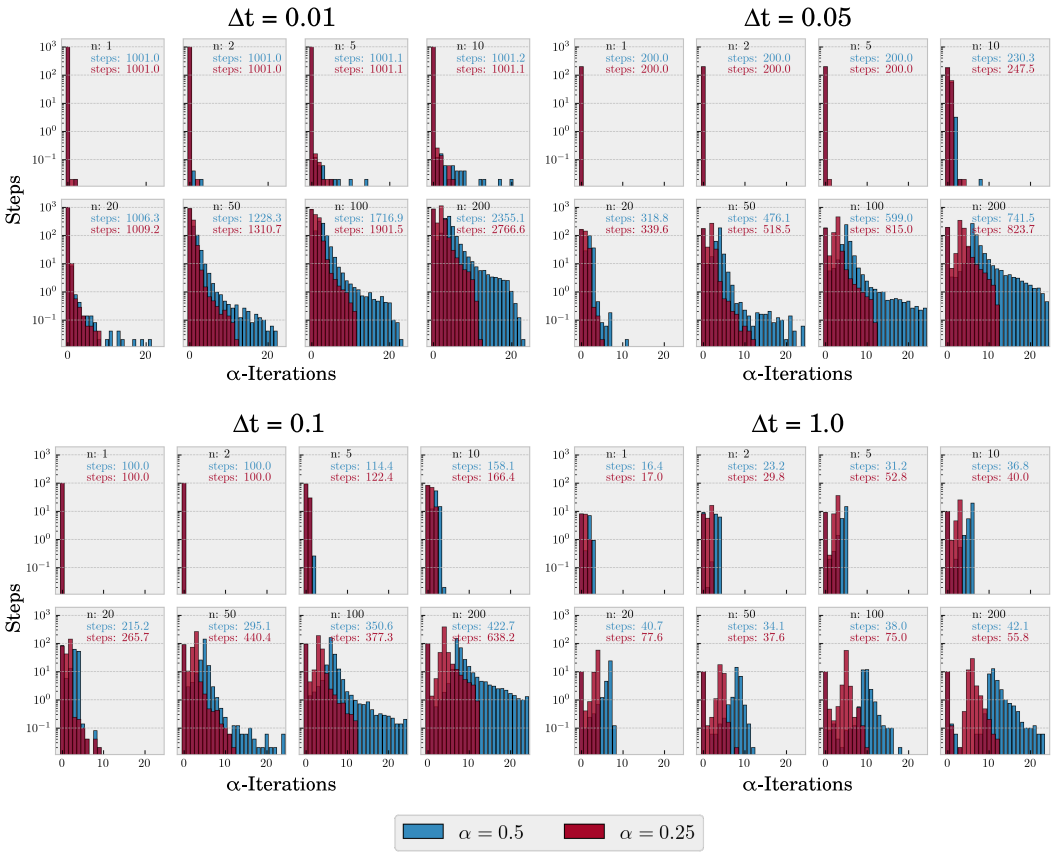

Figure 9.3: Histograms of the number of α-iterations that each step in the algorithm takes to find a valid solution.

9.3 THE OREGONATOR

In the original publication of the Stochastic Simulation Algorithm (SSA) by Daniel T. Gillespi [97], one of the models used to show the algorithm's capabilities was the Oregonator. This model is a chemical oscillator with closed limit cycles proposed by Field and Noyes [117] and is composed as follows:

$$\begin{aligned} X_1 + Y_2 &\xrightarrow{c_1} Y_1 \\ Y_1 + Y_2 &\xrightarrow{c_2} Z_1 \\ X_2 + Y_1 &\xrightarrow{c_3} 2Y_1 + Y_3 \\ 2Y_2 &\xrightarrow{c_4} Z_2 \\ X_3 + Y_3 &\xrightarrow{c_5} Y_2 \end{aligned} \tag{9.11}$$

.

Given that the number of X_1, X_2, and X_3 will be constant along time, we can rewrite the equations transferring their values to the reactions rates:

$$\begin{aligned} Y_2 &\xrightarrow{c_2 X_1} Y_1 \\ Y_1 + Y_2 &\xrightarrow{c_2} \varnothing \\ Y_1 &\xrightarrow{c_3 X_2} 2Y_1 + Y_3 \\ 2Y_2 &\xrightarrow{c_4} \varnothing \\ Y_3 &\xrightarrow{c_5 X_3} Y_2 \end{aligned} \tag{9.12}$$

.

In the same way as performed in [97], we rearranged the reaction rates as a function of the initial amounts of Y_1, Y_2, and Y_3 as well as two additional parameters ρ_1 and ρ_2:

$$\begin{aligned} c_1 X_1 &= \rho_1 / Y_{2s} \\ c_2 &= \rho_2 / Y_{1s} Y_{2s} \\ c_3 X_2 &= (\rho_1 + \rho_2) / Y_{1s} \\ c_4 &= 2\rho_1 / Y_{1s}^2 \\ c_5 X_3 &= (\rho_1 + \rho_2) / Y_{3s} \end{aligned} \tag{9.1}$$

.

The reaction system described in {9.12} can be written as a constraint-based model in terms of S and $R*$ as follows:

$$S = \begin{array}{c} \begin{matrix} r_1 & r_2 & r_3 & r_4 & r_5 \end{matrix} \\ \begin{bmatrix} 1 & -1 & 1 & -2 & 0 \\ -1 & -1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & -1 \end{bmatrix} \end{array} \begin{matrix} Y_1 \\ Y_2 \\ Y_3 \end{matrix} \tag{9.2}$$

$$R^* = \begin{array}{c} \begin{matrix} r_1 & r_2 & r_3 & r_4 & r_5 \end{matrix} \\ \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array} \begin{matrix} Y_1 \\ Y_2 \\ Y_3 \end{matrix} \tag{9.3}$$

.

In Figure 9.4 we can observe that the CBSA can reproduce the results shown in [97]. The system presents a well-defined limit cycle after a small period of convergence.

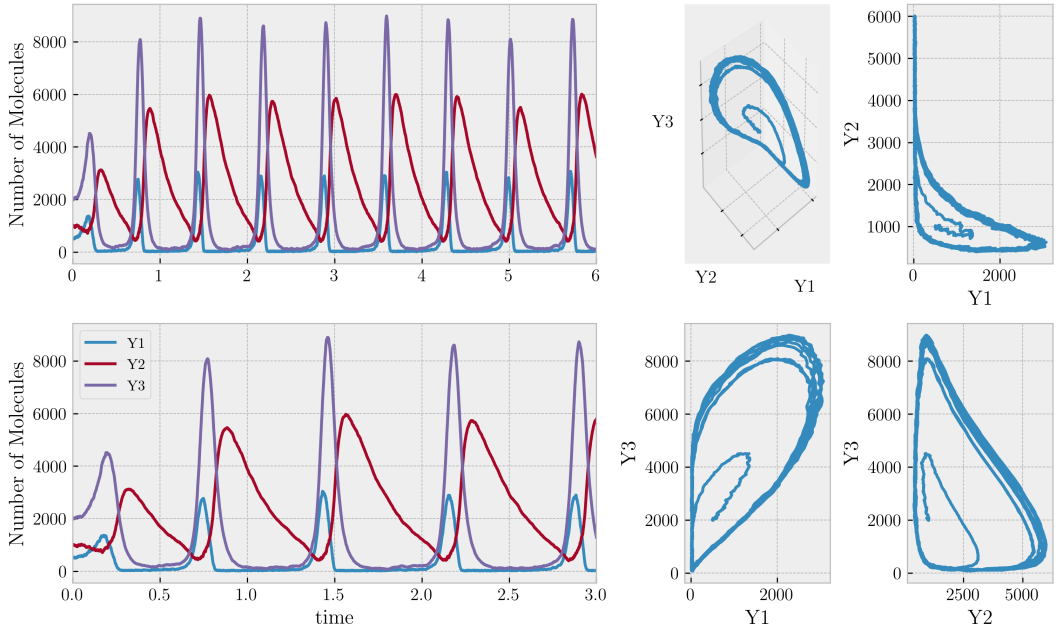

Figure 9.4: Simulation of the Oregonator using the CBSA with the following parameters: $Y_{1s} = 500$,$Y_{2s} = 1000$,$Y_{3s} = 2000$, $\rho_1 = 2000$, $\rho_2 = 50000$, $alpha = 0.5$, and $\Delta t = 0.001$.

10 REAL BIOCHEMICAL MODELS

In this chapter, we use the CBSA to simulate a real biochemical system and validate the results with respect to experimental data. The model is computationally challenging to simulate and we show that the use of CBSA enables the development of a more detailed version of the model.

10.1 BI-STABLE GENETIC SWITCH

One of the goals of Synthetic Biology is to modify living organisms in order to better control their behavior. In 2014, Jerala *et al.* managed to insert an artificial genetic circuit into a human cell so that it could produce two different fluorescent proteins as a response to the presence of two particular activator molecules in the environment [118]. Additionally to the response to the signal, the cells could also retain the memory of their state even when the activator molecule was no longer in the medium or change their state if the signal molecule was changed. The state of the cells could be visually checked through fluorescence microscopy.

The kind of behavior described above can be called a bi-stable switch. The system has two stable states and can transit between each other given a respective stimulus. It was implemented in [118] by employing positive and negative feedback loops in the expression of genetic constructs that contain the genetic code for receptors to signal molecules, promoter enhancers and inhibitors, and the fluorescent reporter molecules. The genetic circuit scheme is depicted in Figure 10.1.

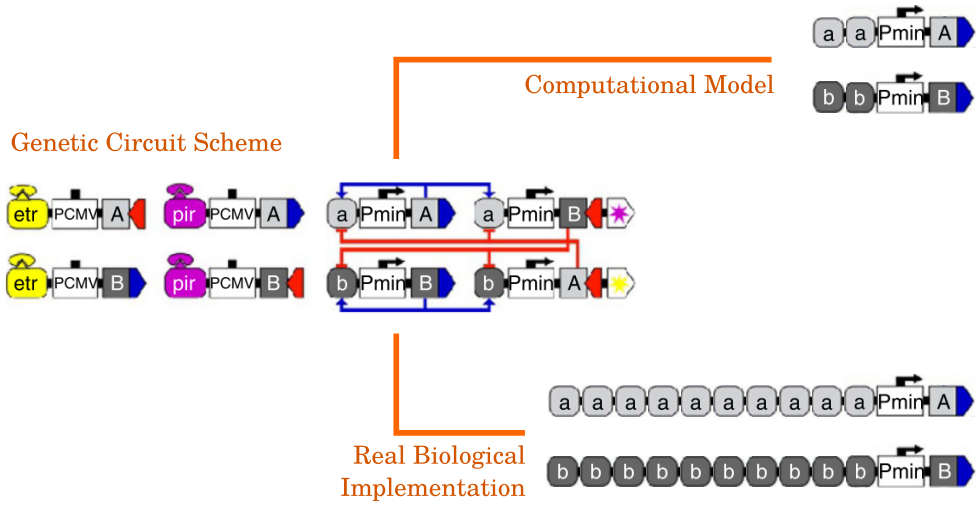

Figure 10.1: This genetic circuit implements a biological bi-stable switch by employing two opposite positive and negative feedback loops [118]. Genetic scheme figures adapted by permission from Springer Nature Customer Service Centre GmbH: Springer Nature, Nature Communications [118] (2014)

The presence of the PI (pir in the figure) molecule triggers the production of A enhancers and B inhibitors. These last molecules will compete to bind the "a" promoter sites. The higher the number of enhancers bound to the promoter sites, the faster is the production of A enhancers, B inhibitors, and the reporter molecule BFP (pink star). The opposite is achieved when the molecule ER (etr in the figure). It will trigger the production of B enhancers, A inhibitors, and the mCT reporter molecule (yellow star). Therefore, the presence of either PI or ER molecules can change the cell state to produce BFP or mCT respectively.

Prior to the experimental implementation, they developed a computational model of this system. However, the computational model is simpler than the actual genetic circuit constructed. In Figure 10.1 we show how the promoter constructs are composed in the real system and the computational model. Instead of the 10 binding sites for the regulation molecules, only two were considered in the computational model. It is because the number of binding pattern combinations produced by the two different regulators in the system grows fast with the number of binding sites.

10.2 SIMULATION OF THE SIMPLIFIED MODEL

Here we reproduce the simulations of the simplified computational model of the bi-stable genetic switch using the CBSA. The set of reactions and reaction rates were obtained from Tables 15 and 7 respectively from the Supplementary Material of [118]. The model comprises a total of 28 molecular species and 46 reactions.

In Figure 10.2 we can see how the levels of the GFP and mCitrine (mCT) reporter molecules change in different scenarios, with or without the presence of the pristinamycin (PI) or erythromycin (ER) inducer molecules. Each curve corresponds to an independent simulation which can be interpreted as an individual cell.

The results obtained using the CBSA are consistent with the simulations and experimental data available in [118]. The presence of PI and ER induces the production of BFP and mCT respectively. The removal of the inducers after a certain period does not affect the production of the respective reporters evidencing the system's memory. The change of ER to PI triggers a change of state in the cell which gradually stops producing mCT and begins to produce BFP.

It was shown experimentally in [118], but not considering simulations, that cells treated with no inducer tend randomly to one of the two states due to random fluctuations in the expression rates of enhancers and inhibitors. The same behavior could be observed in the simulations shown in Figure 10.2.

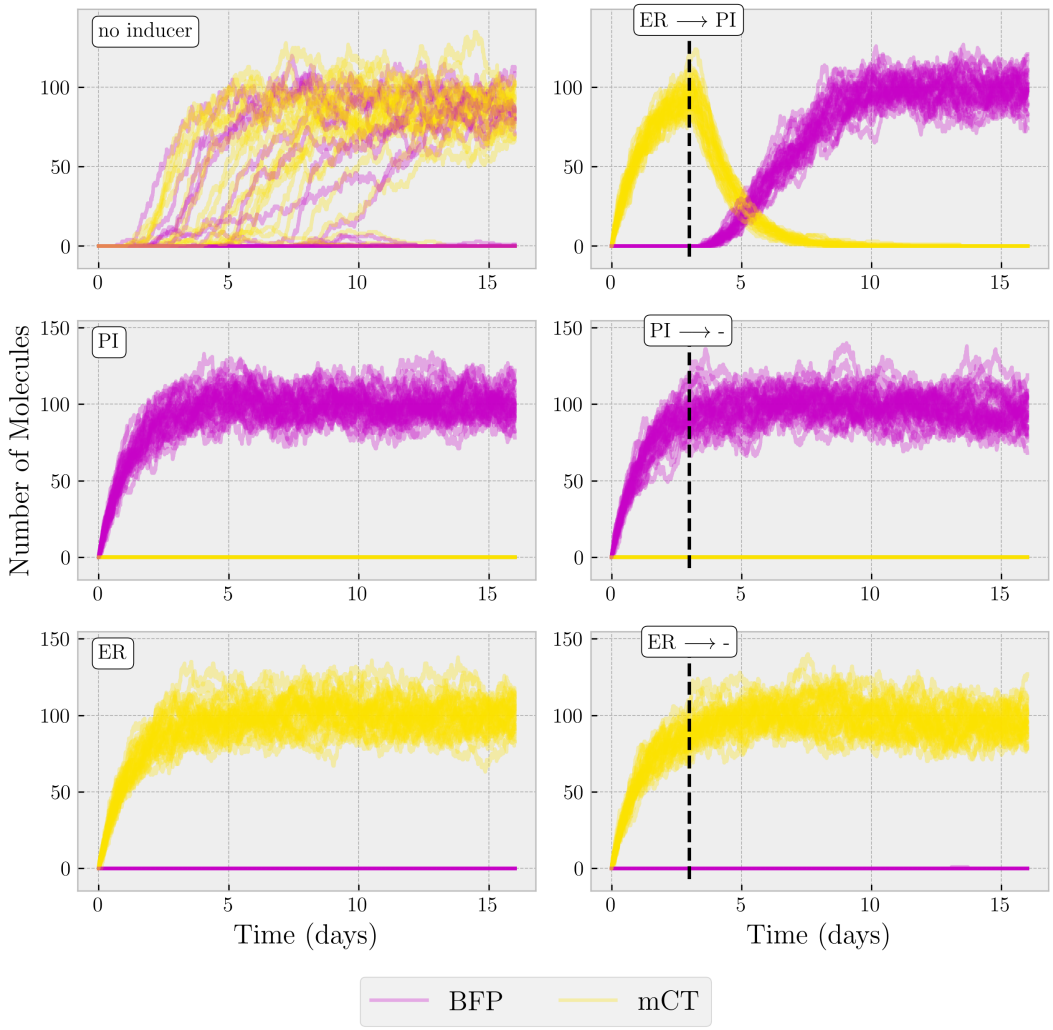

Figure 10.2: Simulations of the bi-stable genetic switch using the CBSA. The black dashed lines indicate the moment when the inducer present in the environment was changed or removed. Each curve corresponds to one of 30 independent simulations, each one taking approximately 17 ± 3 minutes to compute using $\alpha = 0.5$ and $\Delta t = 1.0$ second.

10.3 INCREASINGLY DETAILED MODELS

Given that we have enough knowledge about the real system, we can then scale up the simplified model provided in [118] aiming at producing a more comprehensive description of the biomolecular interactions. We produced intermediary representations of the system considering from 1 to 10 binding sites. Figure 10.3 shows network representations of the model considering two and ten binding sites.

In Figure 10.4 we replicate one of the experiments described in [118] where ER is introduced in the medium first. It triggers the production of A inhibitors and B enhancers, making the cells produce the mCT reporter. After 3h hours, ER is removed and PI is added. In response, cells start producing B inhibitors and A enhancers, thus stopping the production of mCT and starting to produce the BFP reporter. Trajectories obtained by simulating the system considering 2 and 10

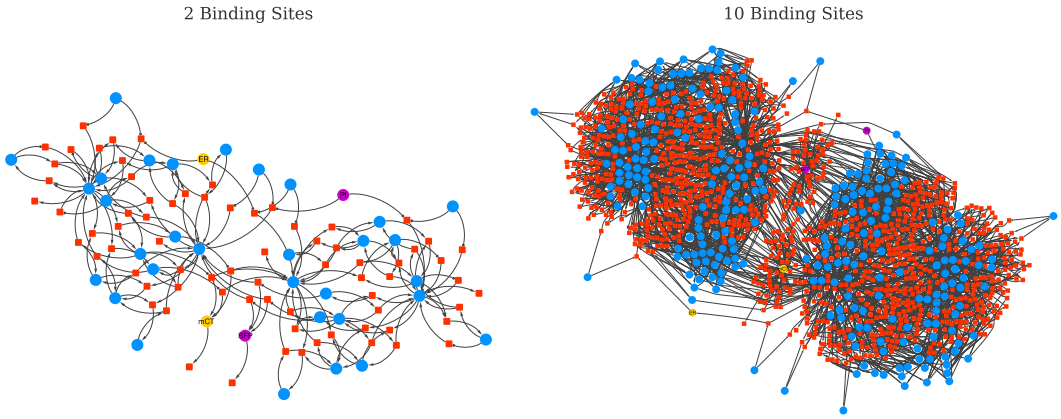

Figure 10.3: Network representations of the bi-stable genetic switch model considering 2 and 10 binding sites. The ER and mCT are represented in yellow and the PI and BFP are represented in pink.

binding sites are shown in the figure. We can observe that the point where the levels of mCT and BFP cross each other is closer to the experimental point in the 10 binding sites version.

We tracked some of the costs associated with the simulations, such as the number of molecules and reactions represented in the system and the time to compute each simulation. We also obtained their error when compared to the experimental data using the Root-Mean-Square Error (RMSE).

The costs and errors obtained from the simulation of each the number of binding sites considered can be observed in Figure 10.4 in the plot on the top-right. It is shown the normalized values for the number of molecular species, reaction channels, computing time, and error obtained from experimental data for the simulation of models considering from 1 to 10 binding sites. In the most detailed considered model, there are 276 molecules and 1114 reactions represented and the simulations took 46 hours and 45 minutes to complete in average [1]. It can be observed that the error to the experimental data obtained in the simulations decreases as the size of the models and the computational cost grows. Although this pattern could be expect, it is only a preliminary result. More models should be evaluated in order to check to what extent this relationship is true.

1 All simulations were performed using the same computational architecture: Linux Debian 10 on an Intel i7-5930K, 64GB of DDR4 RAM

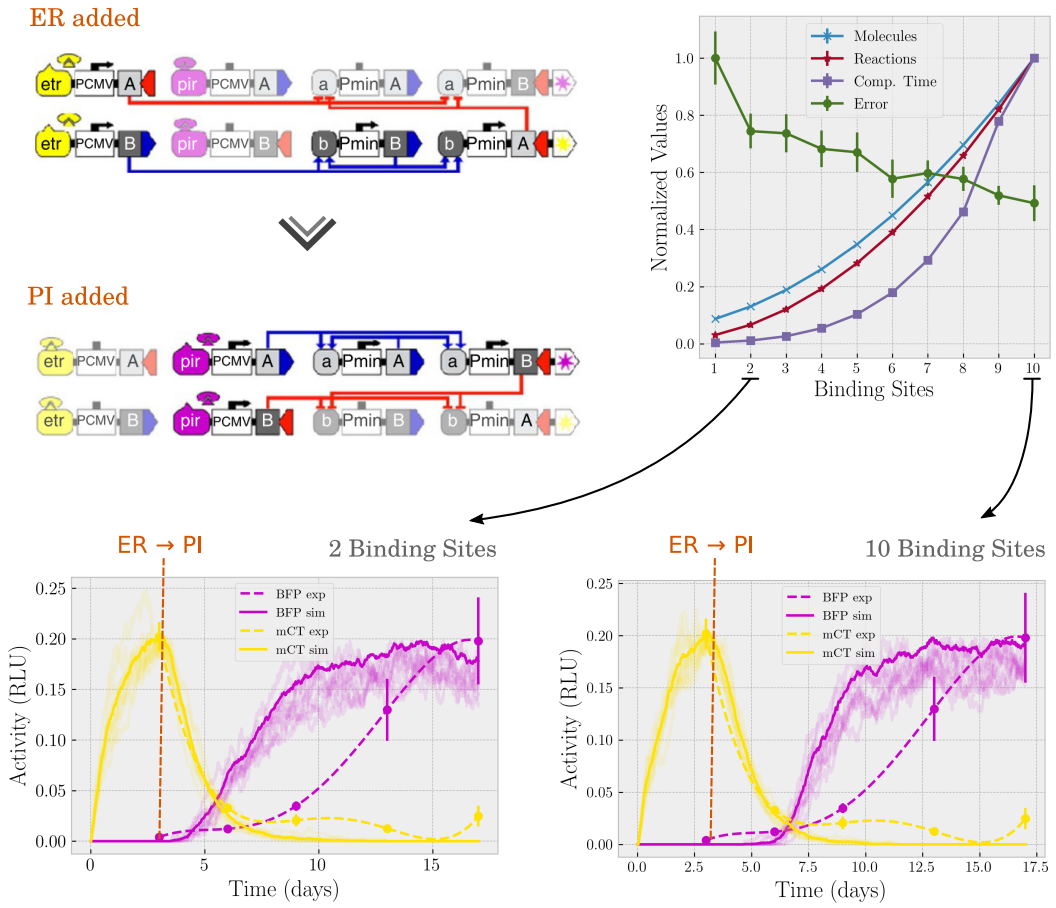

Figure 10.4: Simulation of the bi-stable genetic switch model considering several levels of detail. All simulations were performed using the CBSA parameters $\alpha = 0.5$ and $\Delta t = 1.0$ second. Genetic scheme figures adapted by permission from Springer Nature Customer Service Centre GmbH: Springer Nature, Nature Communications [118] (2014)

11

CONCLUSIONS AND FUTURE WORKS

Computational models of biomolecular systems have been successfully employed in the development of several applied research areas, such as Synthetic Biology [118], Bioengineering [102], and Precision Medicine [119–122]. This has been boosted by the advancements in data-generation technologies allied to the ever-growing availability of computational power [123]. Despite important advancements in the synergy between computational models and applied biomolecular sciences, current models are still often limited to represent only subsystems of cells. Additionally, the methodologies used to model and simulate these systems are highly heterogeneous.

Important advances have been made towards more comprehensive and integrated biomolecular models of cells. These models, which now achieve the scale of whole-cells, are believed to be one of the main prospects for future computational-aided biological research [124, 125]. However, such models still approach cellular subsystems as separated entities using heterogeneous modeling and simulation methodologies to evaluate their dynamics. The integration is achieved by exchanging information from time to time between subsystems as they are simulated separately.

In this work, we proposed modeling and simulation methodologies to enable more homogeneous representations of integrated biomolecular subsystems of cells. We built on the assumption that all cellular systems can be understood by their underlying biochemical interactions and that their integration can naturally occur if they share common molecular species. Therefore, we proposed a modeling framework based on networks to enable such homogeneous and integrated representations, and put it into test by modeling whole-cell scale data about the *Mycoplasma genitalium* organism.

Once having a homogeneous representation of different biochemical processes, we aimed at developing a simulation method that could account for the variety of molecular concentrations, interaction types, and reaction velocities that can be found within the intracellular environment and modeled using the proposed framework. To do so, we employed constraint-based modeling to obtain convenient mathematical representations of the network models and proposed a simulation method that draws concepts from the Stochastic Simulation Algorithm (SSA) and the Dynamic Flux Balance Analysis (dFBA). The so proposed simulation method, called Constraint-Based Simulation Algorithm, has been verified to account for important characteristics of biochemical systems such as discrete-valued molecular counts and stochasticity of the reaction velocities. Also, the algorithm was designed in such a way that it can be computed in high-performance parallel architectures such as GP-GPUs thus enabling efficient simulations of large systems. We explored the correctness and capabilities of the CBSA by simulating several theoretical models and one real biochemical model.

The next two sections elaborate further on the conclusions about the results obtained with the biochemical network modeling framework and the Constraint-

Based Simulation method. Despite the results reported in this work, much still remains to be done in order to achieve whole-cell simulations by applying the CBSA to whole-cell scale biochemical networks. In Section 11.3, we elaborate on the future developments that can be made from where this work stands, aiming at the ultimate goal of simulating a whole-cell. In the last section we enumerate the scientific literature publications which resulted from this work.

11.1 BIOCHEMICAL NETWORK MODELING

In the Chapters 4 and 5, a framework was presented for integrating cellular processes at a whole-cell scale, using rule-based modeling in a broader context. Besides the incorporation of network-modeled processes, such as metabolism, we were also capable to model processes that are not usually represented as networks, such as replication, transcription, and translation. We applied the framework directives to model whole-cell scale information about the *Mycoplasma genitalium* organism stored in specialized databases aiming to probe the capabilities of the framework. The obtained whole-cell biochemical network accounted for a great variety of molecules and cellular structures, as well as the interactions between them covering almost all processes known in the organism [28].

Many are the applications of whole-cell biochemical networks. In bioengineering, for example, it could provide more extensive biochemical interaction maps of given organisms, serving as a tool to better manipulate them. Also, the overlap between whole-cell biochemical networks of interacting organisms could provide insights into their relationship at the molecular level. Whole-cell biochemical networks can also pave the way to more comprehensive complex networks-based investigations, where we could study whole organisms through the topology of their biochemical interactions in a broader sense.

Despite the promising applications of the framework, the current scripts are limited to read data only from the WholeCellKB. Though impressive amounts of biological data continue to be generated, they tend to flow into relatively specific analyses and databases. The next step would be to develop software capable of searching these databases, integrate information from different sources, and therefore provide a more comprehensive and automated approach to whole-cell modeling. Some initiatives are already heading in the direction of aggregating available data [126], making it usable for simulation purposes, and performing community-based development and validation of whole-cell models [22]. In this sense, our framework could pave the way for community-based modeling, where experts in different cellular processes can speak the same modeling language.

At any extent, the *M. genitalium* organism remains a suitable model for whole-cell simulations, and its study has a medical interest as a consequence of its pathogenic nature. Therefore, the so obtained whole-cell biochemical network can provide useful information for the previously mentioned research fields, as can be further enhanced with the emergence of new data.

11.2 SIMULATION OF BIOCHEMICAL SYSTEMS

In Chapter 7, we proposed a simulation method called Constraint-Based Simulation Algorithm – CBSA, which employs constraint-based modeling of chemical reactions in order to simulate large systems efficiently. We showed with theoretical examples that it could provide solutions similar to those produced by the most used method, the Stochastic Simulation Algorithm. It also yielded particularly good performance when simulating several system configurations, especially for a high number of molecules and reactions, thus proving to be a computationally scalable method [29].

Besides the use of the CBSA to simulate biochemical systems modeled using the framework also proposed in this work, we may point out other possible outcomes from this method. The so employed constraint-based modeling is widely used in the representation of biochemical systems. It should then be straightforward to apply CBSA on the simulation of virtually any model available in databases such as BiGG Models [127], and BioModels [128]. The practical implications of its use on real models should be assessed in future works. Furthermore, considering that the presented method was conceived with large biochemical models in mind, it can be a useful tool to investigate more realistic representations of currently available models as tackled in Section 10.3.

11.3 FUTURE DEVELOPMENTS

The modeling framework and the simulation method proposed in this work were developed to work together as a major tool to enable the representation and simulation of large integrated biochemical systems. However, much has to be done in order to simulate whole-cell scale systems using these methodologies.

With the modeling framework, we were able to map together all the processes known about a specific organism. However, this model still has no comprehensive information about reaction kinetics and rate constants for instance. To obtain any dynamic information about the *M. genitalium* by applying the CBSA, we first need to determine the rates of each reaction in the model. Similar to the process of modeling homogenization, an approach to obtain homogeneous data regarding reaction rates still needs to be developed. Databases such as the WholeCellKB already have information about the kinetics of most of the reactions in the model. However, it should be carefully assessed in order to be compatible with the templates developed in this work.

Additionally to the problem of reaction kinetics, any simulation needs an initial state of the system to start with. This is not a trivial task when dealing with thousands of molecular species. Given that we can consider several types of molecules, such as DNA, RNA, Proteins, and metabolites, several different experimental approaches are needed to obtain information about each of these groups of molecules. For instance, transcriptomics can unveil RNA abundances while proteomics can estimate protein counts. On top of that, such approaches should all be applied in the context of single-cell analysis. This is because cells in a culture can be at different states and, if not analyzed individually, the mean of these states might not be a real state at all [129, 130].

Out of the whole-cell context, these methodologies could also be readily applied to very large subcellular systems, such as signaling processes in mammal cells, that are already a big challenge to simulate. The stochastic characteristic of the CBSA can be of much value to understanding heterogeneity in single-cell analyses of such systems generated by intrinsic cellular noise [121, 130].

11.4 PUBLICATIONS

Literature production resulted from the doctoral work.

Peer-Reviewed Journal Publications

Burke, Paulo E. P.; Campos, Claudia B. de L.; Costa, Luciano da F. ; Quiles, Marcos G.; A biochemical network modeling of a whole-cell. *Scientific Reports*, 2020.

Pre-Print Publications

Burke, Paulo E. P.; Costa, Luciano da F.; Accelerated Simulation of Large Reaction Systems Using a Constraint-Based Algorithm. *BioRxiv*, 2020.

Participation in conferences and symposiums

Burke, Paulo EP, and Luciano da F. Costa. "Towards Homogeneous Modeling and Simulation of Whole-Cells", *Inteligent Systems for Molecular Biology*, Basel, Switzerland (2019).

Burke, Paulo EP, and Luciano da F. Costa. "Simulation of Biochemical Systems Using Constraint-Based Methods and Complex Networks", *Workshop Anual do Programa de Pós-Graduação em Bioinformática da USP*, São Paulo - SP, Brazil (2019).

Burke, Paulo EP, and Luciano da F. Costa. "Simulation of Biochemical Systems Using Constraint-Based Methods and Complex Networks", *Conference on Complex Systems*, Thessaloniki, Greece (2018).

Burke, Paulo EP, and Luciano da F. Costa. "Dynamic Constraint-Based Methods on Complex Networks", *Workshop Anual do Programa de Pós-Graduação em Bioinformática da USP*, São Paulo - SP, Brazil (2017).

BIBLIOGRAPHY

[1] Zoltán N. Oltvai and Albert László Barabási. "Systems biology: Life's complexity pyramid." In: *Science* 298.5594 (2002), pp. 763–764. ISSN: 00368075. DOI: 10.1126/science.1078563.

[2] Jeremy M Berg, John L Tymoczko, and Lubert Stryer. *Biochemistry*. 5th ed. New York: W H Freeman, 2002.

[3] Luciano da Fontoura Costa. *Quantifying Complexity (CDT-6)*. Tech. rep. May. 2019. DOI: 10.13140/RG.2.2.29278.08000.

[4] Marko Gosak, Rene Markovič, Jurij Dolenšek, Marjan Slak Rupnik, Marko Marhl, Andraž Stožer, and Matjaž Perc. "Network science of biological systems at different scales: A review." In: *Physics of Life Reviews* 24 (2018), pp. 118–135. ISSN: 15710645. DOI: 10.1016/j.plrev.2017.11.003.

[5] Kieran Smallbone and Pedro Mendes. "Large-Scale Metabolic Models: From Reconstruction to Differential Equations." In: *Industrial Biotechnology* 9.4 (2013), pp. 179–184. ISSN: 1550-9087. DOI: 10.1089/ind.2013.0003.

[6] Thomas Schlitt and Alvis Brazma. "Current approaches to gene regulatory network modelling." In: *BMC Bioinformatics* 8.SUPPL. 6 (2007), S9. ISSN: 14712105. DOI: 10.1186/1471-2105-8-S6-S9.

[7] Miles MacLeod and Nancy J. Nersessian. "Coupling simulation and experiment: The bimodal strategy in integrative systems biology." In: *Studies in History and Philosophy of Science Part C: Studies in History and Philosophy of Biological and Biomedical Sciences* 44.4 (2013), pp. 572–584. ISSN: 13698486. DOI: 10.1016/j.shpsc.2013.07.001.

[8] Noël Malod-Dognin, Julia Petschnigg, Sam F. L. Windels, Janez Povh, Harry Hemmingway, Robin Ketteler, and Nataša Pržulj. "Towards a data-integrated cell." In: *Nature Communications* 10.1 (2019), p. 805. ISSN: 2041-1723. DOI: 10.1038/s41467-019-08797-8.

[9] Nayana G. Bhat and S. Balaji. "Whole-Cell Modeling and Simulation: A Brief Survey." In: *New Generation Computing* 38.1 (2020), pp. 259–281. ISSN: 18827055. DOI: 10.1007/s00354-019-00066-y.

[10] Jonathan R. Karr, Jayodita C. Sanghvi, Derek N. MacKlin, Miriam V. Gutschow, Jared M. Jacobs, Benjamin Bolival, Nacyra Assad-Garcia, John I. Glass, and Markus W. Covert. "A whole-cell computational model predicts phenotype from genotype." In: *Cell* 150.2 (2012), pp. 389–401. ISSN: 00928674. DOI: 10.1016/j.cell.2012.05.044.

[11] Jan Schellenberger, Junyoung O. Park, Tom M. Conrad, and Bernhard T. Palsson. "BiGG: A Biochemical Genetic and Genomic knowledgebase of large scale metabolic reconstructions." In: *BMC Bioinformatics* 11.1 (2010), p. 213. ISSN: 14712105. DOI: 10.1186/1471-2105-11-213.

[12] Lily A. Chylek, Leonard A. Harris, Chang Shung Tung, James R. Faeder, Carlos F. Lopez, and William S. Hlavacek. "Rule-based modeling: A computational approach for studying biomolecular site dynamics in cell signaling systems." In: *Wiley Interdisciplinary Reviews: Systems Biology and Medicine* 6.1 (2014), pp. 13–36. ISSN: 19395094. DOI: 10.1002/wsbm.1245.

[13] Lucian Smith, Darren Wilkinson, Michael Hucka, Frank Bergmann, Stefan Hoops, Sarah Keating, Sven Sahle, and James Schaff. *The Systems Biology Markup Language (SBML): Language Specification for Level 3 Version 1 Core*. 2010. DOI: 10.1038/npre.2010.4959.1.

[14] Dagmar Waltemath et al. "Toward Community Standards and Software for Whole-Cell Modeling." In: *IEEE Transactions on Biomedical Engineering* 63.10 (2016), pp. 2007–2014. ISSN: 15582531. DOI: 10.1109/TBME.2016.2560762.

[15] Stefan Hoops, Sven Sahle, Ralph Gauges, Christine Lee, J. Pahle, Natalia Simus, Mudita Singhal, Liang Xu, Pedro Mendes, and Ursula Kummer. "COPASI–a COmplex PAthway SImulator." In: *Bioinformatics* 22.24 (2006), pp. 3067–3074. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btl485.

[16] Thomas G. Kurtz. "The relationship between stochastic and deterministic models for chemical reactions." In: *The Journal of Chemical Physics* 57.7 (1972), pp. 2976–2978. ISSN: 00219606. DOI: 10.1063/1.1678692.

[17] Daniel T. Gillespie. "Stochastic simulation of chemical kinetics." In: *Annual Review of Physical Chemistry* 58.1 (2007), pp. 35–55. ISSN: 0066426X. DOI: 10.1146/annurev.physchem.58.032806.104637. arXiv: 0808.3863.

[18] Radhakrishnan Mahadevan, Jeremy S. Edwards, and Francis J. Doyle. "Dynamic Flux Balance Analysis of diauxic growth in Escherichia coli." In: *Biophysical Journal* 83.3 (2002), pp. 1331–1340. ISSN: 00063495. DOI: 10.1016/S0006-3495(02)73903-9.

[19] Jong Min Lee, Erwin P. Gianchandani, James A. Eddy, and Jason A. Papin. "Dynamic analysis of integrated signaling, metabolic, and regulatory networks." In: *PLoS Computational Biology* 4.5 (2008). ISSN: 1553734X. DOI: 10.1371/journal.pcbi.1000086.

[20] Peter L. Freddolino and Saeed Tavazoie. "The dawn of virtual cell biology." In: *Cell* 150.2 (2012), pp. 248–250. ISSN: 10974172. DOI: 10.1016/j.cell.2012.07.001.

[21] Masaru Tomita et al. "E-CELL: Software environment for whole-cell simulation." In: *Bioinformatics* 15.1 (1999), pp. 72–84. ISSN: 13674803. DOI: 10.1093/bioinformatics/15.1.72.

[22] Arthur P. Goldberg, Balázs Szigeti, Yin Hoon Chew, John AP Sekar, Yosef D. Roth, and Jonathan R. Karr. "Emerging whole-cell modeling principles and methods." In: *Current Opinion in Biotechnology* 51 (2018), pp. 97–102. ISSN: 18790429. DOI: 10.1016/j.copbio.2017.12.013. arXiv: 1710.02431.

[23] Derek N. Macklin et al. "Simultaneous cross-evaluation of heterogeneous E. coli datasets via mechanistic simulation." In: *Science* 369.6502 (2020). ISSN: 10959203. DOI: 10.1126/science.aav3751.

[24] Ulrike Muenzner, Edda Klipp, and Marcus Krantz. "A comprehensive, mechanistically detailed, and executable model of the Cell Division Cycle in Saccharomyces cerevisiae." In: *Nature communications* 10.1 (2019), p. 1308.

[25] Oliver Purcell, Bonny Jain, Jonathan R. Karr, Markus W. Covert, and Timothy K. Lu. "Towards a whole-cell modeling approach for synthetic biology." In: *Chaos* 23.2 (2013). ISSN: 10541500. DOI: 10.1063/1.4811182.

[26] Jayodita C. Sanghvi, Sergi Regot, Silvia Carrasco, Jonathan R. Karr, Miriam V. Gutschow, Benjamin Bolival, and Markus W. Covert. "Accelerated discovery via a whole-cell model." In: *Nature Methods* 10.12 (2013), pp. 1192–5. ISSN: 15487091. DOI: 10.1038/nmeth.2724. arXiv: 9605103 [cs].

[27] Joshua Rees-Garbutt, Oliver Chalkley, Sophie Landon, Oliver Purcell, Lucia Marucci, and Claire Grierson. "Designing minimal genomes using whole-cell models." In: *Nature Communications* 11.1 (2020), p. 836. ISSN: 2041-1723. DOI: 10.1038/s41467-020-14545-0.

[28] Paulo E.P. Burke, Claudia B.de L. Campos, Luciano da F. Costa, and Marcos G. Quiles. "A biochemical network modeling of a whole-cell." In: *Scientific Reports* 10.1 (2020), p. 13303. ISSN: 20452322. DOI: 10.1038/s41598-020-70145-4.

[29] Paulo E.P. Burke and Luciano da F. Costa. "Accelerated simulation of large reaction systems using a constraint-based algorithm." In: *bioRxiv* (2020), p. 2020.10.31.362442. ISSN: 26928205. DOI: 10.1101/2020.10.31.362442.

[30] M. P. Crosland. "The use of diagrams as chemical 'equations' in the lecture notes of William Cullen and Joseph Black." In: *Annals of Science* 15.2 (1959), pp. 75–90. ISSN: 1464505X. DOI: 10.1080/00033795900200088.

[31] John. McMurry, Robert C Fay, and Jill K Robinson. *Chemistry*. English. 7th ed. Pearson, 2015, p. 492. ISBN: 9780321943170 0321943171 9781292092867 1292092866.

[32] S. W. Benson. *Chemical Kinetics*. English. Vol. 242. 5400. New York: Harper & Row, 1973, p. 587. ISBN: 0060438622 9780060438623. DOI: 10.1038/242587a0.

[33] Mark Newman. *Networks: An Introduction*. 2010, pp. 1–784. ISBN: 9780191594175. DOI: 10.1093/acprof:oso/9780199206650.001.0001. arXiv: 1212.2425.

[34] William S. Bowie. "Applications of graph theory in computer systems." In: *International Journal of Computer & Information Sciences* 5.1 (1976), pp. 9–31. ISSN: 00917036. DOI: 10.1007/BF00991069.

[35] L. Da F. Costa, F. A. Rodrigues, G. Travieso, and P. R.Villas Boas. "Characterization of complex networks: A survey of measurements." In: *Advances in Physics* 56.1 (2007), pp. 167–242. ISSN: 00018732. DOI: 10.1080/00018730601170527. arXiv: 0505185 [cond-mat].

[36] E. N. Gilbert. "Random Graphs." In: *The Annals of Mathematical Statistics* 30.4 (1959), pp. 1141–1144. ISSN: 0003-4851. DOI: 10.1214/aoms/1177706098.

[37] Albert-László Barabási and Eric Bonabeau. "Scale-Free Networks." In: *Scientific American* 288.5 (2003), pp. 60–69. ISSN: 0036-8733. DOI: 10.1038/scientificamerican0503-60.

[38] M G Quiles, E R Zorzal, and E E N Macau. "A dynamical model for community detection in complex networks." In: *Neural Networks (IJCNN), The 2013 International Joint Conference on* (2013), pp. 1–8. DOI: 10.1109/IJCNN.2013.6706944.

[39] Réka Albert. "Scale-free networks in cell biology." In: *Journal of cell science* 118.21 (2005), pp. 4947–4957. ISSN: 0021-9533. DOI: 10.1242/jcs.02714. arXiv: 0510054 [q-bio].

[40] H. Jeong, B. Tombor, R. Albert, Z. N. Oltval, and A. L. Barabásl. "The large-scale organization of metabolic networks." In: *Nature* 407.6804 (2000), pp. 651–654. ISSN: 00280836. DOI: 10.1038/35036627. arXiv: 0010278 [cond-mat].

[41] Albert-László Barabási and Zoltán N. Oltvai. "Network biology: understanding the cell's functional organization." In: *Nature Reviews Genetics* 5.2 (2004), pp. 101–113. ISSN: 1471-0056. DOI: 10.1038/nrg1272. arXiv: NIHMS150003.

[42] H. Kitano. "Systems Biology: A Brief Overview." In: *Science* 295.5560 (2002), pp. 1662–1664. ISSN: 00368075. DOI: 10.1126/science.1069492.

[43] Frank J. Bruggeman and Hans V. Westerhoff. "The nature of systems biology." In: *Trends in Microbiology* 15.1 (2007), pp. 45–50. ISSN: 0966842X. DOI: 10.1016/j.tim.2006.11.003.

[44] Rafael Silva-Rocha and Víctor de Lorenzo. "Noise and Robustness in Prokaryotic Regulatory Networks." In: *Annual review of microbiology* 64.1 (2010), pp. 257–275. ISSN: 0066-4227. DOI: 10.1146/annurev.micro.091208.073229.

[45] Marc Vidal, Michael E. Cusick, and Albert László Barabási. "Interactome networks and human disease." In: *Cell* 144.6 (2011), pp. 986–998. ISSN: 00928674. DOI: 10.1016/j.cell.2011.02.016.

[46] Jean-François Rual et al. "Towards a proteome-scale map of the human protein–protein interaction network." In: *Nature* 437.7062 (2005), pp. 1173–1178. ISSN: 0028-0836. DOI: 10.1038/nature04209.

[47] Damian Szklarczyk et al. "The STRING database in 2017: quality-controlled protein-protein association networks, made broadly accessible." In: *Nucleic acids research* 45.D1 (2017), pp. D362–D368. ISSN: 13624962. DOI: 10.1093/nar/gkw937.

[48] A Sancar, L A Lindsey-Boltz, K Unsal-Kaçmaz, and S Linn. "Molecular mechanisms of mammalian DNA repair and the DNA damage checkpoints." In: *Annu Rev Biochem* 73.1 (2004), pp. 39–85. ISSN: 0066-4154. DOI: 10.1146/annurev.biochem.73.011303.073723.

[49] Jack T. Zilfou and Scott W. Lowe. "Tumor suppressive functions of p53." In: *Cold Spring Harbor perspectives in biology* 1.5 (2009), a001883–a001883. ISSN: 19430264. DOI: 10.1101/cshperspect.a001883.

[50] Giovanni Chillemi, Sebastian Kehrloesser, Francesca Bernassola, Alessandro Desideri, Volker Dötsch, Arnold J. Levine, and Gerry Melino. "Structural evolution and dynamics of the p53 proteins." In: *Cold Spring Harbor Perspectives in Medicine* 7.4 (2017), pp. 1–16. ISSN: 21571422. DOI: 10.1101/cshperspect.a028308.

[51] Patricia A J Muller and Karen H. Vousden. *Mutant p53 in cancer: New functions and therapeutic opportunities*. 2014. DOI: 10.1016/j.ccr.2014.01.021.

[52] Chung Jung Tsai, Buyong Ma, and Ruth Nussinov. "Protein-protein interaction networks: how can a hub protein bind so many different partners?" In: *Trends in Biochemical Sciences* 34.12 (2009), pp. 594–600. ISSN: 09680004. DOI: 10.1016/j.tibs.2009.07.007.

[53] Francesco Ceccarelli, Francesco Ceccarelli, Denes Turei, Denes Turei, Attila Gabor, Attila Gabor, Julio Saez-Rodriguez, and Julio Saez-Rodriguez. "Bringing data from curated pathway resources to Cytoscape with OmniPath." In: *Bioinformatics* 36.8 (2020). Ed. by Peter Robinson, pp. 2632–2633. ISSN: 14602059. DOI: 10.1093/bioinformatics/btz968.

[54] Paul Shannon, Andrew Markiel, Owen Ozier, Nitin S. Baliga, Jonathan T. Wang, Daniel Ramage, Nada Amin, Beno Schwikowski, and Trey Ideker. "Cytoscape: A software Environment for integrated models of biomolecular interaction networks." In: *Genome Research* (2003). ISSN: 10889051. DOI: 10.1101/gr.1239303.

[55] J. Dedrick Jordan, Emmanuel M. Landau, and Ravi Iyengar. "Signaling networks: The origins of cellular multitasking." In: *Cell* 103.2 (2000), pp. 193–200. ISSN: 00928674. DOI: 10.1016/S0092-8674(00)00112-4.

[56] William S. Hlavacek and James R. Faeder. "The complexity of cell signaling and the need for a new mechanics." In: *Science Signaling* 2.81 (2009), pp. 1–4. ISSN: 19450877. DOI: 10.1126/scisignal.281pe46.

[57] John J. Tyson and Béla Novák. "Functional Motifs in Biochemical Reaction Networks." In: *Annual Review of Physical Chemistry* 61.1 (2010), pp. 219–240. ISSN: 0066-426X. DOI: 10.1146/annurev.physchem.012809.103457. arXiv: NIHMS150003.

[58] Natalie C. Duarte, Scott A. Becker, Neema Jamshidi, Ines Thiele, Monica L. Mo, Thuy D. Vo, Rohith Srivas, and Bernhard Palsson. "Global reconstruction of the human metabolic network based on genomic and bibliomic data." In: *Proceedings of the National Academy of Sciences of the United States of America* 104.6 (2007), pp. 1777–1782. ISSN: 00278424. DOI: 10.1073/pnas.0610772104.

[59] Daniel Machado, Markus J. Herrgård, and Isabel Rocha. "Stoichiometric Representation of Gene–Protein–Reaction Associations Leverages Constraint-Based Analysis from Reaction to Gene-Level Phenotype Prediction." In: *PLoS Computational Biology* 12.10 (2016). Ed. by Kiran Raosaheb Patil, e1005140. ISSN: 15537358. DOI: 10.1371/journal.pcbi.1005140.

[60] Jonathan R. Karr, Jayodita C. Sanghvi, Derek N. Macklin, Abhishek Arora, and Markus W. Covert. "WholeCellKB: Model organism databases for comprehensive whole-cell models." In: *Nucleic Acids Research* 41.D1 (2013), pp. D787–92. ISSN: 03051048. DOI: 10.1093/nar/gks1108.

[61] John I. Glass, Nacyra Assad-Garcia, Nina Alperovich, Shibu Yooseph, Matthew R. Lewis, Mahir Maruf, Clyde A. Hutchison, Hamilton O. Smith, and J. Craig Venter. "Essential genes of a minimal bacterium." In: *Proceedings of the National Academy of Sciences of the United States of America* 103.2 (2006), pp. 425–430. ISSN: 00278424. DOI: 10.1073/pnas.0510013103.

[62] C. M. Fraser et al. "The Minimal Gene Complement of Mycoplasma genitalium." In: *Science* 270.5235 (1995), pp. 397–404. ISSN: 0036-8075. DOI: 10.1126/science.270.5235.397.

[63] Paul Stothard and David S. Wishart. *Circular genome visualization and exploration using CGView*. 2005. DOI: 10.1093/bioinformatics/bti054.

[64] Raya Khanin and Ernst Wit. "How scale-free are biological networks." In: *Journal of computational biology : a journal of computational molecular cell biology* 13.3 (2006), pp. 810–818. ISSN: 1066-5277. DOI: 10.1089/cmb.2006.13.810.

[65] Paolo Crucitti, Vito Latora, and Massimo Marchiori. "Model for cascading failures in complex networks." In: *Physical Review E* 69.4 (2004), p. 045104. ISSN: 1539-3755. DOI: 10.1103/PhysRevE.69.045104.

[66] Wen-xu Wang and Guanrong Chen. "Universal robustness characteristic of weighted networks against cascading failure." In: *Physical Review E* 77.2 (2008), p. 026101. ISSN: 1539-3755. DOI: 10.1103/PhysRevE.77.026101.

[67] Ashley G Smart, Luis A N Amaral, and Julio M Ottino. "Cascading failure and robustness in metabolic networks." In: *Proceedings of the National Academy of Sciences of the United States of America* 105.36 (2008), pp. 13223–8. ISSN: 1091-6490. DOI: 10.1073/pnas.0803571105.

[68] Xuqing Huang, Irena Vodenska, Shlomo Havlin, and H. Eugene Stanley. "Cascading failures in bi-partite graphs: Model for systemic risk propagation." In: *Scientific Reports* 3 (2013), p. 13. ISSN: 20452322. DOI: 10.1038/srep01219. arXiv: 1210.4973.

[69] Ney Lemke, Fabiana Herédia, Cláudia K. Barcellos, Adriana N. dos Reis, and José C M Mombach. "Essentiality and damage in metabolic networks." In: *Bioinformatics* 20.1 (2004), pp. 115–119. ISSN: 13674803. DOI: 10.1093/bioinformatics/btg386.

[70] Zeba Wunderlich and Leonid A. Mirny. "Using the Topology of Metabolic Networks to Predict Viability of Mutant Strains." In: *Biophysical Journal* 91.6 (2006), pp. 2304–2311. ISSN: 00063495. DOI: 10.1529/biophysj.105.080572.

[71] H. B. Fraser. "Evolutionary Rate in the Protein Interaction Network." In: *Science* 296.5568 (2002), pp. 750–752. ISSN: 00368075. DOI: 10.1126/science.1068696.

[72] Daniel Yasumasa Takahashi, João Ricardo Sato, Carlos Eduardo Ferreira, and André Fujita. "Discriminating Different Classes of Biological Networks by Analyzing the Graphs Spectra Distribution." In: *PLoS ONE* 7.12 (2012). ISSN: 19326203. DOI: 10.1371/journal.pone.0049949. arXiv: 1208.2976.

[73] Bin Huang, Mingyang Lu, Dongya Jia, Eshel Ben-Jacob, Herbert Levine, and Jose N. Onuchic. "Interrogating the topological robustness of gene regulatory circuits by randomization." In: *PLoS Computational Biology* 13.3 (2017), pp. 1–21. ISSN: 15537358. DOI: 10.1371/journal.pcbi.1005456.

[74] Nicolas Le Novère. "Quantitative and logic modelling of molecular and gene networks." In: *Nature Reviews Genetics* 16 (2015), p. 146.

[75] Venkata Swamy Martha, Zhichao Liu, Li Guo, Zhenqiang Su, Yanbin Ye, Hong Fang, Don Ding, Weida Tong, and Xiaowei Xu. "Constructing a robust protein-protein interaction network by integrating multiple public databases." In: *BMC Bioinformatics* 12.SUPPL. 10 (2011), S7. ISSN: 14712105. DOI: 10.1186/1471-2105-12-S10-S7.

[76] Shirin Taghipour, Peyman Zarrineh, Mohammad Ganjtabesh, and Abbas Nowzari-Dalini. "Improving protein complex prediction by reconstructing a high-confidence protein-protein interaction network of Escherichia coli from different physical interaction data sources." In: *BMC Bioinformatics* 18.1 (2017), p. 10. ISSN: 14712105. DOI: 10.1186/s12859-016-1422-x.

[77] Damian Szklarczyk et al. "STRING v11: Protein-protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets." In: *Nucleic Acids Research* 47.D1 (2019), pp. D607–D613. ISSN: 13624962. DOI: 10.1093/nar/gky1131.

[78] Markus W. Covert, Christophe H. Schilling, and Bernhard Palsson. "Regulation of gene expression in flux balance models of metabolism." In: *Journal of Theoretical Biology* 213.1 (2001), pp. 73–88. ISSN: 00225193. DOI: 10.1006/jtbi.2001.2405.

[79] Ana Paula Oliveira, Christina Ludwig, Paola Picotti, Maria Kogadeeva, Ruedi Aebersold, and Uwe Sauer. "Regulation of yeast central metabolism by enzyme phosphorylation." In: *Molecular Systems Biology* 8.1 (2012). ISSN: 17444292. DOI: 10.1038/msb.2012.55.

[80] Yibo Wu et al. "Multilayered genetic and omics dissection of mitochondrial activity in a mouse reference population." In: *Cell* 158.6 (2014), pp. 1415–1430. ISSN: 10974172. DOI: 10.1016/j.cell.2014.07.039.

[81] Katsuyuki Yugi, Hiroyuki Kubota, Atsushi Hatano, and Shinya Kuroda. "Trans-Omics: How To Reconstruct Biochemical Networks Across Multiple 'Omic' Layers." In: *Trends in Biotechnology* 34.4 (2016), pp. 276–290. ISSN: 18793096. DOI: 10.1016/j.tibtech.2015.12.013.

[82] Antonio Fabregat et al. "The Reactome Pathway Knowledgebase." In: *Nucleic Acids Research* 46.D1 (2018), pp. D649–D655. ISSN: 13624962. DOI: 10.1093/nar/gkx1132. arXiv: NIHMS150003.

[83] Joshua J Hamilton and Jennifer L Reed. "Software platforms to facilitate reconstructing genome-scale metabolic networks." In: *Environmental microbiology* 16.1 (2014), pp. 49–59.

[84] Richard A. Notebaart, Frank H.J. van Enckevort, Christof Francke, Roland J. Siezen, and Bas Teusink. "Accelerating the reconstruction of genome-scale metabolic networks." In: *BMC Bioinformatics* 7.1 (2006), p. 296. ISSN: 14712105. DOI: 10.1186/1471-2105-7-296.

[85] B. Kholodenko, M. B. Yaffe, and W. Kolch. "Computational Approaches for Analyzing Information Flow in Biological Networks." In: *Science Signaling* 5.220 (2012), re1–re1. ISSN: 1945-0877. DOI: 10.1126/scisignal.2002961.

[86] Xi Zhuo Jiang, Muye Feng, Kai H. Luo, and Yiannis Ventikos. "Large-scale molecular dynamics simulation of flow under complex structure of endothelial glycocalyx." In: *Computers and Fluids* 173 (2018), pp. 140–146. ISSN: 00457930. DOI: 10.1016/j.compfluid.2018.03.014.

[87] Barbara Di Ventura, Caroline Lemerle, Konstantinos Michalodimitrakis, and Luis Serrano. "From in vivo to in silico biology and back." In: *Nature* 443.7111 (2006), pp. 527–533. ISSN: 0028-0836. DOI: 10.1038/nature05127.

[88] G. Wayne Brodland. "How computational models can help unlock biological systems." In: *Seminars in Cell \& Developmental Biology* 47-48 (2015), pp. 62–73. ISSN: 1084-9521. DOI: 10.1016/J.SEMCDB.2015.07.001.

[89] Sayuri K. Hahl and Andreas Kremling. "A comparison of deterministic and stochastic modeling approaches for biochemical reaction systems: On fixed points, means, and modes." In: *Frontiers in Genetics* 7.AUG (2016). ISSN: 16648021. DOI: 10.3389/fgene.2016.00157.

[90] Willi Gottstein, Brett G. Olivier, Frank J. Bruggeman, and Bas Teusink. "Constraint-based stoichiometric modelling from single organisms to microbial communities." In: *Journal of The Royal Society Interface* 13.124 (2016), p. 20160627. ISSN: 1742-5689. DOI: 10.1098/rsif.2016.0627.

[91] Antonio Fernández-Ramos, James A. Miller, Stephen J. Klippenstein, and Donald G. Truhlar. "Modeling the kinetics of bimolecular reactions." In: *Chemical Reviews* 106.11 (2006), pp. 4518–4584. ISSN: 00092665. DOI: 10.1021/cr050205w.

[92] Nobuyoshi Ishii, Yoshihiro Suga, Akiko Hagiya, Hisami Watanabe, Hirotada Mori, Masataka Yoshino, and Masaru Tomita. "Dynamic simulation of an in vitro multi-enzyme system." In: *FEBS Letters* 581.3 (2007), pp. 413–420. ISSN: 00145793. DOI: 10.1016/j.febslet.2006.12.049.

[93] Kirill Peskov, Ekaterina Mogilevskaya, and Oleg Demin. "Kinetic modelling of central carbon metabolism in Escherichia coli." In: *FEBS Journal* 279.18 (2012), pp. 3374–3385. ISSN: 1742464X. DOI: 10.1111/j.1742-4658.2012.08719.x.

[94] Donald A. McQuarrie. "Stochastic approach to chemical kinetics." In: *Journal of Applied Probability* 4.03 (1967), pp. 413–478. ISSN: 0021-9002. DOI: 10.2307/3212214.

[95] Daniel T. Gillespie. "A rigorous derivation of the chemical master equation." In: *Physica A: Statistical Mechanics and its Applications* 188.1-3 (1992), pp. 404–425. ISSN: 03784371. DOI: 10.1016/0378-4371(92)90283-V.

[96] Daniel T. Gillespie. "A general method for numerically simulating the stochastic time evolution of coupled chemical reactions." In: *Journal of Computational Physics* 22.4 (1976), pp. 403–434. ISSN: 10902716. DOI: 10.1016/0021-9991(76)90041-3.

[97] Daniel T. Gillespie. "Exact stochastic simulation of coupled chemical reactions." In: *Journal of Physical Chemistry* 81.25 (1977), pp. 2340–2361. ISSN: 00223654. DOI: 10.1021/j100540a008. arXiv: arXiv:1011.1669v3.

[98] D. T. Gillespie. "Approximate accelerated stochastic simulation of chemically reacting systems." In: *Journal of Chemical Physics* 115.4 (2001), pp. 1716–1733. ISSN: 00219606. DOI: 10.1063/1.1378322.

[99] Yang Cao, Daniel T. Gillespie, and Linda R. Petzold. "Efficient step size selection for the tau-leaping simulation method." In: *Journal of Chemical Physics* 124.4 (2006). ISSN: 00219606. DOI: 10.1063/1.2159468. arXiv: NIHMS150003.

[100] M R Watson. "Metabolic maps for the Apple II." In: *Biochemical Society Transactions* 12 (1984), pp. 1093–1094. ISSN: 03005127. DOI: 10.1042/bst0121093.

[101] Paula Jouhten, Marilyn Wiebe, and Merja Penttilä. "Dynamic flux balance analysis of the metabolism of Saccharomyces cerevisiae during the shift from fully respirative or respirofermentative metabolic states to anaerobiosis." In: *FEBS Journal*. Vol. 279. 18. 2012, pp. 3338–3354. ISBN: 1495614964. DOI: 10.1111/j.1742-4658.2012.08649.x.

[102] Verónica S. Martínez, Stefanie Dietmair, Lake-Ee Quek, Mark P. Hodson, Peter Gray, and Lars K. Nielsen. "Flux balance analysis of CHO cells before and after a metabolic switch from lactate production to consumption." In: *Biotechnology and Bioengineering* 110.2 (2013), pp. 660–666. ISSN: 00063592. DOI: 10.1002/bit.24728.

[103] Francesco Gatto, Heike Miess, Almut Schulze, and Jens Nielsen. "Flux balance analysis predicts essential genes in clear cell renal cell carcinoma metabolism." In: *Scientific Reports* 5 (2015), p. 10738. ISSN: 20452322. DOI: 10.1038/srep10738.

[104] Amit Varma and Bernhard O. Palsson. "Metabolic flux balancing: Basic concepts, scientific and practical use." In: *Bio/Technology* 12.10 (1994), pp. 994–998. ISSN: 0733222X. DOI: 10.1038/nbt1094-994. arXiv: arXiv:1011.1669v3.

[105] Jeffrey D. Orth, Ines Thiele, and Bernhard O. Ø. Palsson. "What is Flux Balance Analysis ?" In: *Nature Biotechnology* 28.3 (2010), pp. 245–248. ISSN: 10870156. DOI: 10.1038/nbt.1614. arXiv: NIHMS150003.

[106] Erwin P. Gianchandani, Arvind K. Chavali, and Jason A. Papin. "The application of flux balance analysis in systems biology." In: *Wiley Interdisciplinary Reviews: Systems Biology and Medicine* 2.3 (2010), pp. 372–382. ISSN: 19395094. DOI: 10.1002/wsbm.60.

[107] Elsa W. Birch, Madeleine Udell, and Markus W. Covert. "Incorporation of flexible objectives and time-linked simulation with flux balance analysis." In: *Journal of Theoretical Biology* 345 (2014), pp. 12–21. ISSN: 10958541. DOI: 10.1016/j.jtbi.2013.12.009. arXiv: NIHMS150003.

[108] Jong Min Lee, Erwin P. Gianchandani, James A. Eddy, and Jason A. Papin. "Dynamic analysis of integrated signaling, metabolic, and regulatory networks." In: *PLoS Computational Biology* 4.5 (2008). ISSN: 1553734X. DOI: 10.1371/journal.pcbi.1000086.

[109] Daniel T. Gillespie. "Chemical Langevin equation." In: *Journal of Chemical Physics* 113.1 (2000), pp. 297–306. ISSN: 00219606. DOI: 10.1063/1.481811.

[110] Ivan Komarov and Roshan M. D'Souza. "Accelerating the Gillespie Exact Stochastic Simulation Algorithm Using Hybrid Parallel Execution on Graphics Processing Units." In: *PLoS ONE* 7.11 (2012), pp. 1–9. ISSN: 19326203. DOI: 10.1371/journal.pone.0046693.

[111] Marco S. Nobile, Paolo Cazzaniga, Daniela Besozzi, Dario Pescini, and Giancarlo Mauri. "cuTauLeaping: A GPU-powered tau-leaping stochastic simulator for massive parallel analyses of biological systems." In: *PLoS ONE* 9.3 (2014). Ed. by Jean Peccoud, e91963. ISSN: 19326203. DOI: 10.1371/journal.pone.0091963.

[112] Kei Sumiyoshi, Kazuki Hirata, Noriko Hiroi, and Akira Funahashi. "Acceleration of discrete stochastic biochemical simulation using GPGPU." In: *Frontiers in Physiology* 6.FEB (2015), pp. 1–10. ISSN: 1664042X. DOI: 10.3389/fphys.2015.00042.

[113] Weiliang Chen and Erik de Schutter. "Parallel STEPS: Large scale stochastic spatial reaction-diffusion simulation with high performance computers." In: *Frontiers in Neuroinformatics* 11.February (2017), pp. 1–15. ISSN: 16625196. DOI: 10.3389/fninf.2017.00013.

[114] Timo R. Maarleveld, Brett G. Olivier, and Frank J. Bruggeman. "StochPy: A comprehensive, user-friendly tool for simulating stochastic biological processes." In: *PLoS ONE* (2013). ISSN: 19326203. DOI: 10.1371/journal.pone.0079345.

[115] Iain Hepburn, Weiliang Chen, Stefan Wils, and Erik De Schutter. "STEPS: Efficient simulation of stochastic reaction-diffusion models in realistic morphologies." In: *BMC Systems Biology* 6 (2012). ISSN: 17520509. DOI: 10.1186/1752-0509-6-36.

[116] John H. Abel, Brian Drawert, Andreas Hellander, and Linda R. Petzold. "GillesPy: A Python Package for Stochastic Model Building and Simulation." In: *IEEE Life Sciences Letters* 2.3 (2017), pp. 35–38. ISSN: 2332-7685. DOI: 10.1109/lls.2017.2652448.

[117] Richard J. Field and Richard M. Noyes. "Oscillations in chemical systems. IV. Limit cycle behavior in a model of a real chemical reaction." In: *The Journal of Chemical Physics* 60.5 (1974), pp. 1877–1884. ISSN: 00219606. DOI: 10.1063/1.1681288.

[118] Tina Lebar et al. "A bistable genetic switch based on designable DNA-binding domains." In: *Nature Communications* 5.1 (2014), p. 5007. ISSN: 20411723. DOI: 10.1038/ncomms6007.

[119] Michael Schubert, Bertram Klinger, Martina Klünemann, Anja Sieber, Florian Uhlitz, Sascha Sauer, Mathew J. Garnett, Nils Blüthgen, and Julio Saez-Rodriguez. "Perturbation-response genes reveal signaling footprints in cancer gene expression." In: *Nature Communications* 9.1 (2018), p. 20. ISSN: 20411723. DOI: 10.1038/s41467-017-02391-6.

[120] Federica Eduati, Patricia Jaaks, Jessica Wappler, Thorsten Cramer, Christoph A Merten, Mathew J Garnett, and Julio Saez-Rodriguez. "Patient-specific logic models of signaling pathways from screenings on cancer biopsies to prioritize personalized combination therapies." In: *Molecular Systems Biology* 16.2 (2020). ISSN: 1744-4292. DOI: 10.15252/msb.20188664.

[121] Christian H. Holland et al. "Robustness and applicability of transcription factor and pathway analysis tools on single-cell RNA-seq data." In: *Genome Biology* 21.1 (2020), p. 36. ISSN: 1474760X. DOI: 10.1186/s13059-020-1949-z.

[122] Aurelien Dugourd et al. "Causal integration of multi-omics data with prior knowledge to generate mechanistic hypotheses." In: *bioRxiv* (2020). ISSN: 26928205. DOI: 10.1101/2020.04.23.057893.

[123] Samuel J. Aronson and Heidi L. Rehm. *Building the foundation for genomics in precision medicine*. 2015. DOI: 10.1038/nature15816.

[124] Jonathan R. Karr, Koichi Takahashi, and Akira Funahashi. *The principles of whole-cell modeling*. 2015. DOI: 10.1016/j.mib.2015.06.004.

[125] John A.P. Sekar, Arthur P. Goldberg, Jonathan R. Karr, Balázs Szigeti, Yosef D. Roth, and Saahith C. Pochiraju. "A blueprint for human whole-cell modeling." In: *Current Opinion in Systems Biology* 7 (2017), pp. 8–15. ISSN: 24523100. DOI: 10.1016/j.coisb.2017.10.005.

[126] Daniel S Weaver, Ingrid M Keseler, Amanda Mackie, Ian T Paulsen, and Peter D Karp. "A genome-scale metabolic flux model of Escherichia coli K–12 derived from the EcoCyc database." In: *BMC systems biology* 8.1 (2014), p. 79.

[127] Philip Miller, Nathan E. Lewis, Andreas Dräger, Joshua A. Lerman, Zachary A. King, Stephen Federowicz, Bernhard O. Palsson, Justin Lu, and Ali Ebrahim. "BiGG Models: A platform for integrating, standardizing and sharing genome-scale models." In: *Nucleic Acids Research* 44.D1 (2015), pp. D515–D522. ISSN: 0305-1048. DOI: 10.1093/nar/gkv1049.

[128] Vijayalakshmi Chelliah et al. "BioModels: Ten-year anniversary." In: *Nucleic Acids Research* 43.D1 (2015), pp. D542–D548. ISSN: 13624962. DOI: 10.1093/nar/gku1181.

[129] Ruben Perez-Carrasco, Casper Beentjes, and Ramon Grima. "Effects of cell cycle variability on lineage and population measurements of messenger RNA abundance." In: *Journal of The Royal Society Interface* 17.168 (2020), p. 20200360. ISSN: 1742-5689. DOI: 10.1098/rsif.2020.0360.

[130] Tatiana Filatova, Nikola Popovic, and Ramon Grima. "Statistics of Nascent and Mature RNA Fluctuations in a Stochastic Model of Transcriptional Initiation, Elongation, Pausing, and Termination." In: *Bulletin of Mathematical Biology* 83.1 (2021), p. 3. ISSN: 15229602. DOI: 10.1007/s11538-020-00827-7.