Unifying metabolic networks, regulatory constraints, and resource allocation

Dissertation

Zur Erlangung des Grades

Doctor rerum naturalium

Fachbereich Mathematik und Informatik
Freie Universität Berlin

Lin Liu

Berlin, 2020

Betreuer: Prof. Dr. Alexander Bockmayr

Zweitgutachter: Dr. Hidde de Jong

Tag der Disputation: 16. April 2020

Contents

Abstract

Metabolic and gene regulatory networks are two classic models of systems biology. Biologically, gene regulatory networks are the control system of protein expression while metabolic networks, especially the genome-scale reconstructions consist of thousands of enzymatic reactions breaking down nutrients into precursors and energy to support the cellular survival. Metabolic-genetic networks, in addition, include the translational processes as an integrated model of classical metabolic networks and the gene expression machinery. Conversely, genetic regulation is also affected by the metabolic activities that provide feedbacks and precursors to the regulatory system. Thus, the two systems are highly interactive and depend on each other.

Up to now, various efforts have been made to bridge the two network types. Yet, the dynamic integration of metabolic networks and genetic regulation remains a major challenge in computational systems biology.

This PhD thesis is a contribution to mathematical modeling approaches for studying metabolic-regulatory systems. Inspired by *regulatory flux balance analysis* (rFBA), we first propose an analytic pipeline to explore the optimal solution space in rFBA. Then, our efforts focus on the dynamic combination of metabolic networks together with enzyme production costs and genetic regulation. For this purpose, we first explore the intuitive idea that incorporates Boolean regulatory rules while iterating *resource balance analysis*. However, with the iterative strategy, the gene expression states are only updated in discrete time steps. Furthermore, formalizing the *metabolic-regulatory networks* (MRNs) by hybrid automata provides a new mathematical framework that allows the quantitative integration of the metabolic-genetic network with the genetic regulation in a hybrid discrete-continuous system. For the application of this theoretical formalization, we develop a constraint-based approach *regulatory dynamic enzyme-cost flux balance analysis* (r-deFBA) as an optimal control strategy for the hybrid automata representing MRNs. This allows the prediction of optimal regulatory state transitions, dynamics of metabolism, and resource allocation capable of achieving a maximal biomass production over a time interval. Finally, this PhD project ends with a chapter on perspectives; we apply the theory of product automata to model the dynamics at population-level, integrating continuous metabolism and discrete regulatory states.

Chapter 1

Introduction

Since the days of Claude Shannon and Norbert Wiener, who introduced information theory (Shannon, 1948) and cybernetics (Wiener, 1948), the system-level understanding of biology using information and communication has gained much attention. People noticed that a biological system is not merely composed of different kinds of chemical molecules with a certain structure. Organisms, including human beings, operate like a machine based on the processing of information between different components such as genes and proteins (Quastler, 1953; Simon, 1991). This involves the interconnection of multiple biological parts and an enormous amount of information communicated as a huge network (Hartwell et al., 1999).

Yet, systematic studies were hindered by limited datasets in the early days. Computational systems biology did not come of age until the emergence of high throughput technologies, particularly in genomics (Sanger and Coulson, 1975; Maxam and Gilbert, 1977; Anderson, 1981) and proteomics (Aebersold and Mann, 2003). Rather than focusing on isolated molecules, systems biology reveals the fundamental mechanism of molecular interactions at a systematic level (Kitano, 2002; Westerhoff and Palsson, 2004). In the past, many formal mathematical frameworks have been introduced to understand biological systems, especially in relevance to the main topics of this thesis: cellular metabolism and genetic regulation.

In the conversion of nutrients into energy and building blocks to produce functional macromolecules, cellular metabolism implicates thousands of biochemical molecules that are interconnected with one another in a huge metabolic network. So is the genetic regulation evolved in parallel by nature to assure the cellular performance in response to a changing environment or stimulus. The gene regulatory network transfers the internal/external signals to the genes encoding enzymes of metabolic reactions via regulatory proteins or transcription factors. This functions as a control system for metabolism. Conversely, the genetic regulation is affected by metabolism as well. This is because the internal metabolism gives feedback to the regulatory system and provides precursors for regulatory proteins or transcription factors. Thus, the integration of cellular metabolism with genetic regulation

is indispensable for elucidating the molecular mechanisms of cells particularly the genotype-phenotype relationships. To date, however, the investigation of interactions between the metabolic network and the gene regulatory system, especially their dynamic interplay, remains a challenge because of the intricacies involved.

In the past, several formal mathematical frameworks have been developed to study genetic regulation and metabolism, either separately or integrated together. However, the dynamic integration of metabolic and gene regulatory networks is still under study. Thus, there is a strong need for more efficient computational algorithms and approaches that will allow understanding and predicting the metabolic-regulatory processes. For this purpose, this PhD thesis concentrates on developing computational methods to improve dynamic modeling of the integration of metabolism and genetic regulation.

Before we go on to the main part, some basics of this thesis are introduced next.

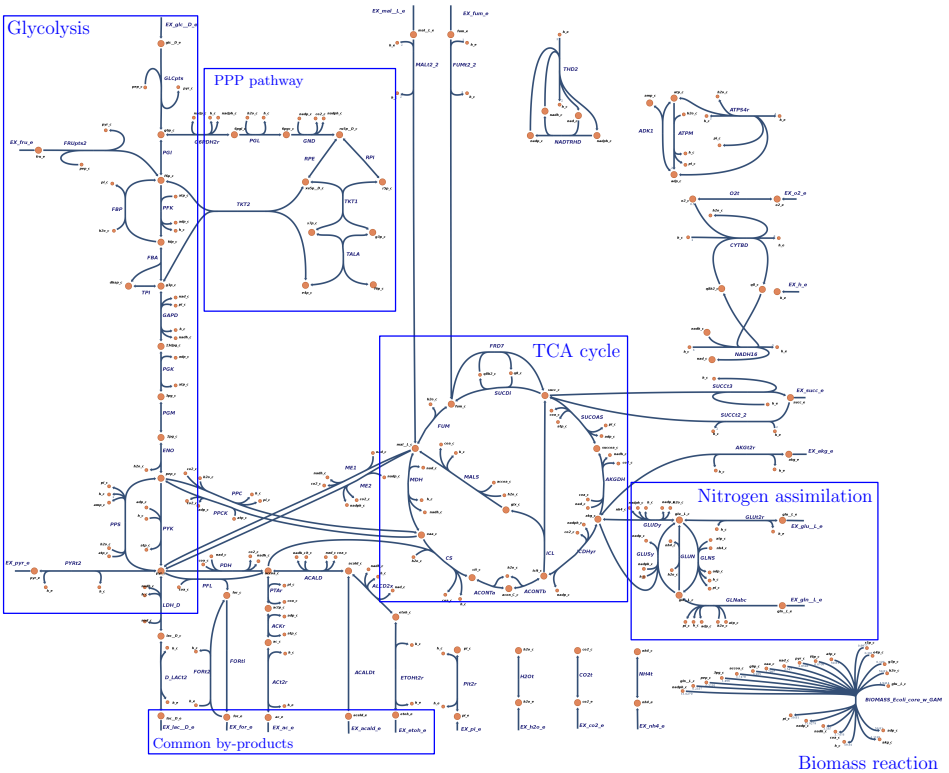


Figure 1.1: Central metabolic network of *E. coli* condensed via genome-scale model iAF1260 (Feist et al., 2007; Orth, 2010). The nodes represent the intracellular metabolites and edges denote the reactions. The key metabolic pathways are marked by annotations.

1.1 Metabolic networks

Cellular metabolism includes all enzymatic biochemical reactions in a cell, which can be classified as anabolism and catabolism, referring to the 'building up' and 'breaking down' of large molecules respectively. Mathematically, metabolism can be represented by a directed hypergraph called metabolic network (MN), with the metabolites as nodes and the reactions as hyperarcs.

With the flood of high-throughput datasets, hundreds of metabolic networks from microbes to plants, to human beings, have been successfully reconstructed based on the annotated genome and proteome (Förster et al., 2003a; de Oliveira Dal'Molin et al., 2010; Duarte et al., 2007). Many of them are available in the online databases, e.g., Biomodels (Le Novere et al., 2006). As an example, the directed hypergraph in Figure 1.1 represents the metabolic network for the central metabolism of *Escherichia coli* (*E. coli*) (Orth, 2010), which has been obtained by a reduction from the genome-scale metabolic network iAF1260 (Feist et al., 2007). It is composed of 95 reactions and 72 intracellular metabolites which convert external sources like glucose into a biomass reaction. The relationship between reactions and metabolites is represented by a stoichiometric matrix normally denoted by S. Each row of S represents an intermediate metabolite and each column a reaction. Then entry $S_{i,j}$ denotes the stoichiometric coefficient of the $i-$th metabolite in the $j-$th reaction. When the $i-$th metabolite operates as a substrate in the $j-$th reaction, $S_{i,j}$ is often given a negative value. Conversely, $S_{i,j}$ is often defined as a positive value if the $i-$th metabolite is produced. Thus, the stoichiometric matrix of the central metabolic network of *E. coli* has 72 rows and 95 columns, which denote the 72 metabolites and 95 reactions respectively.

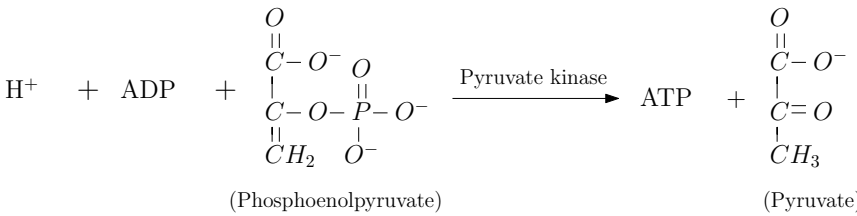


Figure 1.2: A biochemical reaction converting phosphoenolpyruvate (PEP) into pyruvate, releasing a ATP and catalyzed by the enzyme pyruvate kinase.

$$S = \begin{array}{c} \\ \text{H}^+ \\ \text{ADP} \\ \text{PEP} \\ \text{ATP} \\ \text{Pyruvate} \\ \\ \end{array} \begin{pmatrix} \cdots & \cdots & \cdots \\ \cdots & -1 & \cdots \\ \cdots & -1 & \cdots \\ \cdots & -1 & \cdots \\ \cdots & 1 & \cdots \\ \cdots & 1 & \cdots \\ \cdots & \cdots & \cdots \end{pmatrix}. \tag{1.1}$$

Although there are fewer reactions and metabolites than in the genome-scale model iAF1260, the principle pathways, including glycolysis, citric acid cycle (TCA), pentose phosphate pathway (PPP), and nitrogen assimilation reactions are still covered. For instance, the reaction shown in Figure 1.2 is a key biochemical reaction of the glycolysis pathway that transfers a hydrogen ion (H^+), adenosine diphosphate (ADP) and phosphoenolpyruvate (PEP) to a pyruvate, while yielding a ATP at the same time. It is catalyzed by an enzyme named pyruvate kinase, which is somehow ignored in this central network. The stoichiometric ratios of all the metabolites in this reaction are 1:1. So, the stoichiometry of this reaction is shown in Eq. 1.1, from which we can see how it appears in the whole stoichiometric matrix S. The five intermediate metabolites of the reaction are annotated in the left column, which corresponds to the related rows in S. The middle column represents the reaction. For the substrates of the reaction, the coefficients of H^+, ADP and PEP are -1, while the stoichiometric coefficients of ATP and pyruvate are 1 because they are products.

The extracellular metabolites, such as the external nutrients and by-products, exist in the environment outside the cell and may be shared by numerous cells in the community. However, the cells are unable to survive without external resources, and such extracellular metabolites also interconnect with the intracellular metabolism. Therefore, in addition to internal reactions, exchange reactions are added to the metabolic network to transport the extracellular metabolites. The metabolic reactions that can go in both directions are defined as reversible. The reactions that can only go in one direction are defined as irreversible.

Therefore, in the directed hypergraph of a metabolic network, the nodes denote all the metabolites. The directed hyperedges indicate the reactions. Mathematically, these connections between nodes are encoded by a stoichiometric matrix S, in which the rows denote all the metabolites, and the columns denote all the reactions including internal and exchange reactions. Therefore, a metabolic network can be mathematically defined as a tuple $(\mathcal{M}, \mathcal{R}, S)$. $\mathcal{M}$ represents the set of metabolites. $\mathcal{R} = \mathcal{R}_{rev} \cup \mathcal{R}_{Irr}$ is the reaction set including reversible reactions $\mathcal{R}_{rev}$ and irreversible reactions $\mathcal{R}_{Irr}$. $S \in \mathbb{R}^{\mathcal{M}\times\mathcal{R}}$ is the stoichiometric matrix of a metabolic network. For a metabolic network $(\mathcal{M}, \mathcal{R}, S)$, we define a flux vector $\mathbf{v} \in \mathbb{R}^{|\mathcal{R}|}$ to denote the reaction rates of turnover of molecules through the metabolic network, and a vector $\mathbf{M} \in \mathbb{R}^{|\mathcal{M}|}_{\geq 0}$ represents the quantities of metabolites in $\mathcal{M}$.

1.2 Computational modeling of metabolism

1.2.1 Kinetic modeling

Computational modeling approaches have become a powerful tool for elucidating the fundamental mechanisms of cellular metabolism. A first approach to model cellular metabolism is kinetic modeling. Dating back to the year 1943, Chance studied a biochemical system by ordinary differential equations (ODEs) and numerical simulation (Chance, 1943). For decades, ODE-based kinetic models have

been widely developed to study metabolism (Heinrich and Schuster, 1996). Typically, the metabolite concentrations are modeled with continuous variables. The derivative of each metabolite concentration can be described by the sum of the fluxes producing the metabolite (input fluxes) minus the sum of the fluxes utilizing the metabolite (output fluxes). That is, for a metabolic network $(\mathcal{M}, \mathcal{R}, S)$, the dynamics can be written as

$$\frac{d\mathbf{M}}{dt} = S \cdot \mathbf{v}. \tag{1.2}$$

The reaction fluxes $\mathbf{v}$ can be further described by rate laws, as a function depending on metabolite concentrations and kinetic parameters. Then we obtain

$$\frac{d\mathbf{M}}{dt} = S \cdot \mathbf{v}(\mathbf{M}, \mathbf{K}), \tag{1.3}$$

where $\mathbf{K}$ is a vector with kinetic parameters. Thus, the metabolic network can be simulated by solving the system of ODEs.

Kinetic models must be built according to the rate laws and the associated kinetics. Selecting the appropriate chemical rate law is pivotal to kinetic modeling. Up till now, several rate laws have been proposed, and two widely used rate laws are described below.

Law of mass action

The law of mass action is a foundation principle of chemical kinetics originally posited by Guldberg and Waage (Waage and Guldberg, 1864). The law states a relationship between the chemical reaction rate and the corresponding reactant concentrations. Specifically, it implies that the velocity of a chemical reaction is proportional to the product of concentrations of reactants. To illustrate this law suppose a sample reaction $aA + bB \rightarrow C$. According to the law of mass action, its reaction velocity v can be calculated by

$$v = \alpha[A]^a[B]^b. \tag{1.4}$$

Here α is termed as a rate constant for such a reaction, while $[A], [B]$ are the concentrations of substrates A and B, and a, b are the stoichiometric coefficients. It is a non-linear mathematical equation and might be time-varying. Despite the usefulness of mass action in modeling reactions (Voit et al., 2015), its non-linearity makes it difficult to be analyzed, especially for complex reactions.

Michaelis-Menten kinetics

Another common chemical rate law is Michaelis-Menten kinetics, which focuses on enzyme-catalytic reactions. Since its proposal by Leonor Michaelis and Maud Menten (Menten and Michaelis, 1913), Michaelis-Menten equations have been widely used to simulate enzymatic reactions. Suppose a reaction is catalyzed by an

enzyme E in the conversion of a substrate s to a product p, then the reaction rate v can be expressed by the Michaelis-Menten equation

$$v = \frac{dp}{dt} = v_{max} \cdot \frac{s}{K_M + s}. \tag{1.5}$$

Here v_{max} is the maximal rate that can be achieved by saturating the substrate concentration, and K_M is called Michaelis constant, which indicates the substrate concentration for which the reaction rate is half of the maximal value v_{max}. In order to further determine the maximal reaction rate, certain assumptions have been made in the investigation of the enzymatic kinetics. For example, under the condition that the substrates are highly saturated, the maximal reaction rate is assumed to be linear with the enzyme concentration E by the corresponding turnover rate k_{cat}. The turnover rate indicates the maximum number of substrates converted into product per enzyme per time. That is

$$v_{max} = k_{cat} \cdot E. \tag{1.6}$$

Thus, Michaelis-Menten kinetics is commonly formulated as

$$v = \frac{dp}{dt} = \frac{s \cdot k_{cat} \cdot E}{K_M + s} \tag{1.7}$$

to describe the dynamics of p and to calculate the reaction rate v depending on the substrate concentration, the enzyme abundance and the associated kinetic parameters.

Although there are limitations, the law of mass action and Michaelis-Menten kinetics are still the most common chemical laws used in kinetic modeling. Using kinetic modeling, not only can the enzyme kinetics be simulated, but the genetic regulation can also be quantified using Hill functions (Hill, 1910; De Jong, 2002; Rosenfeld et al., 2005). Hill functions are commonly represented by a sigmoid curve and will be detailed later. Although many studies have shown that kinetic modeling is considerably realistic in simulation, pure kinetic models are currently highly difficult to analyze. As a lot of unknown kinetic parameters and non-linear computations are often required, the kinetic modeling of complex reaction systems must resort to simplification. For example, order reduction algorithms and reliable reduced models can be used to obtain simpler models while keeping the key features of the complex systems (Okino and Mavrovouniotis, 1998).

Later, a simplified mathematical framework —constraint-based modeling —was proposed, which only considers certain constraints while ignoring the kinetic details. Constraint-based modeling has become an effective tool in the analysis of metabolic behaviors, especially in genome-scale metabolic networks after 30 years of development (Bordbar et al., 2014; King et al., 2015; Lloyd et al., 2018).

1.2.2 Constraint-based modeling

As the biological information, especially kinetic parameters, is often not available, constraint-based modeling was introduced based on the fact that the cellular behaviors are subject to some constraints. By imposing these constraints in a model, we can make inferences what is possible and what is not, and how a cell might behave, but never predict it deterministically (Palsson, 2000; Price et al., 2004). Such modeling approaches target a solution space, where each solution represents a possible behavior that satisfies the imposed constraints. Thus, in spite of using incomplete information, one can interpret and even predict cellular functions.

The constraints that cells are subject to limit the possible behaviors. It is also these constraints that define the solution space for constraint-based models. The solutions that violate these constraints are excluded from the solution space. The solution space can be reduced by additional constraints. One necessary type of constraint are invariants, which are inviolable and not adjusted in the biological processes. For example, the thermodynamic constraints must always be conserved. There are also time-varying constraints that are adjusted depending on time or conditions, e.g., nutrient availability.

One of the core constraints for this type of modeling are the stoichiometric constraints. They are based on the assumption that all the intermediate metabolites are at steady-state. The fluxes of their production and consumption are balanced under the steady-state assumption. For a metabolic network $(\mathcal{M}, \mathcal{R}, S)$ at steady-state, we obtain the stoichiometric constraints:

$$\frac{d\mathbf{M}}{dt} = S\,\mathbf{v} = 0. \tag{1.8}$$

The thermodynamic constraints in steady-state stoichiometric models regard the reversibility of a reaction. For irreversible reactions, the reactants convert to products, and the products cannot turn back to reactants. Thus, constraints

$$\mathbf{v}_{\mathcal{R}_{Irr}} \geq 0 \tag{1.9}$$

are added for irreversible reactions $\mathcal{R}_{Irr}$.

Another critical point in constraint-based modeling is that the detailed enzyme kinetics describing reaction rates, e.g., Michaelis-Menten kinetics, is replaced by suitable lower and upper bounds on the reaction fluxes. This leads to the flux capacity constraints

$$\mathbf{lb} \leq \mathbf{v} \leq \mathbf{ub}, \tag{1.10}$$

where $\mathbf{lb}, \mathbf{ub}$ are vectors of lower and upper bounds for the reaction fluxes $\mathbf{v}$. However, with more and more biological information becoming available, one can constrain the reaction rates with their enzyme abundances and kinetic parameters. This leads to the, so-called, enzymatic capacity constraints in resource allocation models, which will be discussed later. Besides the stoichiometric, thermodynamic, and

flux capacity constraints described above, there are possibly other constraints to determine the solution space for a metabolic network, such as regulatory constraints and environmental constraints (Price et al., 2004).

Geometrically, the stoichiometric matrix S mathematically defines a linear space for the fluxes $\mathbf{v}$ at steady-state by $S\mathbf{v} = 0$. Putting together constraints $S\mathbf{v} = 0$ and $\mathbf{v}_{\mathcal{R}_{Irr}} \geq 0$, the solution space becomes a polyhedral cone, which is called the flux cone (Terzer, 2009). If there are lower and upper bounds on the reaction fluxes, the solution space is a polytope defined by $S\mathbf{v} = 0$ and $\mathbf{lb} \leq \mathbf{v} \leq \mathbf{ub}$.

To further study the metabolic behaviors under these constraints, many constraint-based approaches are using linear optimization, which can be solved very efficiently. Linear optimization is a technique to optimize a linear objective function subject to linear equalities and linear inequalities. The best known optimization-based method is *flux balance analysis* (FBA), which was shown to be in agreement with experimental observations by many studies (Förster et al., 2003b; Shlomi et al., 2005). It assumes that the cells maximize the biomass production rate or some other cellular objectives (Savinell and Palsson, 1992; Orth et al., 2010). With this objective function and the stoichiometric, thermodynamic, and flux capacity constraints, a linear optimization problem (LP) can be constructed to predict the flux distribution. Mathematically, FBA can be expressed as

$$\begin{aligned} \max_{\mathbf{v}} : \quad & \mathbf{c}^T\mathbf{v} \\ s.t. \quad & S\mathbf{v} = 0 \\ & \mathbf{lb} \leq \mathbf{v} \leq \mathbf{ub}, \end{aligned} \tag{1.11}$$

where $\mathbf{c}$ is a vector with the coefficient of each reaction contributing to the objective function, $\cdot^T$ denotes the transposition.

Although FBA provides a unique value for the objective function, the optimal solution is not always unique. There often exist alternative optimal solutions wherein the same maximal objective value can be achieved. Even though FBA maximizes objective function with the constraints shown in Eq. 1.11, it is often the case that the optimal solution space does not shrink to a single point. There may be alternative pathways leading to the same optimal value. To explore the alternate optimal solutions, several approaches have been proposed, such as *flux variability analysis* (FVA) (Mahadevan and Schilling, 2003).

The objective function in FBA is often defined as the biomass reaction flux in order to maximize the growth rate. Several other objectives have also been used in constraint-based modeling approaches (Holzhütter, 2004; Knorr et al., 2006; Schuetz et al., 2007). As Price *et.al* summarized in (Price et al., 2004), the objective functions are generally formed for three purposes: first, the exploration of the phenotypic potential of a metabolic network (Price et al., 2002); second, the determination of likely physiological behaviors by choosing different objective functions (Edwards et al., 2001), e.g., biomass and ATP productions; third, the design of strains to satisfy an engineering purpose (Burgard et al., 2003).

1.2.3 Resource allocation modeling

In recent years, genome-scale reconstructions have increasingly moved towards the integrated models of metabolism and macromolecular production (Thiele et al., 2009). Some parts neglected in FBA-type approaches are more and more noted essential for understanding cellular growth, which is the topic of this section: trade-off between costs and benefits of enzyme production. Although the enzymes speed up metabolic reactions, they consume a lot of precursors and energy as well. The costs and benefits must be balanced for optimal growth (Weiße et al., 2015). It has been revealed that how the cell allocates its resources to build numerous functional macromolecules, especially ribosomes and enzymes, is significant to the understanding of cellular growth, behaviors and physiology (Dekel and Alon, 2005). The resulting resource allocation models can be described by so-called self-replicator models (Koch, 1988; Molenaar et al., 2009; Giordano et al., 2016; Yegorov et al., 2019). Compared with classical metabolic networks, these introduced an additional precursor costs for macromolecules such as enzymes, transporters, constructional compounds and ribosomes.

Goelzer and coworkers further developed a constraint-based approach *resource balance analysis* (RBA) to predict the resource allocation based on the self-replicator models (Goelzer and Fromion, 2011; Goelzer et al., 2011). RBA considers the translational apparatus and the enzyme-catalytic relationships besides the stoichiometric information of a metabolic network at steady-state. Independently, classical metabolic models also have been extended to integrate protein expression machinery by Lerman *et al.*, which are termed as *ME-models* (Lerman et al., 2012). Instead of directly maximizing the growth rate in RBA, they proposed to minimize the production rate of ribosomes while still supporting its growth.

However, such static approaches cannot allow the inspection of dynamic adaptations of resource allocation in response to the changing environment. Recently, several dynamic approaches have been developed to predict the dynamic metabolism, including the costs of enzyme production. *Dynamic enzyme-cost flux balance analysis* (deFBA) was proposed to predict the dynamic reaction fluxes and enzyme amounts using a dynamic optimization algorithm (Waldherr et al., 2015). To simulate the phototrophic growth of cyanobacteria over a diurnal cycle, *conditional flux balance analysis* (cFBA) was introduced in consideration of the changing light intensity of the day (Rügen et al., 2015). In addition, RBA was extended to *dynamical resource balance analysis* (dRBA) and available for modeling dynamical conditions during the batch culture in a bioreactor (Jeanne et al., 2018). In 2019, Yang *et al.* also developed an algorithm to cover time-course metabolism based on ME-models. They called it *DynamicME* (Yang et al., 2019). In that work, two procedures for the iteration of ME-models were proposed, one of which accounted for protein dynamics while the other did not.

Yet, the gene regulatory system is not considered in all these existing approaches capable of predicting resource allocation. Instead, it is considered to be substituted

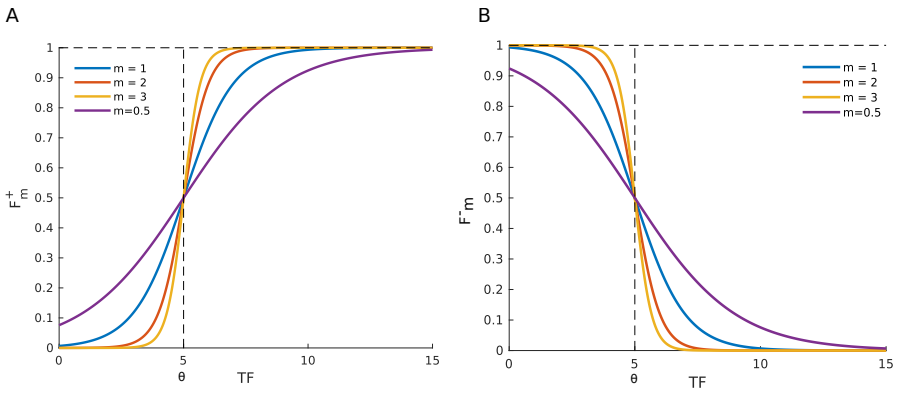


Figure 1.3: Sigmoid curves. m is the Hill coefficient. When $m \to \infty$, the associated Hill function can be approximated by a step function.

by the optimization. For example, deFBA and cFBA maximize the total biomass along a time course. DynamicME maximizes the growth rate at each time interval with the iterative strategy. However, there is no guarantee of consistency between metabolism and regulation. Also, genetic regulation does not always maximize the growth rate. All in all, there is a strong need for new mathematical frameworks capable of integrating regulatory events with these resource allocation models or ME-models.

1.3 Modeling gene regulation

It is generally believed that the gene regulatory system functions as a control system that administrates when and which protein is translated to which extent and in which location inside a cell. Gene regulation also involves thousands of molecules, such as protein, DNA and mRNA, interconnected mutually as a huge net akin to metabolic networks. Mathematical models and computational tools are essential for revealing the fundamental mechanism of regulatory networks because of its complexity. Here, we give an overview of a few main established mathematical frameworks to model, analyze and simulate gene regulation.

1.3.1 Hill functions

Hill functions have been extensively used to quantify the relationship between the evolution of regulated protein and the corresponding transcription factor (TF) or regulatory protein (RP) concentration. They were first formalized in 1910 by Archibald Hill (Hill, 1910) to describe how the fraction of a macromolecule depends on the concentration of its ligand. Since the regulatory process is exerted by TF binding with the translational apparatus such as DNA and mRNA, Hill functions are also extended to model gene regulation.

Hill functions are often represented by sigmoid curves, see Figure 1.3. Suppose a gene encoding protein p is up- or down-regulated by a transcription factor TF, then the regulatory effect can be individually calculated by the Hill functions

$$F_m^+(TF,\theta) = \frac{TF^m}{\theta^m + TF^m}; \quad F_m^-(TF,\theta) = \frac{\theta^m}{\theta^m + TF^m}, \tag{1.12}$$

where m is the Hill coefficient indicating the number of binding sites for ligands in a protein. θ is a threshold indicating that the gene encoding p can be activated or inactivated when the relevant transcription factor abundance TF reaches θ. Hence, the translation rate of regulated protein p can be quantified as

$$\frac{dp}{dt} = v_p \cdot \frac{TF^m}{\theta^m + TF^m} - \gamma_d \cdot p; \quad \frac{dp}{dt} = v_p \cdot \frac{\theta^m}{\theta^m + TF^m} - \gamma_d \cdot p, \tag{1.13}$$

where the reaction rate parameter v_p is often assumed to be the maximum expression rate of protein p and γ_d represents the degradation or dilution rate. By the Hill functions, the up- or down-regulation can be directly integrated with ODE-based kinetic modeling.

1.3.2 Piecewise linear differential equations

Similar to the kinetic models, modeling genetic regulation by Hill functions must resort to finer simplification, especially because of the Hill type (Okino and Mavrovouniotis, 1998). Therefore, people's attention shifted from quantifying the regulatory effects to the qualitative features of regulatory controls (Glass and Kauffman, 1973; Thomas, 1973). Monod and Jacob first introduced gene states and logical equations to model gene regulatory systems (Jacob and Monod, 1961). Further, Sugita posited that genetic regulation is a biochemical molecular system. Actually, the rate of the dynamic process is controlled by parameters, the logic nature of which is equivalent to the switch of a logical circuit having input, output and a time delay (Sugita, 1963). The logical states depend on the concentrations of related TFs and thresholds. To perform a comparison with Hill functions, we introduce a gene state $\overline{g}$ encoding the protein p in Eq. 1.13. $\overline{g}$ performs as the logical control for the synthesis of protein p, which is classified as $\bar{g}^+$ and $\bar{g}^-$ depending on the up- or down-regulation of a TF. As shown in Figure 1.3, when m in the Hill function 1.12 is big enough, the relevant Hill function can be approximated as a step function that can be formalized as

$$\bar{g}^+(TF,\theta) = \begin{cases} 1 & TF \geq \theta \\ 0 & TF < \theta \end{cases}; \quad \bar{g}^-(TF,\theta) = \begin{cases} 0 & TF \geq \theta \\ 1 & TF < \theta \end{cases}. \tag{1.14}$$

Based on the approximation, a Hill function can be substituted by a Boolean variable as the logical control on gene expression, resulting in a Piecewise Linear Differential Equation (PLDE) (Glass and Kauffman, 1973; Snoussi, 1989; De Jong, 2002; De Jong et al., 2004)

$$\frac{dp}{dt} = v_p \cdot \overline{g} - \gamma_d \cdot p, \tag{1.15}$$

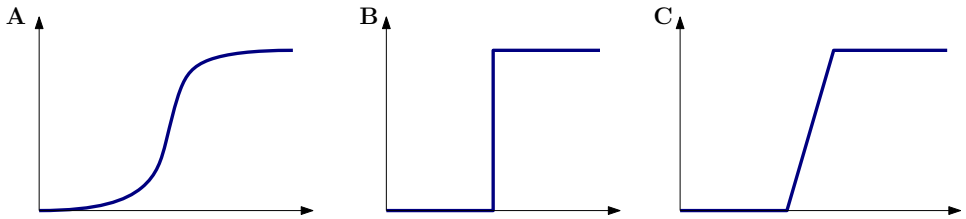


Figure 1.4: A sigmoid curve (**A**), its Boolean logical state (**B**) and PLDE curve (**C**).

where p is the protein concentration. $\overline{g}$ is the associated gene state that can be $\bar{g}^{+}$ or $\bar{g}^{-}$ in Eq. 1.14 depending on the TF abundance and the threshold. The reaction rate parameter v_p here can also be defined based on chemical laws such as the mass action and Michaelis-Menten kinetics, as suggested in the studies of (Chaves et al., 2019).

1.3.3 Boolean logical networks

The early ideas (Jacob and Monod, 1961; Sugita, 1963) of modeling gene regulation as logical systems resulted in the development of Boolean regulatory networks and discrete regulatory networks (Kauffman, 1969; Thomas, 1973). Nearly in the same year in which PLDE was introduced to model gene regulation, these authors formalized the gene regulatory network by Boolean rules, which combine Boolean variables by logical operators AND, OR and NOT (Kauffman, 1969; Thomas, 1973). In the formalization, each gene has two expression state (`on` or `off`) and completely ignored the quantitative regulatory effect. Figure 1.4 shows the caricatures of sigmoid curve, the Boolean logical state and PLDE. Compared with Hill functions and PLDE, Boolean regulatory networks only require the Boolean logical rules to perform regulatory control. Thus, the essential control behavior is preserved and able to capture the qualitative control of the genetic regulation in a simplified fashion. Besides Boolean variables, the gene states can also have multiple values, which are extended into a discrete logical network. Currently, Boolean regulatory networks are one of the best-known methodologies in the investigation of gene-state transitions. Moreover, they have been applied in the study of molecular interactions on the genome scale (Kauffman et al., 2003; Li et al., 2004; Samal and Jain, 2008).

Mathematically, the classic *transcriptional regulatory networks* (TRNs) having n genes can be defined as a directed network with nodes denoted by a vector $\overline{\mathbf{g}} \in \{0, 1\}^n$ where the value 0 corresponds to `off` and 1 to `on`. The edges represent gene interactions, which are usually labeled by negative/positive sign. Each gene state can be determined by a Boolean function f_i of Boolean variables with the logical connectives OR, AND and NOT. Therefore, for a TRN with n genes (nodes), the Boolean states are composed of the expression states of n genes, which are 2^n in total.

To analyze the dynamics of TRNs, discrete time was introduced so that the gene state at a given time point t is $\overline{\mathbf{g}}(t)$. Synchronous and asynchronous algorithms are matured for an understanding of the dynamic state transitions of a TRN (Bernot et al., 2004; Fauré et al., 2006). For synchronous dynamics, all the genes update their states simultaneously. This never happens in real cells in that all the genes will not be expressed within the same period to reach the thresholds to activate/inhibit the other genes. As Thomas and Kaufman argued in (Thomas and Kaufman, 2001), there is no reason why the time delays should be equal and in fact they are often very unequal. The duration of the delay depends mainly on the production/decay rate of gene product involved in the regulatory process. They proposed that different delays must be considered for the update of gene states, which would lead to the non-deterministic asynchronous dynamics. In the absence of precise information about the time delays in biological systems, Siebert and Bockmayr proposed to impose constraints on the time delays in the form of inequalities (Siebert and Bockmayr, 2008). They applied the theory of timed automata to track the time while the system evolves. Thus, the imposed constraints on the time delays can be evaluated for state transitions in the evolution of the system. Unlike the ODE-based kinetic modeling that gives precise dynamics, the transition systems of such logic-based regulatory networks are non-deterministic. Therefore, they are highly favorable in the exploration of state transitions, searching for attractors, as well as in model-checking (Thieffry et al., 1993; Devloo et al., 2003; Bernot et al., 2004; Batt et al., 2005). At present, computational and simulation tools permit the systematic study of gene regulatory networks, even at the genome-scale (Luscombe et al., 2004).

1.4 Integration of metabolism and regulation

1.4.1 Hybrid modeling

Dating back to 1963, Sugita already introduced the idea of molecular automata (Sugita, 1963). The cellular molecular system, for instance protein expression, is controlled by the genetic circuits. Typically, the protein expression that is encoded by a gene is a continuous process. On the other hand, its regulatory control involves a TF binding with the gene, which is considered a discrete event. It was realized later that the metabolic-regulatory system exhibits both discrete and continuous behaviors, and is, in fact, a hybrid system. A hybrid system combines discrete states and continuous dynamics, which perfectly captures the cellular regulatory control and dynamic metabolism. In the past, several attempts and processes have been made in modeling such discrete-continuous dynamics.

For instance, McAdams *et al.* first developed a hybrid modeling method by incorporating the biochemical kinetic model into the gene circuits (McAdams and Shapiro, 1995). Hybrid system approaches combining the idea of control theory and computer science are then introduced to model biomolecular networks with genetic circuits (Alur et al., 2001; Lincoln and Tiwari, 2004), which, for instance,

have been successfully applied in simulating Delta-Notch signaling (Ghosh and Tomlin, 2001) and cell cycle (Singhania et al., 2011). Bockmayr and Courtois proposed that hybrid concurrent constraint programming may provide a promising tool in modeling dynamics of biological systems, such as genetic regulation on biochemical pathways, by reaching thresholds and enzyme kinetics (Bockmayr and Courtois, 2002). Hybrid automata, as a formal hybrid system, have also been introduced to model genetic regulatory networks (De Jong et al., 2003; Bortolussi and Policriti, 2008; Ye et al., 2008).

1.4.2 Constraint-based modeling with genetic regulation

Constraint-based modeling approaches permit the reconstruction and analysis of genome-scale metabolic networks. However, such methods lead to incorrect predictions when the metabolism is significantly affected by regulatory events. To extend the predictive capability of classic FBA, Covert *et al.* developed *regulatory flux balance analysis* (rFBA) to incorporate Boolean regulatory rules into the metabolic network (Covert et al., 2001). This has been implemented in *Saccharomyces cerevisiae* (Herrgård et al., 2006) and *Escherichia coli* (Covert et al., 2008). By a similar strategy, *FlexFlux* (Marmiesse et al., 2015) was proposed to integrate multi-valued discrete logical rules into the metabolic network. The basic idea of both rFBA and FlexFlux is iterating FBA by splitting the growth phase into discrete time steps. At each time step, the gene states are updated. The updated gene states are then imposed as bounds onto the reaction fluxes.

Integrated FBA (iFBA) was developed to model the dynamics of metabolism, regulatory Boolean logic and signaling networks, the strategy of which is to integrate the rFBA framework and ODE-based kinetic models (Covert et al., 2008). In iFBA, rFBA is combined with a set of ODEs by passing variables from one to each other, showing significant improvement as compared with rFBA and ODE-based kinetic modeling. In parallel, *integrated dynamic FBA* (idFBA) was proposed as an iteratively dynamic approach for modeling an integrated system which includes signaling, metabolic and regulatory networks (Lee et al., 2008). Compared to rFBA and iFBA, idFBA integrates regulatory rules by introducing a binary parameter for each reaction at each time point, thereby indicating whether the reaction is active or not. As for the whole network, the binary parameters are represented by a matrix, in which the rows are reactions and the columns are time points. At each time step, the binary parameter is used to constrain the reaction activity by multiplying the upper bound of the corresponding reaction flux.

The static steady-state approaches are particularly advantageous to the genome-scale metabolic-regulatory models. *Steady-state regulatory flux balance analysis* (SR-FBA) allows, for the first time, predicting the steady-state behaviors in an integrated genome-scale metabolic-regulatory model, which was presented in (Shlomi et al., 2007). Mathematically, by the gene-reaction mapping, it converts the Boolean logic of gene regulation to the linear inequalities. Combining these

linear inequalities with FBA formalism, SR-FBA can be formulated as a mixed-integer linear program (MILP) (Shlomi et al., 2007; Jensen et al., 2011). Another idea is to integrate high-throughput data with metabolic networks directly since the gene regulatory networks are inferred based on biological datasets, e.g. microarray datasets (Wang et al., 2006). In 2010, Chandrasekaran and Price put forward another approach to integrate the regulatory information with FBA, based on high-throughput data called *Probabilistic regulation of metabolism* (PROM) (Chandrasekaran and Price, 2010). Compared to Boolean logic-based approaches, this permits the construction of genome-scale metabolic-regulatory networks and allows the integration of quantitative regulation with constraint-based modeling.

Yet, the protein expression machinery is not considered in all these approaches. Overall, the interplay between the gene regulatory system and the metabolic network is still poorly understood, despite these modeling frameworks have been developed. There is a need for formal, dynamic and efficient mathematical frameworks to integrate diverse biological processes particularly genetic regulation, protein production costs and metabolism.

1.5 Structure of the thesis

Based on the previous endeavors in modeling metabolism and gene regulation, this PhD thesis concentrates on developing computational algorithms that allow the study of the dynamic integration of the two systems. Chapter 2 describes several extension approaches of classic flux balance analysis, including *dynamic flux balance analysis* (DFBA), *regulatory flux balance analysis* (rFBA), *resource balance analysis* (RBA) and *dynamic enzyme-cost flux balance analysis* (deFBA), which are the basis for the following chapters.

In Chapter 3, we propose an analytic pipeline to explore the optimal solution space in rFBA with applications to the core carbon network and the central metabolic network of *E. coli*.

We then present our early attempts for iterating resource balance analysis with Boolean regulatory rules in Chapter 4. Specifically, two approaches are introduced, one of which merely iterates RBA while updating the external substrates and biomass. It is called *iterative resource balance analysis* (iRBA). The other further incorporates Boolean regulatory rules into iRBA using a strategy similar to rFBA. It is called *regulatory iterative resource balance analysis* (riRBA).

In Chapter 5, we propose *metabolic-regulatory networks* (MRNs) to represent the typical dynamic interaction between cellular metabolism and transcriptional gene regulation. Formalizing metabolic-regulatory networks by hybrid automata captures well the continuous dynamics of metabolism and the discrete regulatory control. The formalization introduces an innovative mathematical framework that combines metabolic-genetic networks and gene regulatory networks in a hybrid discrete-continuous system.

To apply the theoretical formalization in Chapter 5, we further show in Chapter 6 that the optimal control strategy for the hybrid discrete-continuous systems representing our MRNs can be constructed as a dynamic optimization program. Thus, a constraint-based approach termed *regulatory dynamic enzyme-cost flux balance analysis* (r-deFBA) is developed to predict the dynamics of metabolism, discrete state transitions and optimal resource allocation that maximize the biomass production over a time interval.

Chapter 7 continues presenting another possible application of the theoretical formalization in Chapter 5. Instead of focusing on an individual-level metabolic-regulatory systems, we use product automata to model the population-level *metabolic-regulatory networks* in Chapter 7. At last in Chapter 8, we summarize and conclude this PhD thesis.

Chapter 2

Extensions of classic FBA

This chapter aims at providing the basis for the following chapters. Several extensions of classic flux balance analysis: *dynamic flux balance analysis* (dFBA), *regulatory flux balance analysis* (rFBA), *resource balance analysis* (RBA) and *dynamic enzyme-cost flux balance analysis* (deFBA) will be presented.

2.1 Dynamic flux balance analysis

Since the 1980s, *flux balance analysis* (FBA) has been a powerful method (Orth et al., 2010) for the prediction of reaction fluxes in metabolic networks. Later, *dynamic flux balance analysis* (DFBA) was developed by extending FBA to account for the dynamics of extracellular metabolites and biomass. There exist two variants of DFBA: dynamic optimization approach (DOA) and static optimization-based DFBA approach (SOA) (Mahadevan et al., 2002). To illustrate FBA-type formalisms intuitively, we construct a schematic metabolic network in Figure 2.1.

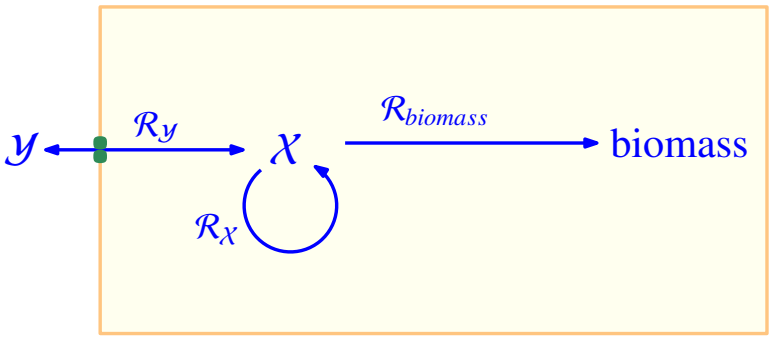


Figure 2.1: A schematic metabolic network.

$\mathcal{Y}$ represents extracellular metabolites with the associated concentrations $\mathbf{Y}$, in

which $\mathcal{N}$ are nutrients. $\mathcal{N} \subseteq \mathcal{Y}$. $\mathcal{R}_\mathcal{Y}$ is the set of exchange reactions between $\mathcal{Y}$ and intermediate metabolites $\mathcal{X}$ with the fluxes $\mathbf{v}_\mathcal{Y}$. In particular, $\mathcal{R}_\mathcal{N}$ is the set of uptake reactions of nutrients $\mathcal{N}$ and $\mathbf{v}_\mathcal{N}$ is the vector of uptake fluxes. $\mathbf{X}$ represents the concentrations of $\mathcal{X}$. $\mathcal{R}_\mathcal{X}$ is the set of intermediate reactions with the corresponding fluxes $\mathbf{v}_\mathcal{X}$. $\mathcal{R}_{biomass}$ is the biomass reaction to produce biomass with the flux $\mathbf{v}_{biomass}$, which indicates the growth rate μ. $\mathcal{R}$ is a set including all reactions such that $\mathcal{R} = \mathcal{R}_\mathcal{Y} \cup \mathcal{R}_\mathcal{X} \cup \mathcal{R}_{biomass}$.

With the abbreviations and notation above, we will first explain the strategy of static optimization-based DFBA.

2.1.1 Updating biomass and external metabolites

Consider the metabolic network presented in Figure 2.1. People often model the dynamics of extracellular metabolite and biomass by ordinary differential equations (Varma and Palsson, 1994; Mahadevan et al., 2002):

$$\frac{d\text{Bio}}{dt} = \mu \cdot \text{Bio}, \tag{2.1}$$

$$\frac{d\mathbf{Y}}{dt} = S_{\mathcal{Y},\mathcal{R}_\mathcal{Y}} \cdot \mathbf{v}_\mathcal{Y} \cdot \text{Bio}, \tag{2.2}$$

where Bio is the biomass concentration and μ denotes the growth rate. $\mathbf{Y}$ are external metabolite concentrations. $S_{\mathcal{Y},\mathcal{R}_\mathcal{Y}}$ is the stoichiometric matrix in which the rows represent external metabolites and the columns denote exchange reactions. $\mathbf{v}_\mathcal{Y}$ are the exchange fluxes following the convention that nutrient uptake fluxes are negative and secretion fluxes are positive.

DFBA (SOA) connects classic FBA with a dynamically changing environment in an iterative way. The simulation period is divided into n time intervals of length Δt. In DFBA (SOA), an FBA program is solved at the beginning of each interval to get the flux distribution at time step k, denoted by $\mathbf{v}^k$. In the iterative course, the external metabolite and biomass concentrations are updated stepwise by

$$\text{Bio}^k = \text{Bio}^{k-1} \cdot e^{\mu^k \cdot \Delta t}, \tag{2.3}$$

$$\mathbf{Y}^k = \mathbf{Y}^{k-1} - \frac{S_{\mathcal{Y},\mathcal{R}_\mathcal{Y}} \cdot \mathbf{v}^k_\mathcal{Y}}{\mu^k} \cdot \text{Bio}^k \cdot (1 - e^{\mu^k \cdot \Delta t}), \tag{2.4}$$

where Bio^k represents the biomass concentration and $\mathbf{Y}^k$ represents the external metabolite concentrations at the end of time step k. μ^k is the growth rate and $\mathbf{v}^k_\mathcal{Y}$ is the set of exchange reaction fluxes at time step k. Here, uptake fluxes are assumed to be negative while exchange reaction fluxes for secretions are positive. We note that the starting time point of the current time step is equal to the endpoint of the previous time step. To start the iterative process, the initial concentrations of biomass and external species have to be specified.

2.1.2 Updating constraints on uptake fluxes

The uptake fluxes of nutrients are constrained by the varying external metabolites. For example, the uptake flux $\mathbf{v}_i$ of a nutrient $\mathcal{N}_i$ may follow a Michaelis-Menten kinetics

$$\mathbf{v}_i^k = \frac{\mathbf{N}_i^{k-1}}{K_M^i + \mathbf{N}_i^{k-1}} \cdot \mathbf{v}_i^*, \tag{2.5}$$

where $\mathbf{v}_i^k$ represents the uptake flux at time step k of the nutrient $\mathcal{N}_i$, $\mathbf{N}_i^{k-1}$ denotes the nutrient concentration at the end of time step $(k-1)$, K_M^i is the relevant Michaelis constant and $\mathbf{v}_i^*$ is the maximum uptake flux. Differently, in the implementation of DFBA (SOA) of Cobra Toolbox (`https://opencobra.github.io/cobratoolbox`) (Becker et al., 2007), the uptake flux of nutrient $\mathcal{N}_i$ is constrained with

$$\frac{-\mathbf{N}_i^{k-1}}{\mathrm{Bio}^{k-1} + \Delta t} \leq \mathbf{v}_i^k \leq 0, \tag{2.6}$$

where Bio^{k-1} is the total biomass concentration at the end of time step $(k-1)$. By convention, the uptake fluxes of nutrients are negative. So, the maximal value of uptake flux is 0. Mathematically, such a dependency of uptake rates on nutrient concentrations can be formulated as

$$\mathbf{lb}_i(\mathbf{N}^{k-1}) \leq \mathbf{v}_i^k \leq \mathbf{ub}_i(\mathbf{N}^{k-1}), \quad i \in \mathcal{R}_\mathcal{N} \tag{2.7}$$

where $\mathbf{N}^{k-1}$ represents the nutrient concentrations at the end of the previous time step $(k-1)$, which actually denotes the nutrient concentrations available for the growth at time step k. $\mathbf{lb}_i, \mathbf{ub}_i$ are lower and upper bounds on the fluxes of uptake reactions $i \in \mathcal{R}_\mathcal{N}$ depending on the nutrient concentrations $\mathbf{N}^{k-1}$.

2.1.3 Static optimization problem

Given FBA formalism and the effect of dynamically changing nutrients, we obtain the static optimization problem in DFBA (SOA)

$$\begin{aligned} \max_{\mathbf{v}^k} \quad & \mathbf{v}_{biomass}^k \\ s.t. \quad & S_{\mathcal{X},\mathcal{R}} \cdot \mathbf{v}^k = 0 \\ & \mathbf{lb}_i(\mathbf{N}^{k-1}) \leq \mathbf{v}_i^k \leq \mathbf{ub}_i(\mathbf{N}^{k-1}), \quad i \in \mathcal{R}_\mathcal{N} \\ & \mathbf{lb} \leq \mathbf{v}^k \leq \mathbf{ub}, \end{aligned} \tag{2.8}$$

at time step $k \in \{1, \ldots, n\}$. Here $\mathbf{v}_{biomass}^k$ is the flux of biomass reaction that is normally defined as the growth rate μ^k, $S_{\mathcal{X},\mathcal{R}}$ is the stoichiometry matrix in which the rows are the intermediate metabolites $\mathcal{X}$ and the columns are the reactions $\mathcal{R}$, $\mathbf{v}^k$ denotes the flux distribution at time step k, and $\mathbf{lb}, \mathbf{ub}$ are the vectors of lower and upper bounds on reaction fluxes.

To start DFBA (SOA), the initial concentrations of the extracellular metabolites and biomass are given by the modeler. The static optimization problem in Eq. 2.8 is iteratively solved to obtain the flux distribution $\mathbf{v}^k$ and growth rate μ^k at time step k. Eq. 2.3 and Eq. 2.4 are used to calculate the concentrations of biomass and external species at the end of time step k. The iterative process stops at the end of the simulation period, which is set based on the initial concentrations of nutrients. It does not make sense to continue the process after the nutrients are exhausted and unable to support cellular growth. Rather than solving FBA iteratively in DFBA (SOA), DFBA (DOA) maximizes the whole biomass over a given simulation period and solves the optimization problem once. This results in a non-linear dynamic optimization problem which is hard to solve, especially for large networks. That limits its application. In the following DFBA denotes the SOA approach unless we specifically stress DFBA (DOA).

2.2 Regulatory flux balance analysis

Based on the strategy of DFBA, *regulatory flux balance analysis* (rFBA) aims at predicting the metabolic shifts in a changing environment taking into account genetic regulation (Covert et al., 2001). It incorporates Boolean regulatory rules into the iterative dynamic process of DFBA. In a Boolean *transcriptional regulatory network* (TRN), each gene state $\overline{\mathbf{g}}_i \in \{0, 1\}$ takes the values of 1 resp. `on` or 0 resp. `off`, which is decided by a Boolean-valued function f_i. For a Boolean dynamical system regulating metabolism, $\overline{\mathbf{g}}_i^k$ represents the gene state with values 0 or 1 at time step k. Biologically, the dynamics of gene states of a TRN is determined by the presence or absence of extracellular species $\mathcal{Y}$ and regulatory proteins $\mathcal{RP}$, as well as the reaction activities. So, the Boolean dynamical rules for updating gene states from one time step to the next can be formalized by an equation (Samal and Jain, 2008)

$$\overline{\mathbf{g}}_i^k = f_i(\overline{\mathbf{I}}_{\mathcal{Y}}^k, \overline{\mathbf{I}}_{\mathbf{v}_\mathcal{Y}}^k, \overline{\mathbf{I}}_{\mathbf{v}_\mathcal{X}}^k, \overline{\mathbf{g}}_{\mathcal{RP}}^{k-1}), \tag{2.9}$$

where $\overline{\mathbf{I}}_{\mathcal{Y}}^k \in \{0, 1\}$ indicates the absence or presence of extracellular metabolites $\mathcal{Y}$ at time step k, and $\overline{\mathbf{I}}_{\mathbf{v}_\mathcal{Y}}^k, \overline{\mathbf{I}}_{\mathbf{v}_\mathcal{X}}^k \in \{0, 1\}$ indicate the activity states of reactions $\mathcal{R}_\mathcal{Y}$ and $\mathcal{R}_\mathcal{X}$. $\overline{\mathbf{g}}_{\mathcal{RP}}^{k-1} \in \{0, 1\}^{|\mathcal{RP}|}$ represents the gene expression states encoding regulatory proteins $\mathcal{RP}$ at time step $(k-1)$. There is a time delay between the gene states in this equation because the regulatory proteins are able to activate/inhibit the next gene expression state only when their synthesis or decay time has elapsed.

To integrate the Boolean regulatory rules with DFBA, the gene states updated by Eq. 2.9 are imposed to constrain the reaction activities in the static optimization problem of FBA in Eq. 2.8. The flux distribution $\mathbf{v}$ resulting from FBA, in turn, affects the update of the gene states. Therefore, on one hand, rFBA iterates FBA by knocking out genes resp. reactions whose states are `off` in the current time step. The biomass and external substrate concentrations are updated in every time step according to the Eq. 2.3 and Eq. 2.4. On the other hand, the gene states in the

regulatory network are updated iteratively, depending on the external metabolite concentrations and some reaction fluxes. The dependency of the uptake fluxes on the dynamically changing nutrients is formulated in Eq. 2.7. Again, at the next step, the genes resp. reactions are knocked out if the gene states are `off` in running the static optimization of FBA.

Therefore, rFBA is also an iterative approach. The growth phase is divided into n time intervals of length Δt. The flux distribution at time step k is denoted by $\mathbf{v}^k$. For each reaction flux $\mathbf{v}_i^k$ there is binary variable $\overline{\mathbf{r}}_i^k$ describing whether reaction i is active or not. The value of $\overline{\mathbf{r}}_i^k$ is determined by a set of Boolean rules that encode how the current activity of a reaction is affected by the extracellular metabolite concentrations and the reaction fluxes. If $\overline{\mathbf{r}}_i^k = 1$, the reaction flux is constrained by the original FBA bounds $\mathbf{lb}_i$ and $\mathbf{ub}_i$. If $\overline{\mathbf{r}}_i^k = 0$, the reaction flux is set to 0. Mathematically, at time step $k \in \{1, \ldots, n\}$, rFBA solves the static optimization problem:

$$
\begin{aligned}
\max_{\mathbf{v}^k} \quad & \mathbf{v}_{biomass}^k \\
s.t. \quad & S_{\mathcal{X},\mathcal{R}} \cdot \mathbf{v}^k = 0 \\
& \overline{\mathbf{r}}_i^k = f_i(\mathbf{Y}^{k-1}, \mathbf{v}^{k-1}) \\
& \mathbf{lb}_i \cdot \overline{\mathbf{r}}_i^k \leq \mathbf{v}_i^k \leq \mathbf{ub}_i \cdot \overline{\mathbf{r}}_i^k \\
& \mathbf{lb}_i(\mathbf{N}^{k-1}) \leq \mathbf{v}_i^k \leq \mathbf{ub}_i(\mathbf{N}^{k-1}), \quad i \in \mathcal{R}_{\mathcal{N}} \\
& \mathbf{lb} \leq \mathbf{v}^k \leq \mathbf{ub},
\end{aligned}
\tag{2.10}
$$

where $\mathbf{v}_{biomass}^k$ denotes the biomass reaction flux and indicates the growth rate μ^k at time step k, $S_{\mathcal{X},\mathcal{R}}$ is the stoichiometric matrix, $\mathbf{Y}^k$ is a vector that includes the extracellular metabolite concentrations, and f_i is a Boolean-valued function that determines the activity state of reaction i at time step k, depending on the external metabolite concentrations and reaction fluxes at the end of the previous time step.

Altogether, Eq. 2.10 is solved to get the flux distribution $\mathbf{v}^k$ and growth rate μ^k at time step k. Eq. 2.3 and Eq. 2.4 are used to update the biomass and substrate concentrations at the end of time step k. The resulting flux distribution and substrate concentrations are further used to decide the reaction activity states at the next time step. The initial values of the extracellular metabolite concentrations and biomass are given according to the environmental conditions.

One thing that must be taken note of when running rFBA is the initialization of the Boolean variables. Although the initial external metabolite concentrations are fixed, the initial fluxes must also be specified in order to determine the initial values of the Boolean variables. To avoid that the predicted metabolism oscillates in the beginning, the authors of (Covert et al., 2001) suggested that all initial fluxes are set to zero to calculate a stable state under which the Boolean variables do not change any more. Then, use the stable state to initiate the first time step of rFBA.

Another thing we wish to point out is that the time interval of rFBA shall be defined based on the protein synthesis and degradation rate. This is because it requires a certain time period to update the expression states between genes. Also, there exists a time delay between the reaction activities and the gene states encoding the corresponding enzymes. In a steady-state condition, the average protein synthesis and decay time are assumed to be equal. It was set to 15 min in (Covert et al., 2001).

2.3 Resource allocation analysis

Resource allocation models were proposed to reveal how the cell allocates its resources among proteins to achieve a maximal growth rate (Goelzer and Fromion, 2011; Lerman et al., 2012). Indeed, the idea originates from the hypothesis that cellular growth rate results from the trade-off between the investments among enzymes, ribosomal proteins, structural components and catabolic pathways. It is even demonstrated that allocating the resources and energy plays an essential role during cellular growth since giving the resources to one biological process would reduce the amounts available for the others. In this section, we aim to detail cellular resource allocation principles and formalisms with the metabolic-genetic network inspired by a self-replicator model designed by Molenaar and coworkers (Molenaar et al., 2009). Compared with classical metabolic networks, it includes the protein expression costs and a detailed relationship between genotype and phenotype.

2.3.1 Metabolic-genetic networks

Cellular metabolism is a system of chemical reactions that on one hand converts the external nutrients into building blocks, while on the other hand, builds macromolecules, especially enzymes, to speed up the reactions. It integrates two kinds of metabolism: catabolism and anabolism. Catabolism aims at breaking down the nutrients into small precursors, e.g. amino acids and ATP, and providing the building blocks for the macromolecules. It also has some small recyclable intermediate metabolites such as hydrogen ion, ADP, NADH etc. Anabolism then uses these precursors to assemble macromolecules which drive the catabolism conversely. At the same time, by-products are frequently secreted. For mathematical modeling, such processes can be abstracted to a simplified network shown in Figure 2.2, which is used to introduce the structure of *metabolic-genetic network* models.

Specifically, $\mathcal{Y}$ is a set including all the external metabolite species that are transported by exchange reactions $\mathcal{R}_{\mathcal{Y}}$. $\mathcal{R}_{\mathcal{X}}$ represents the set of reactions that conduct the conversion between internal metabolites $\mathcal{X}$, which are used to assemble the macromolecules including all the enzymes, ribosomes, transporters $\mathcal{T}$ and non-catalytic macromolecules, named quota compounds $\mathcal{Q}$. At the same time, enzymes and transporters are utilized to catalyze all the intermediate reactions and exchange reactions. The production of proteins requires the binding of ribosomes with messenger RNA. The ribosomes, to some extent, can be considered as the enzyme for

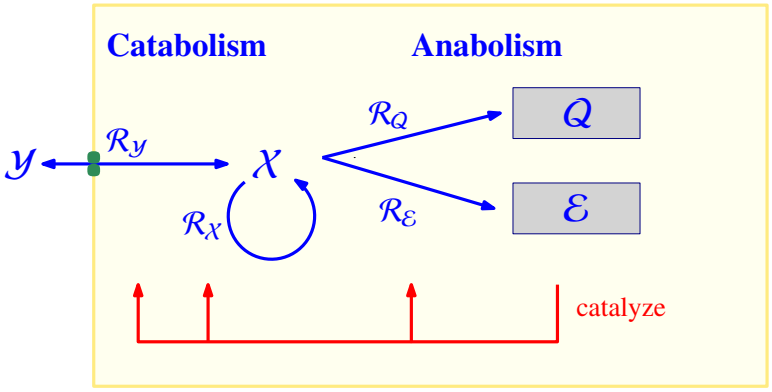


Figure 2.2: A schematic metabolic-genetic network. $\mathcal{Y}$ represents extracellular metabolites with the associated concentrations $\mathbf{Y}$. $\mathcal{R}_{\mathcal{Y}}$ is the set of exchange reactions between extracellular species $\mathcal{Y}$ and intermediate metabolites $\mathcal{X}$ with the fluxes $\mathbf{v}_{\mathcal{Y}}$. $\mathbf{X}$ represents the concentrations of intermediate species. $\mathcal{R}_{\mathcal{X}}$ is a set of intermediate reactions with the associated fluxes $\mathbf{v}_{\mathcal{X}}$. $\mathcal{E}$ is a set of all the catalytic molecules and $\mathbf{E}$ is a vector of their abundances. $\mathcal{Q}$ denotes a set of quota compounds with $\mathbf{Q}$ of abundances. $\mathcal{R}_{\mathcal{E}}, \mathcal{R}_{\mathcal{Q}}, \mathbf{v}_{\mathcal{E}}, \mathbf{v}_{\mathcal{Q}}$ are the corresponding production reactions and fluxes.

all protein production reactions, including ribosomes themselves. For simplicity, we use $\mathcal{E}$ to represent all the catalytic molecules including all the enzymes, ribosomes and transporters.

The metabolic-genetic network model is constructed using several types of metabolic species, reactions and parameters. The abbreviations and notation are as follows:

- the set of extracellular metabolites $\mathcal{Y}$ with the associated concentrations $\mathbf{Y}$. $\mathcal{N} \subseteq \mathcal{Y}$ denotes the nutrients with the associated concentrations $\mathbf{N}$,
- the set of intermediate metabolites $\mathcal{X}$ with the corresponding concentrations $\mathbf{X}$,
- the set of quota compounds $\mathcal{Q}$, such as lipids, DNA, RNA etc., with concentrations $\mathbf{Q}$,
- the set of catalytic molecules $\mathcal{E}$ with concentrations $\mathbf{E}$ for catalyzing chemical reactions $\mathcal{R}$, with $\mathcal{T} \subseteq \mathcal{E}$ the set of transporters for exchange reactions $\mathcal{R}_{\mathcal{Y}}$.

We define the set of macromolecules $\mathcal{P} = \mathcal{E} \cup \mathcal{Q}$ with $\mathbf{P}$ representing the concentrations. The species here can also be measured by molar amounts, depending on the modeler, while units have to be consistent within the approach. The set of reactions is denoted by $\mathcal{R}$ with the relevant flux set $\mathbf{v}$. The reactions are classified into three types:

- the set of exchange reactions $\mathcal{R}_{\mathcal{Y}}$ with the flux set $\mathbf{v}_{\mathcal{Y}}$, in which $\mathcal{R}_{\mathcal{N}}$ represents the uptake reactions for nutrients with fluxes $\mathbf{v}_{\mathcal{N}}$,

- the set of intermediate reactions $\mathcal{R}_\mathcal{X}$ with the flux set $\mathbf{v}_\mathcal{X}$,
- the set of macromolecule synthesis reactions $\mathcal{R}_\mathcal{P}$ with the fluxes $\mathbf{v}_\mathcal{P}$, in which $\mathcal{R}_\mathcal{E} \subseteq \mathcal{R}_\mathcal{P}$ is the set of reactions producing $\mathcal{E}$ with fluxes $\mathbf{v}_\mathcal{E}$, and $\mathcal{R}_Q \subseteq \mathcal{R}_\mathcal{P}$ is the set of reactions producing Q with fluxes $\mathbf{v}_Q$.

Parameters are

- $k_{cat}^{i,j}$: turnover rate of enzyme j catalyzing reaction i,
- $cat(E) := \{i \in \mathcal{R} \mid \text{enzyme } E \text{ catalyzes reaction } i\}$,
- $S_{i,j}$ describing the stoichiometric coefficient of metabolite i in reaction j,
- K_M^i denoting the Michaelis constant giving the concentration of substrate i needed to reach half of the maximal reaction rate.

2.3.2 Steady-state assumption during growth phase

During the exponential growth phase, microbes grow and divide as fast as they can. A significant phenotypic feature during this period is that the biomass and the volume of the microbial population increase exponentially with a growth rate μ measured by a unit of time, such as 1/min. Particularly, as the major components of biomass, proteins have to be produced along with the growth rate to support the exponential growth. Degradation is normally ignored since the degradation rate is negligible compared to the high growth rate.

If we measure the macromolecule abundances by amounts, e.g. mmol, they are obviously increasing with the production rates. However, if we measure the macromolecular abundances as concentrations, e.g. g/l and mmol/gDW, they are directly diluted after the production by enlarging cellular volume or biomass. In other words, the macromolecular concentrations remain stable if we model the metabolites using concentrations. Therefore, the significant traits of exponential growth phase are summarized below:

- the metabolic transients are extremely rapid so that all the intermediate metabolites are assumed to be at steady-state;
- each macromolecule amount, cellular size, total biomass and population size are assumed to increase exponentially according to the growth rate;
- the concentrations of macromolecules also remain constant because the increased amounts are diluted by the increasing cellular volume or biomass.

Instead of a biomass reaction composed of certain coefficients of metabolites in FBA-type models (see Figure 2.1), resource allocation models define the growth

rate either as the exponential increase of biomass, or by the dilution rate of macromolecule concentrations. Considering the dilution by growth, the dynamics of macromolecular concentrations can be expressed as

$$\frac{d\mathbf{P}}{dt} = S_{\mathcal{P},\mathcal{R}_\mathcal{P}} \cdot \mathbf{v}_\mathcal{P} - \mu \cdot \mathbf{P}, \tag{2.11}$$

where $\mathbf{P}$ represents the vector of macromolecular concentrations in $\mathcal{P}$. $S_{\mathcal{P},\mathcal{R}_\mathcal{P}}$ is the stoichiometric matrix of macromolecules $\mathcal{P}$ in reactions $\mathcal{R}_\mathcal{P}$. We can see that the macromolecular concentrations increase with the production rate and decrease with the dilution rate. At steady-state, each macromolecular concentration remains constant because the produced molecules are directly diluted by enlarging the volume or biomass, written as

$$S_{\mathcal{P},\mathcal{R}_\mathcal{P}} \cdot \mathbf{v}_\mathcal{P} - \mu \cdot \mathbf{P} = 0. \tag{2.12}$$

To avoid the computation of cellular volume and to maintain consistency with FBA-type approaches, the unit of macromolecular concentrations is suggested to be mmol/gDW instead of mmol/l.

2.3.3 Resource allocation constraints

Based on the classical metabolic networks, resource allocation models additionally consider protein production costs besides the stoichiometry and thermodynamics of biochemical reactions. Using the abbreviations and notation of metabolic-genetic network shown in Figure 2.2, we formulate the basic biological principles as mathematical constraints to predict the optimal resource allocation.

Metabolic capability constraints

The metabolic capability constraints ensure that the metabolic network has the ability to provide sufficient precursors for macromolecules. Enough intermediate metabolites $\mathcal{X}$ must be produced to run all the macromolecule productions. Due to the steady-state assumption for intermediate metabolites, the constraint can be expressed as

$$S_{\mathcal{X},\mathcal{R}_\mathcal{Y}} \cdot \mathbf{v}_\mathcal{Y} + S_{\mathcal{X},\mathcal{R}_\mathcal{X}} \cdot \mathbf{v}_\mathcal{X} + S_{\mathcal{X},\mathcal{R}_\mathcal{P}} \cdot \mathbf{v}_\mathcal{P} = 0, \tag{2.13}$$

where $S_{\mathcal{X},\mathcal{R}_\mathcal{Y}}, S_{\mathcal{X},\mathcal{R}_\mathcal{X}}, S_{\mathcal{X},\mathcal{R}_\mathcal{P}}$ are the stoichiometric matrices, in which each row represents an intermediate metabolite in $\mathcal{X}$ and each column is an individual reaction in the set of exchange reactions $\mathcal{R}_\mathcal{Y}$, the set of intermediate reactions $\mathcal{R}_\mathcal{X}$ or the set of macromolecular synthesis reactions $\mathcal{R}_\mathcal{P}$. $\mathbf{v}_\mathcal{Y}$, $\mathbf{v}_\mathcal{X}$ and $\mathbf{v}_\mathcal{P}$ are the corresponding fluxes.

At steady-state, the metabolic network capability has to be large enough to main the stable macromolecule concentrations. Specifically, with the schematic metabolic-genetic network in Figure 2.2, we obtain

$$S_{\mathcal{E},\mathcal{R}_\mathcal{E}} \cdot \mathbf{v}_\mathcal{E} - \mu \cdot \mathbf{E} = 0, \tag{2.14}$$

$$S_{\mathcal{Q},\mathcal{R}_\mathcal{Q}} \cdot \mathbf{v}_\mathcal{Q} - \mu \cdot \mathbf{Q} = 0, \tag{2.15}$$

where $S_{\mathcal{E},\mathcal{R}_\mathcal{E}}, S_{\mathcal{Q},\mathcal{R}_\mathcal{Q}}$ are the stoichiometric matrices in which each row denotes a macromolecule in $\mathcal{E}$ and $\mathcal{Q}$, and each column a macromolecular synthesis reaction in reaction sets $\mathcal{R}_\mathcal{E}$ and $\mathcal{R}_\mathcal{Q}$. $\mathbf{v}_\mathcal{E}, \mathbf{v}_\mathcal{Q}$ are the relevant fluxes.

Enzymatic and translational capacity constraints

The performance of the metabolic machinery is determined not only by the nutrients but also by the catalytic macromolecules. According to Michaelis-Menten kinetics, the maximal reaction rate v_{max} at saturating substrate concentration is given by $v_{max} = k_{cat}E$, where E is the enzyme concentration and k_{cat} is the catalytic constant. Therefore, the metabolic reaction fluxes are bounded by the related enzyme abundances and the associated turnover rates.

For the reactions sharing the same enzyme, the sum of all the reaction fluxes is limited by the enzyme concentration. For example, ribosomes can be considered as the shared enzyme for all the protein production reactions, which includes ribosomal proteins themselves. Thus, the enzymatic and translational capacity constraints can be formalized as

$$\sum_{i \in cat(E)} \frac{|\mathbf{v}_i|}{k_{cat}^{i,E}} \leq E, \quad \text{for all } E \in \mathcal{E} \tag{2.16}$$

where $\mathbf{v}_i$ represents the flux of reaction $\mathcal{R}_i$ catalyzed by enzyme E. E also denotes the enzyme concentration. $k_{cat}^{i,E}$ is the corresponding catalytic constant, and $cat(E)$ the set of reactions i catalyzed by enzyme E.

The external nutrients are not at steady-state. They are consumed due to the cellular growth. Naturally, their uptake rates are also limited by the associated nutrient concentrations, in addition to the transporters. Thus, an additional constraint is imposed for the uptake reactions catalyzed by the transporters $\mathcal{T}$. By the convention that the uptake fluxes are negative, the uptake reaction flux of a nutrient $\mathcal{N}_i$ is extra constrained by the Michaelis–Menten equation

$$\mathbf{v}_i = -\frac{\mathbf{N}_i}{K_M^i + \mathbf{N}_i} \cdot k_{cat}^{i,T} \cdot T, \quad \text{for all } i \in \mathcal{R}_\mathcal{N}, T \in \mathcal{T}, \tag{2.17}$$

where $k_{cat}^{i,T}$ is the catalytic rate of transporter T for uptake reaction i. K_M^i represents the half-saturation constant. $\mathbf{N}_i$ is the concentration of the nutrient converted through exchange flux $\mathbf{v}_i$. By convention, the uptake fluxes of nutrients are negative. Nutrient concentrations $\mathbf{N}$ are fixed as an environmental condition for static resource allocation approaches, such as RBA. T is the corresponding transporter concentration.

Density constraint

A cell can not grow without limit. The cell volume has a limitation since it is composed of the membrane. This means that the sum of all macromolecular concentrations is bounded by a maximal density. Therefore, a cellular density constraint is

introduced to ensure suitable macromolecular concentrations. It can be formulated as

$$\mathbf{d}_{\mathcal{P}}^{T} \cdot \mathbf{P} \leq D. \tag{2.18}$$

Here, D is a constant value representing the maximal density, $\mathbf{P}$ denotes the vector of macromolecular concentrations while $\mathbf{d}_{\mathcal{P}}$ is a vector with the density coefficients of macromolecules. The operation $\cdot^{T}$ denotes transposition.

Fixing quota compound concentrations

Quota compounds, e.g. DNA, the lipid membrane, or the cell wall, are essential to the cellular growth and survival. Their concentrations have been found to be independent of growth rate (Marr, 1991). Since these macromolecules do not have any catalytic functionalities and only require resources, maximizing the growth rate results in allocating few resources to them. The authors of RBA (Goelzer et al., 2011) proposed to fix quota compound concentrations $\mathbf{Q}$ for running RBA. Therefore, we obtain

$$\mathbf{Q} = \mathbf{Q}_{fix}, \tag{2.19}$$

where $\mathbf{Q}_{fix}$ is a vector with the fixed concentrations of quota compounds in $\mathcal{Q}$.

2.3.4 Non-linear optimization problem

All the constraints presented above make up the feasible space for cellular metabolism capable of supporting a growth rate μ. Adding an objective function maximizing the growth rate, we obtain a non-linear optimization problem predicting resource allocation,

$$\begin{aligned}
\max_{\mathbf{v},\mathbf{E},\mu} \quad & \mu \\
s.t. \quad & S_{\mathcal{X},\mathcal{R}_{\mathcal{Y}}} \cdot \mathbf{v}_{\mathcal{Y}} + S_{\mathcal{X},\mathcal{R}_{\mathcal{X}}} \cdot \mathbf{v}_{\mathcal{X}} + S_{\mathcal{X},\mathcal{R}_{\mathcal{P}}} \cdot \mathbf{v}_{\mathcal{P}} = 0, \\
& S_{\mathcal{E},\mathcal{R}_{\mathcal{E}}} \cdot \mathbf{v}_{\mathcal{E}} - \mu \cdot \mathbf{E} = 0, \\
& S_{\mathcal{Q},\mathcal{R}_{\mathcal{Q}}} \cdot \mathbf{v}_{\mathcal{Q}} - \mu \cdot \mathbf{Q} = 0, \\
& \sum_{i \in cat(E)} \frac{|\mathbf{v}_i|}{k_{cat}^{i,E}} \leq E, \quad \text{for all } E \in \mathcal{E} \setminus \mathcal{T} \\
& \mathbf{v}_i = -\frac{\mathbf{N}_i}{K_M^i + \mathbf{N}_i} \cdot k_{cat}^{i,T} \cdot T, \quad \text{for all } i \in \mathcal{R}_{\mathcal{N}},\ T \in \mathcal{T} \\
& \mathbf{d}_{\mathcal{P}}^{T} \cdot \mathbf{P} \leq D, \\
& \mathbf{Q} = \mathbf{Q}_{fix}, \\
& \mathbf{P}, \mu \geq 0,
\end{aligned} \tag{2.20}$$

where $\mathbf{N}$ is the vector of the nutrient concentrations given as the environmental condition. In (Goelzer et al., 2011), RBA authors relaxed the metabolic capacity

constraints in Eq. 2.13 to

$$S_{\mathcal{X},\mathcal{R}_\mathcal{Y}} \cdot \mathbf{v}_\mathcal{Y} + S_{\mathcal{X},\mathcal{R}_\mathcal{X}} \cdot \mathbf{v}_\mathcal{X} + S_{\mathcal{X},\mathcal{R}_\mathcal{P}} \cdot \mathbf{v}_\mathcal{P} \geq 0, \tag{2.21}$$

to allow the overproduction of intermediate metabolites. Constraints in Eq. 2.14 and Eq. 2.15 were also consistently relaxed to impose that the macromolecules shall increase, at least, by the growth rate μ, leading to

$$S_{\mathcal{E},\mathcal{R}_\mathcal{E}} \cdot \mathbf{v}_\mathcal{E} - \mu \cdot \mathbf{E} \geq 0, \tag{2.22}$$

$$S_{\mathcal{Q},\mathcal{R}_\mathcal{Q}} \cdot \mathbf{v}_\mathcal{Q} - \mu \cdot \mathbf{Q} \geq 0. \tag{2.23}$$

Solving the non-linear program by bisection

We can clearly see that the optimization problem in Eq. 2.20 aims at obtaining a flux distribution $\mathbf{v}$ and macromolecular concentrations $\mathbf{P}$, as well as the optimal growth rate μ. Under a given condition that the nutrient concentrations $\mathbf{N}$ are fixed, however, the optimization problem shown in Eq. 2.20 is still a quadratic and non-linear optimization problem due to the item $\mu \cdot \mathbf{E}$. Since there have been no efficient algorithms to solve such a non-linear program, Goelzer and co-authors applied the bisection idea to convert it into a linear program by fixing μ iteratively (Goelzer et al., 2011).

The basic idea is when μ is fixed, the optimization problem illustrated in Eq. 2.20 denoted by RBA(μ) is equivalent to a linear program and then can be solved by efficient optimization solvers such as Gurobi (`http://www.gurobi.com`), Cplex (`https://www.ibm.com/products/ilog-cplex-optimization-studio`), etc. First, μ must be given a small value. Keep doubling μ and solving for RBA(2μ), until no feasible solution of RBA(2μ) can be found. The maximal growth rate μ must be in the interval $[\mu, 2\mu]$. Next, repeatedly split the interval and fix μ to the midpoint, and then check if the linear program is feasible. At the end, the maximal growth rate will be fixed in a sufficiently small interval. When the subinterval is even smaller than the tolerance δ, the midpoint is considered as the maximal growth rate. Details of this bisection strategy are shown in Algorithm 1. The tolerance can be regarded as the precision of the problem, which should be given initially.

2.3.5 Discussion

Cellular growth requires a balance between resource costs for various functional macromolecules and the corresponding benefits, an issue that is normally ignored in classic constraint-based modeling approaches. In RBA, the authors introduced macromolecule productions and their functionalities with extra linear constraints such as enzymatic and translational capability constraints. Converting the quadratic optimization into a linear program via bisection, RBA allows the efficient prediction of flux distribution and enzyme concentrations with the maximal growth

Algorithm 1: Algorithm to search a maximal μ with the precision δ.

```
Data: RBA(μ), δ
Result: maximal μ
μ = δ
while exist a feasible solution RBA(2μ) do
  | μ = 2μ
end
Δμ = μ/2
while Δμ > δ do
  | if exist a feasible RBA(μ + Δμ) then
  |   | μ = μ + Δμ
  | end
  | Δμ = Δμ/2
end
```

rate. To build an RBA model, information on macromolecule composition, kinetic parameters and realistic quota concentration has to be collected.

Another issue to be discussed here is the extension of resource allocation models, such as RBA and ME-models. As RBA and ME-models are static, one idea is to incorporate a time scale into the resource allocation analysis to predict the dynamics of cellular metabolism while taking into account enzyme costs. One possibility is to iterate resource allocation analysis step by step like in the static approach of DFBA. This is basically the idea of dynamicME method and iRBA, as detailed in Chapter 4. Another way is based on dynamic optimization. This is used in deFBA (Waldherr et al., 2015), which we describe next.

2.4 Dynamic enzyme-cost flux balance analysis

Dynamic enzyme-cost flux balance analysis (deFBA) was proposed to predict the optimal dynamics of metabolism and resource allocation. Compared to the static and iterative approaches to predict resource allocation, deFBA models metabolism with pure continuous variables while developing an optimal control algorithm to maximize the total biomass over the whole simulation time period. By integrating dynamically the metabolic network with enzyme production costs, deFBA enables the prediction of optimal reaction fluxes and the dynamics of macromolecule amounts. Moreover, by discretizing the continuous variables in time, deFBA problems can be transformed into linear optimization problems.

2.4.1 Metabolic constraints and objective

In contrast to RBA and ME-models, macromolecules are modeled by molar amounts with unit mmol in deFBA. As discussed above, macromolecular concentrations do not change during the exponential growth because of the dilution

with cellular volume or biomass. However, the molar amounts of macromolecules should always be evolving with the corresponding production rates. Therefore, in deFBA, external metabolites and macromolecules evolve over time and, of course, they are not at steady-state. However, for all the intermediate metabolites such as building blocks, deFBA assumes they are at steady-state. That means they are immediately used to build macromolecules or converted to other precursors after they have been produced. Like RBA, the mathematical formalization is built by converting the biological laws into linear constraints. Using the schematic metabolic-genetic network (shown in Figure 2.2) and the related abbreviations, the metabolic processes are illustrated together with the corresponding constraints below.

Dynamics of external species

In deFBA, the initial amounts of external metabolites, especially the nutrient amounts, are specified by the modeler. That is

$$\mathbf{Y}(t_0) = \mathbf{Y}_0, \tag{2.24}$$

where $\mathbf{Y}_0$ represents the initial amounts of extracellular species and t_0 indicates the starting time point when running deFBA. As the cell grows by converting the nutrients into the precursors and macromolecules, the external nutrients are continuously depleted and by-products are secreted. So, the external metabolite amounts $\mathbf{Y}$ are changing with the exchange reaction fluxes. Mathematically, this can be expressed as

$$\dot{\mathbf{Y}}(t) = S_{\mathcal{Y},\mathcal{R}_\mathcal{Y}} \cdot \mathbf{v}_\mathcal{Y}(t). \tag{2.25}$$

$S_{\mathcal{Y},\mathcal{R}_\mathcal{Y}}$ is the stoichiometric matrix, in which the rows represent extracellular metabolite species $\mathcal{Y}$ and the columns the exchange reactions $\mathcal{R}_\mathcal{Y}$. $\mathbf{v}_\mathcal{Y}(t)$ is the vector of exchange reaction fluxes at time point t.

Dynamics of macromolecules

Macromolecules are assembled from the building blocks to ensure cellular survival and growth. The total biomass of a cell is assumed to be the sum of all the macromolecules which are accumulated as the cell grows. As such, the derivatives of amounts $\mathbf{P}(t)$ depend on the production reaction fluxes (assumed to be positive). That is

$$\dot{\mathbf{P}}(t) = S_{\mathcal{P},\mathcal{R}_\mathcal{P}} \cdot \mathbf{v}_\mathcal{P}(t), \tag{2.26}$$

where $S_{\mathcal{P},\mathcal{R}_\mathcal{P}}$ is the stoichiometric matrix, in which the rows represent macromolecules $\mathcal{P}$ and the columns the synthesis reactions $\mathcal{R}_\mathcal{P}$. $\mathbf{v}_\mathcal{P}$ is the flux vector of the synthesis reactions. The initial values for $\mathbf{P}(t_0)$ can be given by modeler, i.e.,

$$\mathbf{P}(t_0) = \mathbf{P}_0. \tag{2.27}$$

Additionally, instead of specifying the initial amounts of individual macromolecules, the total initial biomass can be constrained within certain values, e.g. $\text{Bio}(t_0) = 1\text{g}$ (Reimers et al., 2017b). However, as the macromolecule amounts are modeled by molar amounts in deFBA, a vector of molecular weights has to be introduced to convert the molar amounts of macromolecules into their masses. That is

$$\mathbf{b}_{\mathcal{P}}^T \cdot \mathbf{P}(t_0) = \text{Bio}(t_0), \tag{2.28}$$

where $\mathbf{b}_{\mathcal{P}}$ is a vector of molecular weights of all the macromolecules $\mathcal{P}$ with the associated unit mg/mol. The operation $\cdot^T$ denotes transposition.

Steady state for intermediate metabolites

Unlike the extracellular species, all the intermediate metabolites must be balanced since they are used to build macromolecules immediately after they are broken down from the nutrients. Here, all the intermediate metabolites are assumed to be at a steady-state during the whole growth period. This can be expressed as

$$S_{\mathcal{X},\mathcal{R}_\mathcal{Y}} \cdot \mathbf{v}_\mathcal{Y}(t) + S_{\mathcal{X},\mathcal{R}_\mathcal{X}} \cdot \mathbf{v}_\mathcal{X}(t) + S_{\mathcal{X},\mathcal{R}_\mathcal{P}} \cdot \mathbf{v}_\mathcal{P}(t) = 0, \tag{2.29}$$

where $S_{\mathcal{X},\mathcal{R}_\mathcal{Y}}, S_{\mathcal{X},\mathcal{R}_\mathcal{X}}, S_{\mathcal{X},\mathcal{R}_\mathcal{P}}$ are the stoichiometric matrices, in which the rows represent intermediate metabolites $\mathcal{X}$ and the columns are the reactions in $\mathcal{R}_\mathcal{Y}, \mathcal{R}_\mathcal{X}$ and $\mathcal{R}_\mathcal{P}$ respectively. $\mathbf{v}_\mathcal{Y}(t), \mathbf{v}_\mathcal{X}(t), \mathbf{v}_\mathcal{P}(t)$ are the fluxes at time point t.

Biomass composition constraint

Quota compounds without catalytic property, such as lipids and DNA, are indispensable for cellular growth and replication. In order to guarantee the production of non-catalytic compounds $\mathcal{Q}$, deFBA imposes a biomass composition constraint stating that the total mass of quota compounds has to be at least $\Phi_\mathcal{Q}$ of the total biomass at each time point t, where $0 < \Phi_\mathcal{Q} < 1$, e.g. $\Phi_\mathcal{Q} = 35\%$. The total biomass at each time point t is defined by summing up all the masses of macromolecules. Mathematically, we have a constraint

$$\mathbf{b}_\mathcal{Q}^T \cdot \mathbf{Q}(t) \geq \Phi_\mathcal{Q} \cdot \mathbf{b}_\mathcal{P}^T \cdot \mathbf{P}(t), \tag{2.30}$$

where $\mathbf{b}_\mathcal{P}$ is a vector with the molecular weights of all the macromolecules $\mathcal{P}$, $\mathbf{b}_\mathcal{Q}$ is the subvector of the molecular weights of the quota compounds $\mathcal{Q}$, the operation $\cdot^T$ denotes transposition, and $0 < \Phi_\mathcal{Q} < 1$ is a constant.

Enzymatic and translational capacity constraints

As shown in Figure 2.2, catalytic macromolecules $\mathcal{E}$ are used to catalyze the related reactions. Similar to RBA and ME-models, deFBA includes enzymatic and translational capacity constraints that bound the fluxes of catalyzed reactions. If one enzyme catalyzes more than one reaction, the sum of all the reaction fluxes is

limited by the enzyme amount and the catalytic rates. Similar to the enzymatic and translation capacity constraints in RBA, such constraints in deFBA are mathematically expressed by

$$\sum_{i \in cat(E)} \frac{|\mathbf{v}_i(t)|}{k_{cat}^{i,E}} \leq E(t), \quad \text{for all } E \in \mathcal{E}, \tag{2.31}$$

where $cat(E)$ represents the catalytic relationship between enzyme E and the related reaction i, $k_{cat}^{i,E}$ is the relevant turnover rate and E also represents the enzyme amount.

Note that deFBA does not impose any extra constraints on the exchange reaction rates by Michaelis-Menten kinetics as RBA does by Eq. 2.17. If the exchange reaction fluxes are constrained with Michaelis-Menten kinetics, the optimization program would be non-linear. To avoid the non-linearity problem, deFBA directly uses Eq. 2.31 to constrain the fluxes of all the catalyzed reactions.

Objective: maximizing the biomass production over a time interval

Up till now, we have presented the metabolic constraints of deFBA. In order to predict the dynamic flux distribution and macromolecular amounts, a particular objective function has to be defined to build an optimization problem. deFBA has considered three different objectives in (Waldherr et al., 2015), which are the maximization of biomass at the end of the simulation period, the minimization of the time required to exhaust the given nutrients, and the maximization of the total biomass over a simulation time span. Although deFBA is not limiting the objective function, maximizing the biomass over a simulation time span is widely employed. So, for a simulation time span $[t_0, t_f]$, the biomass integral is

$$\int_{t_0}^{t_f} \mathbf{b}_{\mathcal{P}}^T \cdot \mathbf{P}(t) dt, \tag{2.32}$$

where $\mathbf{P}$ denotes all the macromolecular amounts in the model and $\mathbf{b}_{\mathcal{P}}^T$ is the corresponding vector of molecular weights that is introduced to convert the molar amounts into biomass.

2.4.2 Formulation of deFBA

Therefore, deFBA can be formulated as a dynamic optimization problem maximizing the biomass production over a simulation time interval $[t_0, t_f]$ in Figure 2.3. We see that the continuous variables in deFBA include molar amounts $\mathbf{Y}(t)$, $\mathbf{P}(t)$, their derivatives $\dot{\mathbf{Y}}(t), \dot{\mathbf{P}}(t)$, and the fluxes $\mathbf{v}(t)$.

2.4.3 Discretization of the variables in time

An optimization problem as shown in Figure 2.3 has to be solved numerically and there are several methods available. deFBA problems are solved based on the

$$
\begin{aligned}
\max_{\mathbf{Y}(t),\dot{\mathbf{Y}}(t),\mathbf{P}(t),\dot{\mathbf{P}}(t),\mathbf{v}(t)} \quad & \int_{t_0}^{t_f} \mathbf{b}_{\mathcal{P}}^T \cdot \mathbf{P}(t)dt \\
s.t. \quad & S_{\mathcal{X},\mathcal{R}_\mathcal{Y}} \cdot \mathbf{v}_\mathcal{Y}(t) + S_{\mathcal{X},\mathcal{R}_\mathcal{X}} \cdot \mathbf{v}_\mathcal{X}(t) + S_{\mathcal{X},\mathcal{R}_\mathcal{P}} \cdot \mathbf{v}_\mathcal{P}(t) = 0, \\
& \dot{\mathbf{P}}(t) = S_{\mathcal{P},\mathcal{R}_\mathcal{P}} \cdot \mathbf{v}_\mathcal{P}(t), \\
& \dot{\mathbf{Y}}(t) = S_{\mathcal{Y},\mathcal{R}_\mathcal{Y}} \cdot \mathbf{v}_\mathcal{Y}(t), \\
& \sum_{i \in cat(E)} \frac{|\mathbf{v}_i(t)|}{k_{cat}^{i,E}} \leq E(t), \quad \text{for all } E \in \mathcal{E} \\
& \mathbf{b}_\mathcal{Q}^T \cdot \mathbf{Q}(t) \geq \Phi_\mathcal{Q} \cdot \mathbf{b}_\mathcal{P}^T \cdot \mathbf{P}(t), \\
& \mathbf{lb} \leq \mathbf{v}(t) \leq \mathbf{ub}, \\
& \mathbf{Y}(t_0) = \mathbf{Y}_0, \\
& \mathbf{P}(t_0) = \mathbf{P}_0, \\
& \mathbf{Y}(t), \mathbf{P}(t), t \geq 0.
\end{aligned}
$$

Figure 2.3: Formulation of deFBA

discretization of the dynamic variables in time (Reimers, 2017). According to the midpoint rule, a differential equation: $x'(t) = f(x(t), t)$, can be discretized with a step size Δt by

$$x_k = x_{k-1} + \Delta t \cdot f(\frac{x_k + x_{k-1}}{2}, t_{k-1} + \frac{\Delta t}{2}), \tag{2.33}$$

where x_k is the variable at time point t_k.

(Reimers, 2017) suggests to discretize $\mathbf{Y}$ and $\mathbf{P}$ at each time point t_k. The flux variables $\mathbf{v}$ and the derivatives $\dot{\mathbf{P}}, \dot{\mathbf{Y}}$ are discretized at the midpoint $\frac{t_k+t_{k-1}}{2}$. Assuming n discretization points, the dynamic optimization problem in Figure 2.3 can be transformed to the linear optimization problem in Figure 2.4. The dynamic variables are $\mathbf{Y}(t_k)$, $\mathbf{P}(t_k)$, $\mathbf{v}\left(\frac{t_k+t_{k-1}}{2}\right)$, $\dot{\mathbf{Y}}\left(\frac{t_k+t_{k-1}}{2}\right)$ and $\dot{\mathbf{P}}\left(\frac{t_k+t_{k-1}}{2}\right)$, $k = 1, \ldots, n$.

2.5 Conclusions

To summarize, the approaches explained above extend classic flux balance analysis in three different directions: (1) dynamics; (2) whether regulation is included or not; (3) if enzyme production costs are considered. DFBA solves a static FBA problem iteratively accounting for dynamically changing environments. rFBA additionally integrates Boolean regulatory rules that impose the logical control of genetic regulation on the metabolic reaction activities. Considering enzyme production costs, metabolic-genetic networks allow studying resource allocation, either at steady-state (RBA, ME-models) or in a dynamic setting (deFBA).

Given these developments, there is the need for a general mathematical framework

$$
\begin{aligned}
\max_{\mathbf{P},\dot{\mathbf{P}},\mathbf{v},\mathbf{Y},\dot{\mathbf{Y}}} \quad & \sum_{k=1}^{n} \mathbf{b}_{\mathcal{P}}^{T} \cdot \mathbf{P}\,(t_k) \\
s.t. \quad & S_{\mathcal{X},\mathcal{R}_{\mathcal{Y}}} \cdot \mathbf{v}_{\mathcal{Y}}\left(\frac{t_k + t_{k-1}}{2}\right) + S_{\mathcal{X},\mathcal{R}_{\mathcal{X}}} \cdot \mathbf{v}_{\mathcal{X}}\left(\frac{t_k + t_{k-1}}{2}\right) + S_{\mathcal{X},\mathcal{R}_{\mathcal{P}}} \cdot \mathbf{v}_{\mathcal{P}}\left(\frac{t_k + t_{k-1}}{2}\right) = 0, \\
& \dot{\mathbf{P}}\left(\frac{t_k + t_{k-1}}{2}\right) = S_{\mathcal{P},\mathcal{R}_{\mathcal{P}}} \cdot \mathbf{v}_{\mathcal{P}}\left(\frac{t_k + t_{k-1}}{2}\right), \\
& \dot{\mathbf{Y}}\left(\frac{t_k + t_{k-1}}{2}\right) = S_{\mathcal{Y},\mathcal{R}_{\mathcal{Y}}} \cdot \mathbf{v}_{\mathcal{Y}}\left(\frac{t_k + t_{k-1}}{2}\right), \\
& \sum_{i \in cat(E)} \frac{|\mathbf{v}_i\left(\frac{t_k + t_{k-1}}{2}\right)|}{k_{cat}^{i,E}} \leq \frac{E(t_k) + E(t_{k-1})}{2}, \quad \text{for all } E \in \mathcal{E} \\
& \mathbf{b}_{\mathcal{Q}}^{T} \cdot \mathbf{Q}(t_k) \geq \Phi_{\mathcal{Q}} \cdot \mathbf{b}_{\mathcal{P}}^{T} \cdot \mathbf{P}(t_k), \\
& \mathbf{P}(t_k) = \mathbf{P}(t_{k-1}) + \Delta t \cdot \dot{\mathbf{P}}\left(\frac{t_k + t_{k-1}}{2}\right), \\
& \mathbf{Y}(t_k) = \mathbf{Y}(t_{k-1}) + \Delta t \cdot \dot{\mathbf{Y}}\left(\frac{t_k + t_{k-1}}{2}\right), \\
& \mathbf{lb} \leq \mathbf{v}\left(\frac{t_k + t_{k-1}}{2}\right) \leq \mathbf{ub}, \\
& \mathbf{Y}(t_1) = \mathbf{Y}_0, \;\; \mathbf{P}(t_1) = \mathbf{P}_0, \\
& \mathbf{P}(t_k), \mathbf{Y}(t_k) \geq 0, \\
& \text{for all} \quad k = 1, \ldots, n.
\end{aligned}
$$

Figure 2.4: LP for solving deFBA model

to predict the dynamics of cellular metabolism integrating protein expression and regulatory events. This will be further addressed in the following chapters.

Chapter 3

Exploring the optimal solution space in rFBA

Standard *regulatory flux balance analysis* (rFBA) leads to a unique prediction of dynamic growth, ignoring possible alternative solutions. In this chapter, we propose an analytic pipeline that first characterizes the optimal solution space in rFBA by calculating vertices, rays, and linealities. Next, in order to explore the *elementary flux modes* (EFMs) on which the cell is growing, and how these combine with one another at optimal growth, a mixed-integer program is used to decompose the vertices of the optimal solution space to a minimal set of EFMs. In two case studies, the analytic pipeline is applied to a core carbon network and the *Escherichia coli* central network. We describe how the EFMs combine and how they shift upon the change of external substrate concentrations. Section 3.2 and Section 3.3 has been accepted for publication in the *Proceedings of the Evry Spring School on advances in systems and synthetic biology, 2018*.

3.1 Introduction

Methods that integrate the transcriptome with metabolism enable us to better understand how transcriptional regulation is leading to flux responses at the metabolic level (Covert et al., 2001; Samal and Jain, 2008; Covert et al., 2008; Chandrasekaran and Price, 2010). *Regulatory flux balance analysis* (rFBA) (Covert et al., 2001; Covert and Palsson, 2002, 2003) has been proposed to predict the dynamics of metabolism by combining a stoichiometric model of metabolism with Boolean rules for transcriptional regulation. Genome-scale rFBA models have been developed for *Escherichia coli* (*E. coli*) (Covert et al., 2004) and *Saccharomyces cerevisiae* (*S. cerevisiae*) (Herrgård et al., 2006).

As explained in Section 2.2, the basic idea of rFBA is to iterate *flux balance analysis* (FBA) in discrete time steps. At each step, genes resp. reactions are knocked out according to the regulatory rules. These are represented by Boolean-valued functions that depend on the external metabolite concentrations and the reaction fluxes at the previous time step. Since it is based on FBA and linear programming, rFBA computes only one optimal solution at each time step. However, the solution of an FBA problem need not be unique. In general and at each time step, the set of all optimal flux distributions defines a polyhedron. The structure of this polyhedron constrained by the regulatory rules in response to various conditions is not well understood yet.

The polyhedron representing the optimal solution space is expected to change with the regulatory constraints, which are dynamic and depend on the environment. For example, the optimal solution space may be reduced when a certain reaction is inhibited by some regulatory rules from one time step to the next. Some pathways may not be permitted due to the regulatory constraints (Covert and Palsson, 2003). In contrast to the single optimum solution found by executing a linear program in FBA, exploring an optimal space gives a comprehensive description of the set of alternative optima.

In this chapter, we propose an analytic pipeline to explore the optimal solution space in rFBA. First, we characterize the polyhedron through its geometric features, including vertices, rays, and linealities (Kelk et al., 2012). Then, a mixed-integer program is applied to decompose the vertices into minimal sets of *elementary flux modes* (EFMs). Analyzing a core carbon network and the *E. coli* central network, we illustrate the non-decomposable pathways used by the two networks, their combination under certain condition, and the ways they shift upon the changes of external substrates.

3.2 Analytic pipeline

The workflow of the analytic pipeline is shown in Figure 3.1. There are roughly four steps. It begins with a standard rFBA simulation integrating Boolean regulatory rules with the metabolic network. Secondly, *flux variability analysis* (FVA) is used to determine the range of reaction fluxes at each time step (Mahadevan and Schilling, 2003). Thirdly, the optimal polyhedron is characterized by geometric features, including vertices, rays, and linealities (Kelk et al., 2012; Maarleveld et al., 2015). After computing the set of EFMs by EFMtool (Terzer and Stelling, 2008), a mixed-integer linear program is applied to decompose the vertices into a minimal set of EFMs. Standard rFBA is formulated in Eq. 2.10, see Section 2.2 in detail. We start to describe FVA for the analytic pipeline in the following section.

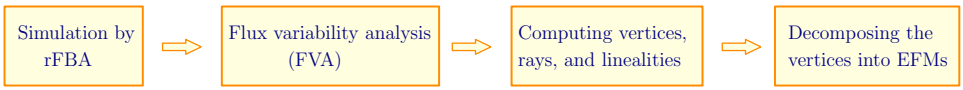


Figure 3.1: Analytic pipeline diagram.

3.2.1 FVA

Flux variability analysis (FVA) analyzes the range of fluxes and alternative optimal solutions achieving the same objective in an FBA problem (Mahadevan and Schilling, 2003). Let z_{opt} be the objective value obtained by FBA. Then FVA can be mathematically described as follows. For all reactions i:

$$\begin{aligned} \max\,/\,\min: \quad & \mathbf{v}_i \\ s.t. \quad & S \cdot \mathbf{v} = 0, \\ & \mathbf{c}^T \cdot \mathbf{v} \geq z_{opt}, \\ & \mathbf{lb} \leq \mathbf{v} \leq \mathbf{ub}, \end{aligned} \tag{3.1}$$

where $\mathbf{c}^T \mathbf{v}$ is the objective function from Eq. 1.11.

3.2.2 Characterizing the optimal solution space

Comprehensive Polyhedron Enumeration Flux Balance Analysis (CoPE_FBA) (Kelk et al., 2012; Maarleveld et al., 2015) is an approach to characterize the polyhedron defined by the optimal solutions of an FBA problem in terms of vertices, rays, and linealities. Vertices correspond to flux vectors, rays to irreversible cycles and linealities to reversible cycles in the network. Any optimal FBA solutions can be written as the sum of a linear combination of linealities, a conic (i.e., non-negative linear) combination of rays, and a convex combination of the vertices according to Minkowski sum (Grötschel et al., 1988):

$$z_{opt} = \sum_{i=1}^{n} \Upsilon_i \cdot \mathbf{lin}_i + \sum_{i=1}^{m} \Psi_i \cdot \mathbf{ray}_i + \sum_{i=1}^{k} \Omega_i \cdot \mathbf{vertex}_i, \tag{3.2}$$

where $\mathbf{lin}_i$, $\mathbf{ray}_i$, $\mathbf{vertex}_i$ are vectors representing the linealities, rays and vertices and Υ_i, Ψ_i, Ω_i are the coefficients. n, m, k represent the maximal numbers of linealities, rays and vertices in this sum. Particularly, Υ_i can be any real number indicating the linear combination of linealities, whereas $\Psi_i \geq 0$ represent the conic combination of rays and $\sum_{i=1}^{k} \Omega_i = 1$, the convex combination of vertices.

In CoPE_FBA, the geometry of the optimal solution space is shaped by the subnetworks which are defined in terms of futile cycles with a fixed input-output relationship between substrates and products. These subnetworks are composed by reactions that carry flux variabilities. Vertices of the network come from the combination of alternative flux routes in these subnetworks.

The CoPE_FBA pipeline is a collection of Python scripts to enumerate the set of vertices, rays, and linealities of the optimal solution space of an FBA problem. This software was developed by the authors of (Kelk et al., 2012; Maarleveld et al., 2015) and can be downloaded from `http://memesa-tools.sf.net`.

3.2.3 Decomposing vertices into EFMs

Each flux distribution can be decomposed into a set of EFMs, the minimal operational units satisfying the stoichiometric and thermodynamic constraints in FBA (Schuster and Hilgetag, 1994; Schuster et al., 2002). The set of EFMs of a metabolic network can be calculated by EFMtool (Terzer and Stelling, 2008). Here we use a simple mixed-integer linear programming (MILP) approach to find all minimal conic combinations of EFMs corresponding to a given vertex v obtained during rFBA. The MILP can be expressed mathematically:

$$\min_{\boldsymbol{\lambda},\mathbf{a}} : \quad \sum \mathbf{a}_i$$

$$s.t. \quad \sum_{i=1}^{u} \boldsymbol{\lambda}_i \cdot \mathbf{e}^i = v, \tag{3.3}$$

$$0 \leq \boldsymbol{\lambda}_i \leq bigM \cdot \mathbf{a}_i, \qquad \mathbf{a}_i \in \{0,1\} \tag{3.4}$$

where $\mathbf{e}^1, \ldots, \mathbf{e}^u$ are the EFMs in the metabolic network and $bigM$ is a sufficiently large constant giving an upper bound for the coefficient $\boldsymbol{\lambda}_i$ ("big M constraint"). The binary variables $\mathbf{a}_i$ indicate whether the EFM $\mathbf{e}^i$ is used in the decomposition of v or not. By solving this optimization problem with the objective of minimizing the sum of binary variables, we obtain a minimal set of EFMs in the decomposition.

In order to explore all the alternative solutions in this MILP, we add constraints to calculate all the solutions ("no-good cuts"). Assume that one solution of the above MILP is $(\mathbf{a}^*, \boldsymbol{\lambda}^*)$. Then we require:

$$\sum \mathbf{a}_i = \sum \mathbf{a}_i^*, \tag{3.5}$$

$$\sum \mathbf{a}_i^* \cdot (1 - \mathbf{a}_i) + \sum (1 - \mathbf{a}_i^*) \cdot \mathbf{a}_i \geq 1. \tag{3.6}$$

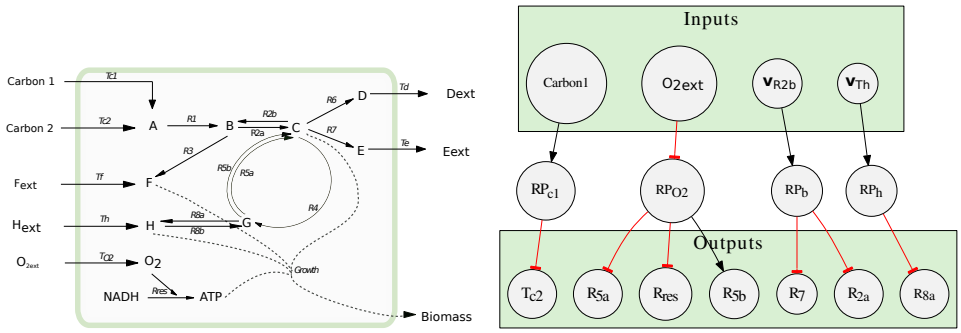
Eq. 3.5 constrains the new solution to have the same objective function value as the original solution. Eq. 3.6 is added to exclude the solutions already found. These constraints are added iteratively until there are no more feasible solutions.

3.2.4 Analytic pipeline

The analytic pipeline that we constructed has been implemented in Matlab. For rFBA we used the Cobra Toolbox (`https://opencobra.github.io/cobratoolbox`). Flux variability can be calculated by adding FVA to rFBA in each time step. To compute the vertices of the optimal flux space we use the CoPE_FBA pipeline (see above). The EFMs of the metabolic network are calculated by EFMtool (Terzer and Stelling, 2008). MILPs are solved by the MILP solver Gurobi (`http://www.gurobi.com`).

3.3 Case study on a core carbon network

We first evaluated our analytic pipeline on a small metabolic network proposed in (Covert et al., 2001), which mimics core carbon metabolism including glycolysis, citric acid cycle (TCA), carbon storage pathway, amino acid synthesis, pentose phosphate pathway (PPP) and fermentation pathway, as shown in Figure 3.2. The detailed reactions and regulatory rules are given in Table 3.1.



(a) Core carbon metabolic network. (b) Regulatory rules

Figure 3.2: Core carbon network and the associated regulatory rules.

3.3.1 Network description

The metabolic network contains 20 reactions, including 7 exchange reactions as shown in Figure 3.2a. The pathway from A to C is regarded as glycolysis, which includes reactions R_1, R_{2a}, R_{2b}. The cycle between C and G is considered as TCA cycle. R_{5a} is the aerobic pathway and R_{5b} is the anaerobic pathway. R_3 connecting B and F denotes PPP. H represents the amino acid. Reactions R_{8a} and R_{8b} link the TCA cycle with the amino acid uptake pathway. D and E represent the secretions. This core carbon network does not consider enzyme production. Instead, the macromolecule production reactions are replaced by a total biomass reaction *Growth*. For the details of the metabolic reactions see Table 3.1.

Besides the metabolism above, four regulatory proteins were added (Covert et al., 2001), which perform the regulatory control of the activities of the metabolic reactions. As shown in Table 3.1, RP_{c1} is the regulatory protein in the *Carbon* 1 – *Carbon* 2 diauxic shift, which is inhibited by *Carbon* 1 and then inactivates the transport of *Carbon* 2. This ensures that the cell cannot uptake *Carbon* 2 if *Carbon* 1 is in the culture.

RP_{O2} is responsible for the switch between aerobic growth and fermentation. The expression state of RP_{O2} is inhibited by the external oxygen O_{2ext}. The aerobic pathway indicated by R_{5a} is further activated by the absence of RP_{O2}, and the anaerobic pathway represented by R_{5b} is inhibited. The metabolite H in this core carbon network represents the amino acid. The reactions R_{8a}, R_{8b} connect the TCA

Reaction	Name	Regulation
Exchange reactions		
$Carbon\ 1 \rightarrow A$	T_{c1}	
$Carbon\ 2 \rightarrow A$	T_{c2}	$Not\ RP_{c1}$
$F_{ext} \rightarrow F$	T_f	
$H_{ext} \rightarrow H$	T_h	
$O_{2ext} \rightarrow O_2$	T_{O2}	
$D \rightarrow D_{ext}$	T_d	
$E \rightarrow E_{ext}$	T_e	
Metabolic reactions		
$A + ATP \rightarrow B$	R_1	
$B \rightarrow C + 2ATP + 2NADH$	R_{2a}	$Not\ RP_b$
$C + 2ATP + 2NADH \rightarrow B$	R_{2b}	
$C \rightarrow 2ATP + 3D$	R_6	
$C + 4NADH \rightarrow 3E$	R_7	$Not\ RP_b$
$B \rightarrow F$	R_3	
$C \rightarrow G$	R_4	
$G + ATP + 2NADH \rightarrow H$	R_{8a}	$Not\ RP_h$
$H \rightarrow G + ATP + 2NADH$	R_{8b}	
$G \rightarrow 0.8C + 2NADH$	R_{5a}	$Not\ RP_{O2}$
$G \rightarrow 0.8C + 2NADH$	R_{5b}	RP_{O2}
$O_2 + NADH \rightarrow ATP$	$Rres$	$Not\ RP_{O2}$
Biomass reaction		
$C + F + H + 10ATP \rightarrow Biomass$	$Growth$	
Regulatory proteins	RP_{O2}	$Not\ O_{2ext}$
	RP_{c1}	$Carbon\ 1$
	RP_h	$\mathbf{v}_{Th} > 0$
	RP_b	$\mathbf{v}_{R2b} > 0$

Table 3.1: Metabolic reactions and regulatory rules of the core carbon network

cycle and the usage of extracellular amino acid. Actually, this is one reversible reaction whose direction is regulated by the transport process T_h via the regulatory protein RP_h. The regulatory protein RP_h is activated when the cells are able to uptake external amino acid, which is indicated by $\mathbf{v}_{Th} > 0$. The activity of reaction R_{8a} is then repressed by the absence of RP_h so that the cells are unable to synthesize amino acid from the TCA cycle. Hence, RP_h is used to balance the assimilation of the carbon source and the amino acid. The regulatory rules ensure that the cells do not synthesize the amino acid from the carbon pathway when it can be taken up from the external environment (Umbarger, 1978).

The last regulation is to balance the key intermediate metabolites in the cells, which are denoted by B and C in this network. By regulating the level of B, the cell can balance the different metabolic pathways like glycolysis, TCA cycle, Pentose phosphate pathway etc. Since the concentration of the internal metabolite B is not available in rFBA, the flux through reaction R_{2b} is used as the signal to determine the value of the regulatory protein RP_b (Covert et al., 2001). The positive flux

of reaction R_{2b} is used as a signal that there is too much C in the cell. Whenever $\mathbf{v}_{R2b} > 0$, the regulatory protein RP_b is activated to inhibit the activity of R_{2a}. Thus, C cannot be produced from B anymore if R_{2a} is repressed.

3.3.2 Results

Time profiles of substrates and secretory products

We consider the growth on two different carbon sources, *Carbon* 1 and *Carbon* 2, which are both set to 10 mM initially. The oxygen concentration O_{2ext} is set to 1000 mM to make it available in excess, whereas H_{ext} and F_{ext} are set to 0. The time step is set to 0.25 hour, which is assumed to be the minimum time for protein synthesis/degradation. The bounds for the exchange reactions are chosen according to (Covert et al., 2001). Figure 3.3 shows that growth is interrupted by a lag phase. Only after the preferred carbon source *Carbon* 1 has been completely exhausted, followed by the lag phase, the cell can restart to grow on the non-preferred carbon source *Carbon* 2 (Stülke and Hillen, 1999). In rFBA, the regulatory protein RP_{c1} plays an essential role in performing the carbon switch during the lag phase.

But why is there a pause after *Carbon* 1 is depleted? How long should it be?

In rFBA, the lag phase is assumed to arise from the time delay between gene states. It requires a time period to transmit the signal to regulatory proteins and then finally, to regulate metabolism. In this case, the key regulatory pathways are managed by the regulatory protein RP_{c1}. The regulatory process is that the expression state of RP_{c1} is initially activated since *Carbon* 1 is in the media. Consequently, T_{c2} is repressed because of the presence of RP_{c1} in growth phase (a). When *Carbon* 1 suddenly disappears, the expression state of RP_{c1} is inhibited immediately. However, at least one time step is required for RP_{c1} to totally disappear either by protein degradation or dilution. Another time step is needed in order to synthesize the transporter of uptake reaction T_{c2} after the expression state of its encoding gene has been activated. Such a delay is represented by the lag phase that includes two time steps, which is the growth phase (b) in Figure 3.3. This ensures that reaction T_{c2} can be activated after the lag phase to continue the growth on *Carbon* 2. Therefore, there are two time steps in the lag phase in this case, starting from the absence of *Carbon* 1 to the activation of reaction T_{c2}.

Flux variability analysis during carbon shift

In order to explore the alternative optima, we use FVA to get insight into the robustness of the reaction fluxes reaching the same optimal objective value, see Figure 3.4. We can observe the robustness of the reaction fluxes during the dynamic growth. Only reactions R_{8a}, R_{8b} exhibit an obvious variability. The fluxes of R_7, T_e are close to zero, and can be neglected. In fact, reactions R_{8a}, R_{8b} make up an irreversible cycle, which indicates a ray in the metabolic network when their fluxes are not bounded. Even though their fluxes can be $+\infty$ if there are no upper bounds, the

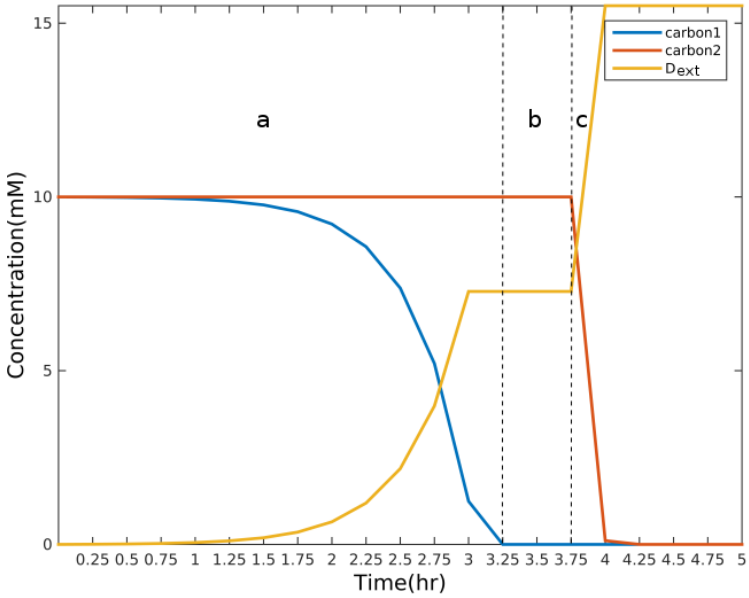


Figure 3.3: Predicted concentrations of the external metabolites during carbon shift. Shown are the substrates *Carbon* 1 and *Carbon* 2 that cells take up and the by-product D_{ext} secreted during the dynamic growth.

pure flux from metabolite G to H must be 2.41 mmol/gDW*hr in this case. This is because H_{ext} is not there and H has to be converted from G to support the growth.

Characterization of the polyhedron in dynamic growth

For the FBA solution of the core carbon network without regulation, 12 possible vertices are found by CoPE_FBA. Consistent with the FVA result that reaction fluxes show little variability in the diauxic shift, the number of vertices in the rFBA simulation is always 1. However, the vertex is changing depending on the external substrate concentrations. Figure 3.5 shows the switch. In growth phase (a), since *Carbon* 1 is sufficient, the fluxes stay constant and vertex (A) is the vertex during this first growth phase. Vertex (B) is the vertex of the growth phase (c) on *Carbon* 2 (see Figure 3.5). The white squares of the bars below the vertex represent the reactions that are switched off according to the regulatory rules. In the time interval 0-3.25hr, for example, T_{c2} and R_{5b} are repressed by *Carbon* 1 and O_{2ext} respectively.

Decomposing the vertices to elementary flux modes

Using the mixed-integer program described in Section 3.2.3, the two vertices (A) and (B) are decomposed into EFMs, which allows us to understand the shift of EFMs during the growth phase. The results are listed in Table 3.2. The complete

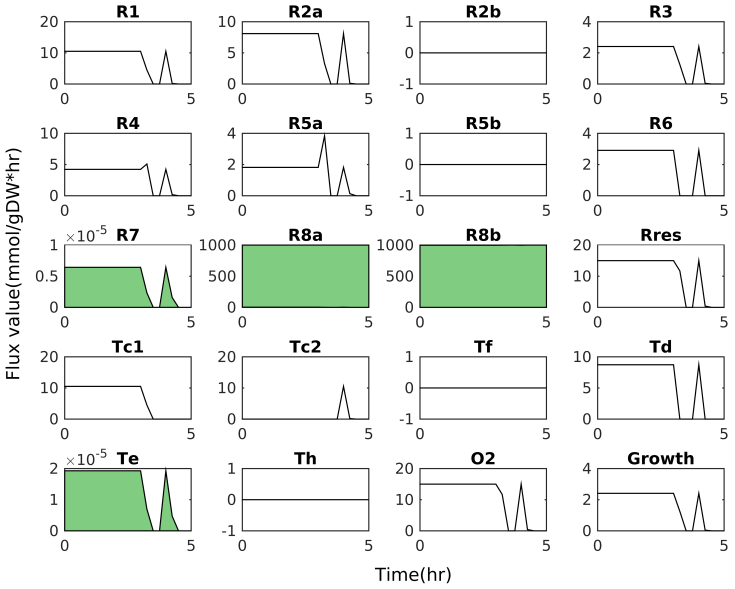


Figure 3.4: Robustness of reaction fluxes during carbon shift. The X axis represents time, the Y axis the reaction fluxes. Each subfigure shows in green the flux variability of one reaction.

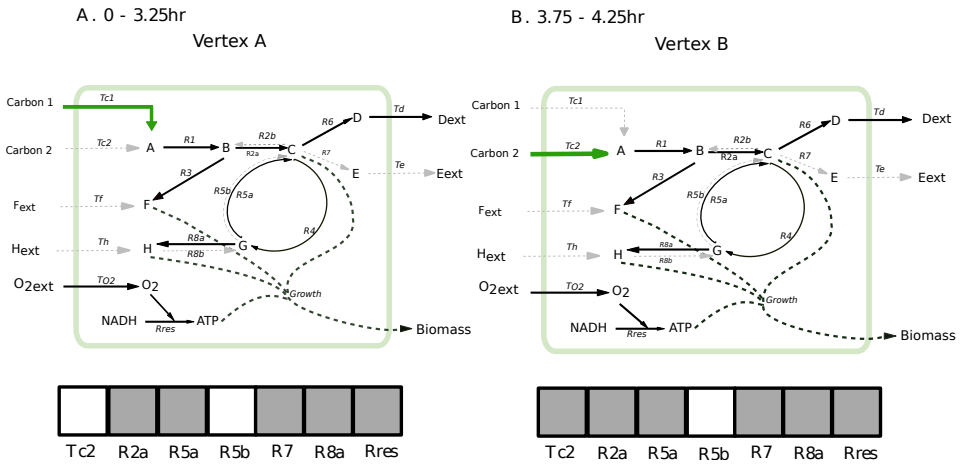


Figure 3.5: Shift of vertices during the carbon switch. Grey dotted resp. black lines indicate the inactive resp. active reactions, green lines show the differences.

set of 81 EFMs for the core carbon network is given in (Covert and Palsson, 2003). We use the same names for the EFMs in Table 3.2 and the details are shown in Figure 3.6 .

EFMs	efm 30	efm 34	efm 58	efm 62
Vertex A (0-3.25hr)	0.59	1.82	0	0
Vertex B (3.75-4.25hr)	0	0	0.59	1.82

Table 3.2: Decomposition of vertices during the diauxic shift. The numbers are the weights of EFMs in the conic combination representing the corresponding vertex.

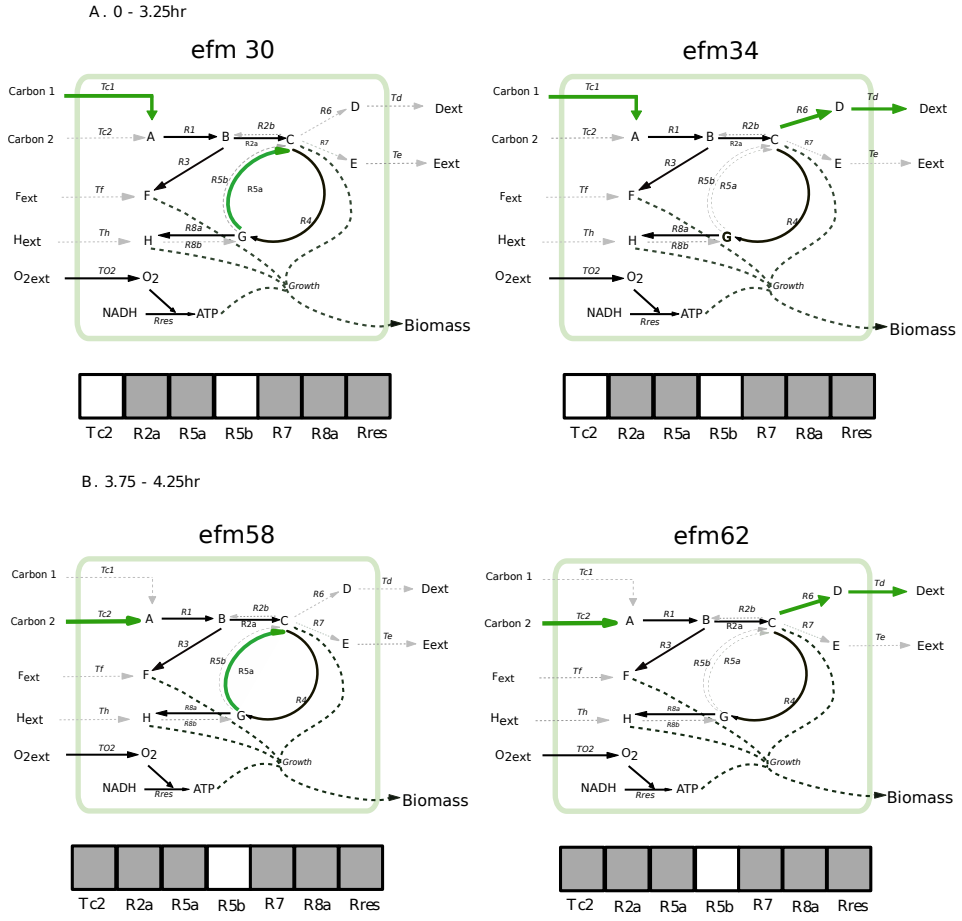


Figure 3.6: Comparison of the EFMs composing the vertex (A) and vertex (B). During the carbon switch, the EFMs shift from efm 30, efm 34 to efm 58, efm 62. Grey dotted resp. black lines indicate the inactive resp. active reactions, green lines show the differences.

In the first exponential growth phase of the diauxic shift, the cells use efm 30 and efm 34 (see Figure 3.6) to achieve optimal growth, along with the weights 0.59 and 1.82. Comparing these two EFMs, we see that they share several active reactions, such as T_{c1}, R_{2a}, T_{o2}, R_4 and $Growth$. However, efm 30 has an active reaction R_{5a}, whereas efm 34 does not. Instead, efm 34 uses R_6 and T_d to build the vertex (A) of

the optimal solution space. For the second growth phase, efm 58 and efm 62 build the vertex (B) with the corresponding weights 0.59 and 1.82. From Figure 3.6, we can see that efm 58 is similar to efm 30 except that T_{c1} is replaced by T_{c2}, since *Carbon* 1 is exhausted in the second growth phase. The same holds for efm 34 and efm 62. Therefore, when *Carbon* 1 is depleted, the EFMs efm 30, efm 34 for vertex (A) shift to the EFMs efm 58, efm 62 for vertex (B).

3.3.3 Conclusions

Analyzing the optimal solution space and looking at non-decomposable EFMs during rFBA enable us to better explore all the alternative pathways. In this section, we proposed an analytic pipeline to evaluate the polyhedron of optimal solutions in rFBA. We then applied this pipeline to a core carbon network in a first case study. We used FVA to study the robustness of the optimal fluxes, calculated the vertices of the optimal solution space, and decomposed each of them into a minimal set of EFMs. At the end, we obtained a combination of EFMs shifting with the change in the external environment. The network used here is very small and only a few alternative pathways could be observed. To observe more variability in the optimal solutions, we next apply our pipeline to a larger metabolic network: the central metabolic network of *E. coli*.

3.4 Case study on the central metabolic network of *E. coli*

3.4.1 Network description

The central metabolic network of *E. coli* was constructed by condensing the genome-scale metabolic network iAF1260 (Feist et al., 2007; Orth, 2010), which has been shown in Figure 1.1. In contrast to the core carbon metabolic network illustrated in Figure 3.2a, the central metabolic model of *E. coli* contains more details of the central metabolism, including glycolysis, TCA cycle, PPP, nitrogen pathway, and by-product secretions. It has 95 reactions, 72 metabolites, and 137 genes, of which 56 genes are regulated (Orth, 2010). There are 11 external substrates and 12 internal reactions involved as the extracellular and intracellular signals in gene regulation. 16 genes encoding regulatory proteins are directly affected by these signals, which are then used to regulate the other genes of regulatory proteins or to directly control the metabolic reactions.

In this case study, we consider the aerobic growth on glucose, which is set to 8 mM initially. The other carbon sources are all set to 0, except that initial acetate is given 0.3 mM. The oxygen concentration is set to 1000 mM to make it available in excess.

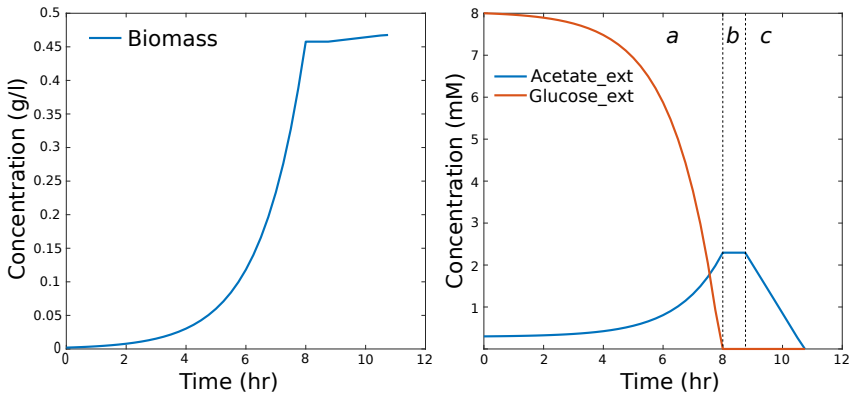


Figure 3.7: Predicted dynamic concentrations of biomass, glucose and acetata. Growth phase is divided into three phases (a, b and c). Glucose is the only carbon source in phase (a). Acetate is first secreted during phase (a) and then is assimilated in phase (c) after a short pause (b).

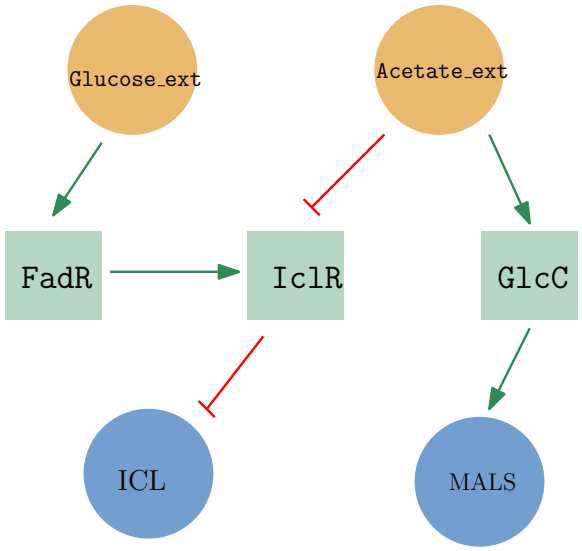


Figure 3.8: Regulatory rules involved in glucose-acetate shift. `Glucose_ext` and `Acetate_ext` denote the external glucose and acetate respectively. `FadR`, `IclR`, and `GlcC` are regulatory proteins. ICL and MALS are the reactions of TCA cycle shown in Figure 1.1.

3.4.2 Results

Dynamics of biomass and external substrates

First, the dynamics of the total biomass and external substrates resulting from rFBA has been investigated. We can see in Figure 3.7 that the model is taking up external glucose `Glucose_ext` to produce biomass during growth phase (a). External acetate `Acetate_ext` is produced in this growth phase. After the external glucose is depleted, there is a lag phase (b). The interesting observation is that external acetate `Acetate_ext` is then used as the carbon source after the pause (b), which

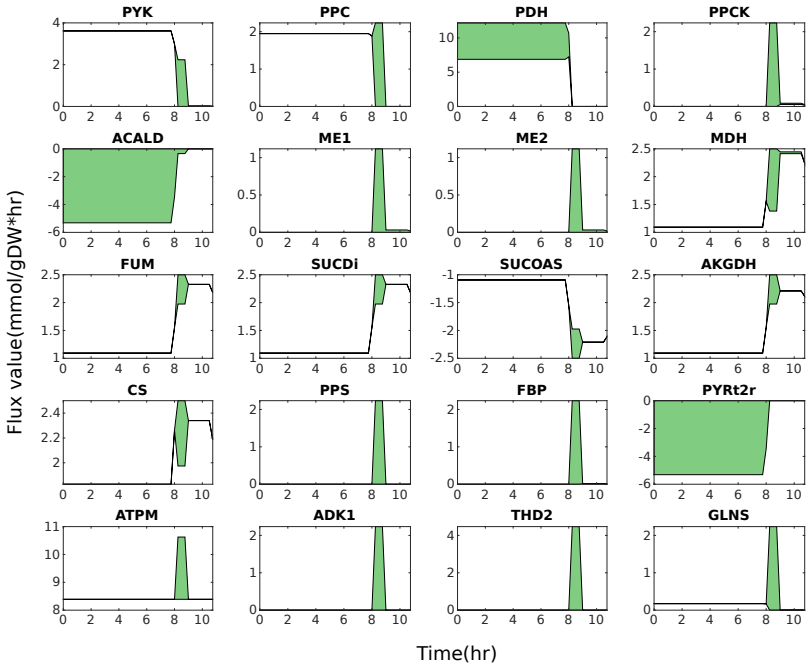


Figure 3.9: Robustness of reaction fluxes during glucose-acetate switch. The X axis represents time, the Y axis the reaction fluxes. Each subfigure shows in green the flux variability of one reaction. Above each subfigure is the corresponding reaction name.

is the growth phase (c) in Figure 3.7. The growth stops when `Acetate_ext` is also exhausted. However, since the acetate produced during the growth phase (a) is limited, the biomass does not increase significantly in phase (c).

Figure 3.8 presents the regulatory rules that are important in this case, which provide an explanation for the lag phase during the glucose-acetate switch. It is clearly observed from this figure that reaction ICL is inhibited by external glucose `Glucose_ext` through the regulatory proteins `FadR` and `IclR`. Therefore, when there is glucose in the environment, the usage of acetate is repressed. This is because the reactions ICL and MALS are required for growing on acetate. When the glucose is exhausted, the expression of `FadR` and `IclR` is inhibited, and reaction ICL is activated. Thus, acetate can be used for supporting growth. In rFBA, three time steps are needed to activate reaction ICL, since the gene regulatory network state is updated stepwise. The three steps are reflected by the lag phase (b). Biologically, the regulatory proteins `FadR` and `IclR` require a time period to vanish after `Glucose_ext` is exhausted. The relevant enzyme must be produced before reaction ICL can be activated. Yet, only the time delay is considered by time steps in rFBA while the protein expression is ignored.

Flux variabilities

In contrast to the case study on the core carbon network, the central metabolic network of *E. coli* shows a larger robustness of reaction fluxes. Figure 3.9 illustrates the variability of some significant reaction fluxes in the network. During growth phase (a), the acetaldehyde secretion pathway including reactions ACALD, PDH, and the pyruvate production reaction PYRt2r contain variabilities. During growth phase (b), there are variabilities in the TCA cycle including reactions such as FUM, CS, MDH, AKGDH, etc. During growth phase (c), flux variability exists in reactions PYK, ME1, ME2, PPCK, which connect the glycolysis pathway and the TCA cycle.

Characterization of the optimal solution space

The number of vertices in the rFBA simulation of the *E. coli* central network is calculated by CoPE_FBA. During growth phases (a), (b), and (c), the optimal solution space has 2, 11, and 3 vertices respectively. In total, 161749 EFMs are computed by EFMTOOL in the central metabolic network of *E. coli*. Then, we individually decompose the calculated vertices into a minimal set of EFMs. The results are listed in Table 3.3.

Growth phase	a (0-8hr)	b (8-8.75hr)	c (8.75-10.75hr)
Vertices	2 (a_1,a_2)	11 (b_1-b_{11})	3 (c_1-c_3)
Number of decomposed EFMs	35	12	4
Number of decompositions	178	13	3

Table 3.3: Number of vertices, decompositions, and EFMs during each growth phase.

Decomposing the vertices into elementary flux modes

The two vertices during growth phase (a) are shown in Figure 3.10 and Figure 3.11. As mentioned in the results of FVA, only the pyruvate and acetaldehyde production pathways contain variabilities in growth phase (a). The two pathways are alternatives. The internal pyruvate in excess is either secreted to `Pyruvate_ext` or converted into external acetaldehyde `Acetaldehyde_ext`. In vertex (a_1), the internal pyruvate is secreted extracellularly at a large rate while the acetaldehyde production pathway is inactive. On the other hand, all is converted into `Acetaldehyde_ext` in vertex (a_2). Both vertices (a_1) and (a_2) are decomposed into 4 EFMs in each decomposition. There are 35 different EFMs in all the decompositions. 64 solutions have been found for decomposing vertex (a_1), and 114 solutions for decomposing vertex (a_2). Thus, the total number of decompositions during growth phase (a) is 178 (see Table 3.3). Particularly, EFM a12 (shown in Figure 3.12) occurs in every decomposition of both vertex (a_1) and vertex (a_2).

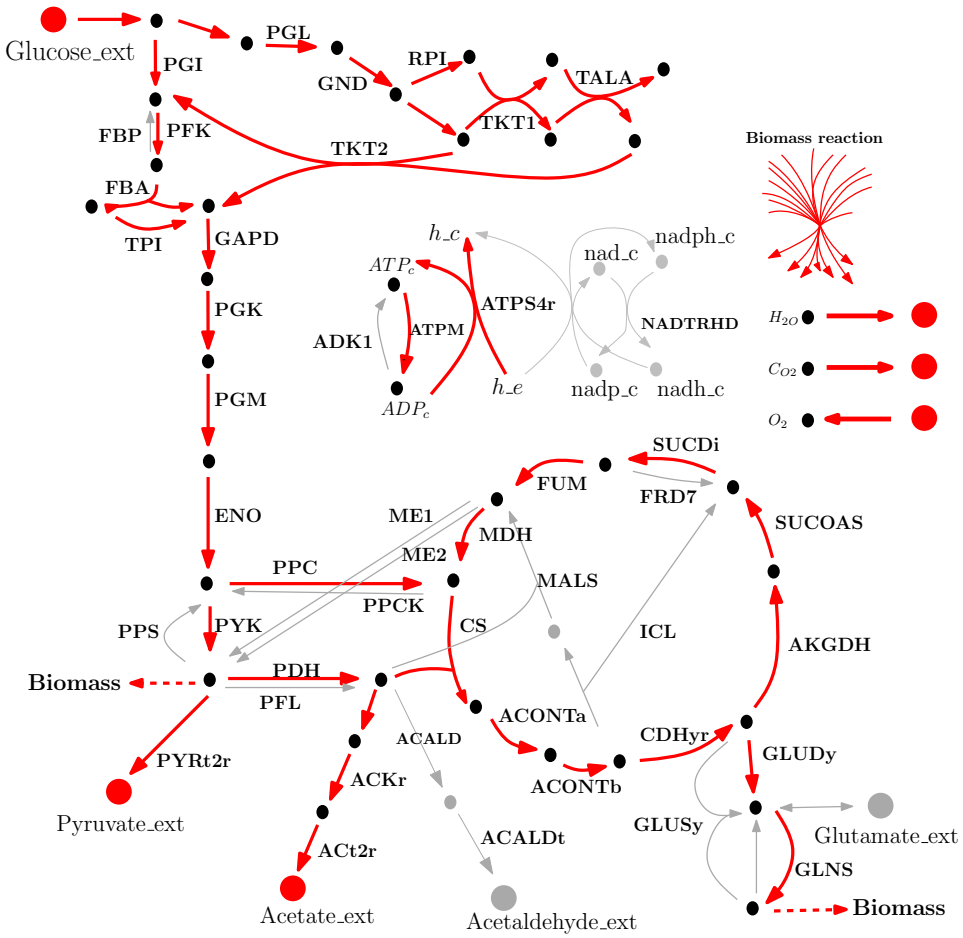


Figure 3.10: Vertex (a_1) of the optimal polyhedron during the growth phase (a) (0-8 hr). Grey lines indicate the inactive reactions, and red lines the active reactions.

Therefore, we may say that EFM a12 is the backbone EFM of the vertices during growth phase (a).

During growth phase (b), the cell does not grow and its biomass production is 0. It only assimilates external acetate `Acetate_ext` and oxygen to release CO_2, H_2O and ATP at the same time. This is because that the lower bound on ATP production is 8.39 in the *E. coli* central metabolic network (Orth, 2010). The solution space has a total of 11 vertices. 2 vertices have two decompositions and the other vertices only have one decomposition. So, the decomposition of these 11 vertices has a total of 13 solutions with 12 EFMs. There is also one EFM that exists in all the decompositions of the vertices in this growth phase. This EFM is named EFMb, as shown in Figure 3.16. Particularly, two vertices are only composed of 1 EFM, but with different coefficients, which is the backbone EFMb. In one vertex, the uptake rate of `Acetate_ext` is 2.5 while in another vertex, it is 1.97 to ensure a minimal ATP releasing rate of 8.39.

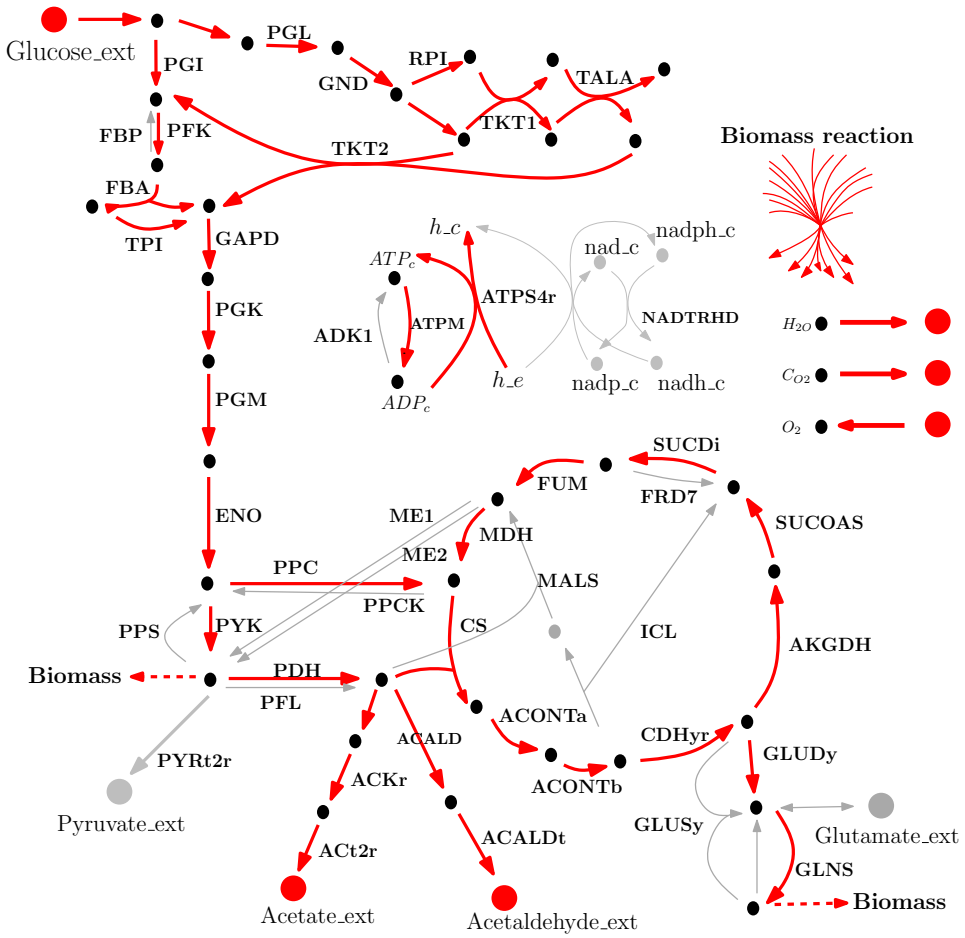


Figure 3.11: Vertex (a_2) of the optimal polyhedron during the growth phase (a) (0-8 hr). Grey lines indicate the inactive reactions, and red lines the active reactions. The difference between vertices (a_1) and (a_2) exists in the production pathways of external pyruvate `Pyruvate_ext` and external acetaldehyde `Acetaldehyde_ext`.

The 3 vertices of growth phase (c) are shown in Figure 3.13, Figure 3.14, and Figure 3.15. They each represent three possible pathways that break down `Acetate_ext` into energy and precursors supporting biomass production through the TCA cycle and glycolysis. Each vertex is decomposed into 2 EFMs with only one solution listed in Table 3.4. All the vertices in the growth phase (c) continue to use EFMb, which is the backbone EFM of all vertices during growth phase (b). In the decompositions of vertices (c_1), (c_2) and (c_3), EFMb must be combined with EFM c1, EFM c2, and EFM c3 respectively.

In this case study, the cell grows on glucose during phase (a) with an optimal polyhedron having two vertices: (a_1) and (a_2). They are composed of at least 4 EFMs with 178 possible combinations involving 35 EFMs. EFM a12 is essential among the 178 decompositions. When the external glucose is exhausted, the metabolic

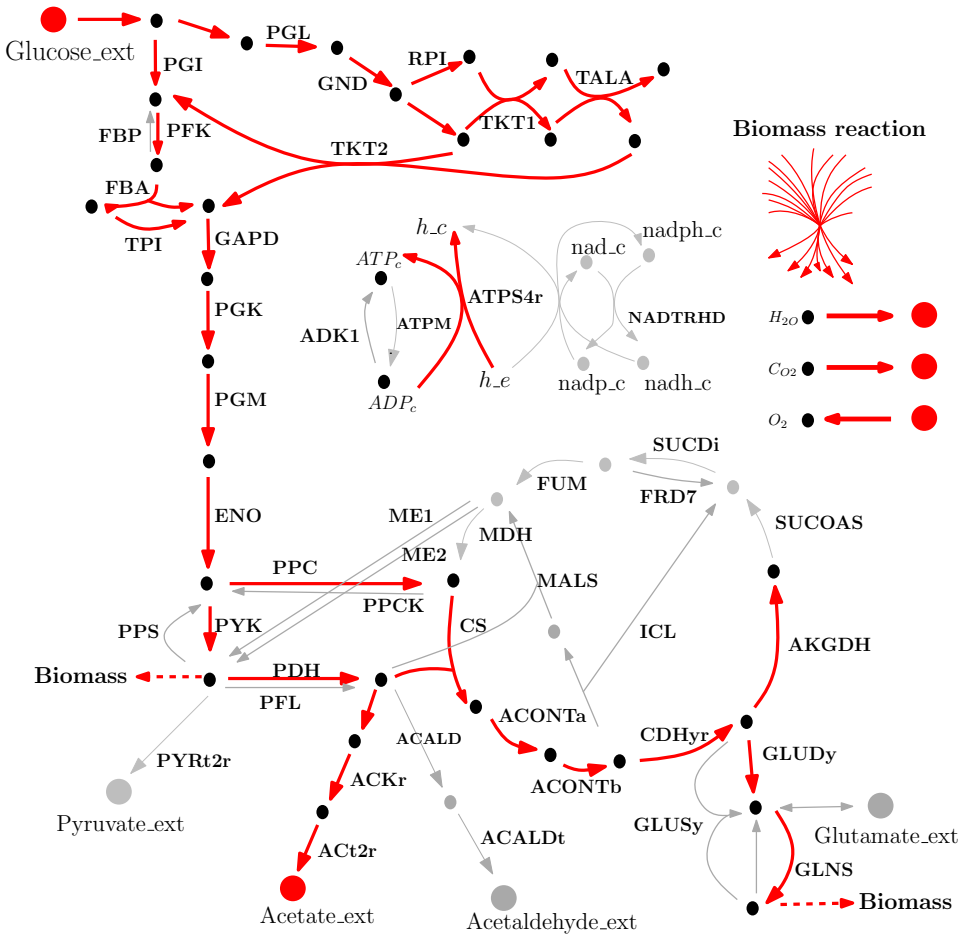


Figure 3.12: EFM a12 existing in all the decompositions of both vertex (a_1) and vertex (a_2). It is considered as the backbone EFM of vertices in growth phase (a). Grey lines indicate the inactive reactions, and red lines the active reactions.

Growth phase (c) (8.75-10.75hr)	Vertex (c_1)	Vertex (c_2)	Vertex (c_3)
EFMb	1.974	1.974	1.974
EFM c1	0.0114	0	0
EFM c2	0	0.0114	0
EFM c3	0	0	0.0114

Table 3.4: Decomposition of vertices (c_1), (c_2), and (c_3) during growth phase (c).

flux space shifts to a polyhedron with 11 vertices with the backbone EFMb in growth phase (b). In this solution space, biomass is not accumulated while ATP is produced at the minimal rate of 8.39. After the lag phase (b), metabolism switches to grow on acetate with an optimal polyhedron that has three vertices. An interest-

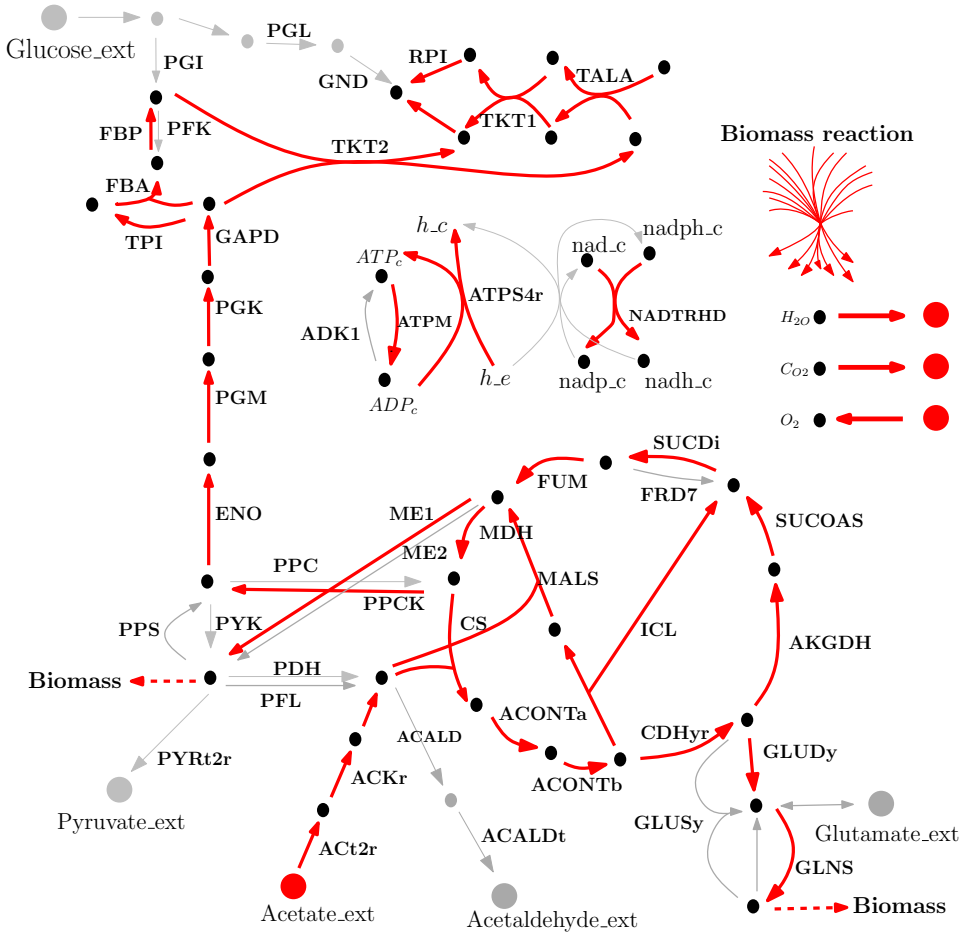


Figure 3.13: Vertex (c_1) of the optimal polyhedron during the growth phase (c).

ing observation is that the backbone EFMb still exists in all decompositions of the three vertices during phase (c). This means the backbone EFMb in growth phase (b) does not change during the shift from phase (b) to phase (c). In contrast, there is no overlap of EFMs between the decompositions during growth phases (a) and (b).

3.4.3 Conclusions

In this case study, our analytic pipeline has been applied to the central metabolic network of *E. coli*. The aerobic growth on glucose is simulated by rFBA. With gene regulatory rules, the model first grows on external glucose to produce acetate and then assimilates the secreted acetate after external glucose has been depleted. FVA is used to study the variability of the optimal fluxes, while the vertices of the optimal polyhedron are calculated by CoPE_FBA. Finally, the vertices are decomposed into EFMs.

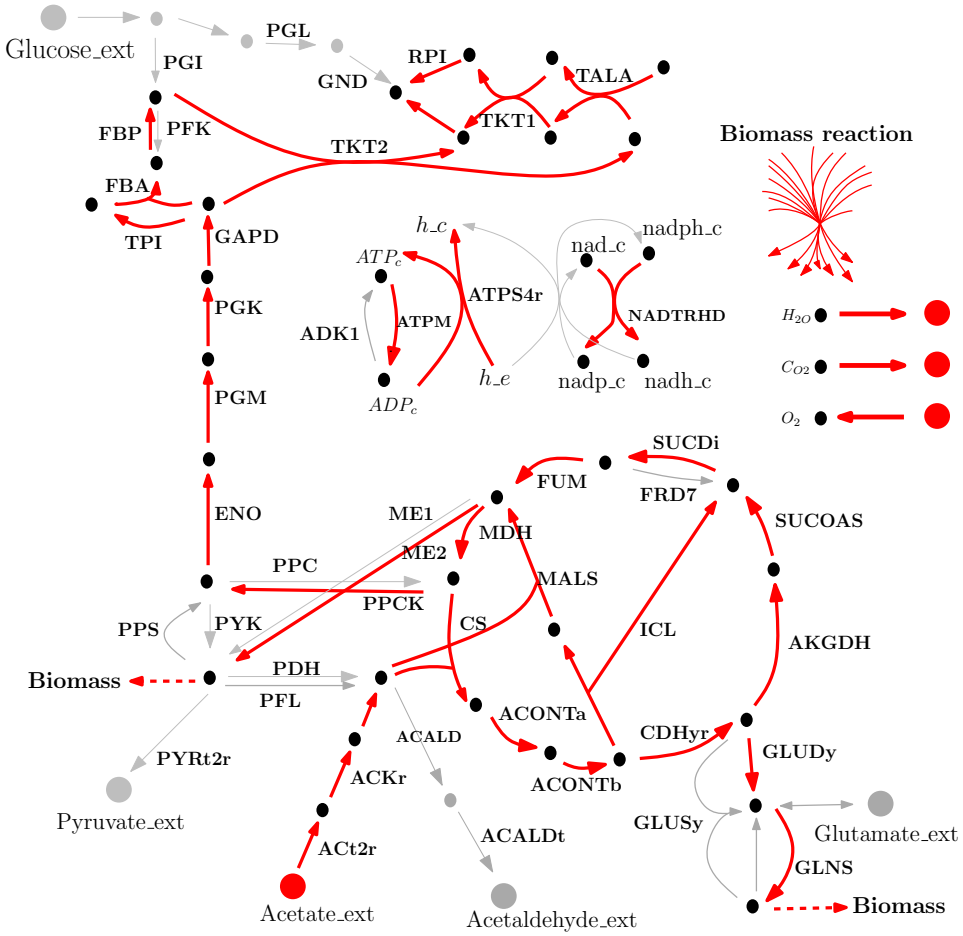


Figure 3.14: Vertex (c_2) of the optimal polyhedron during the growth phase (c).

Compared to the core carbon network in Section 3.3, we obtain a larger optimal solution space in this case study. All the three growth phases have one backbone EFM, which exists in the decomposition of each vertex, such as EFM a12 in growth phase (a). Growth phase (b) shares the backbone EFM EFMb with phase (c).

3.5 Discussion

For the lag phase, there are two time steps in the core carbon metabolic network. However, there are three time steps in the lag phase for the *E. coli* central metabolic network. Comparing the regulatory rules of the two models in Figure 3.2b and Figure 3.8, we can see that the core carbon metabolic network needs two time steps to activate the key reaction T_{c2} to grow on *Carbon* 2 when *Carbon* 1 is exhausted. However, the *E. coli* central metabolic model requires three time steps to activate reaction ICL, which is necessary for growing on external acetate. Thus, in rFBA, the length of the lag phase is determined by the gene regulatory network and the

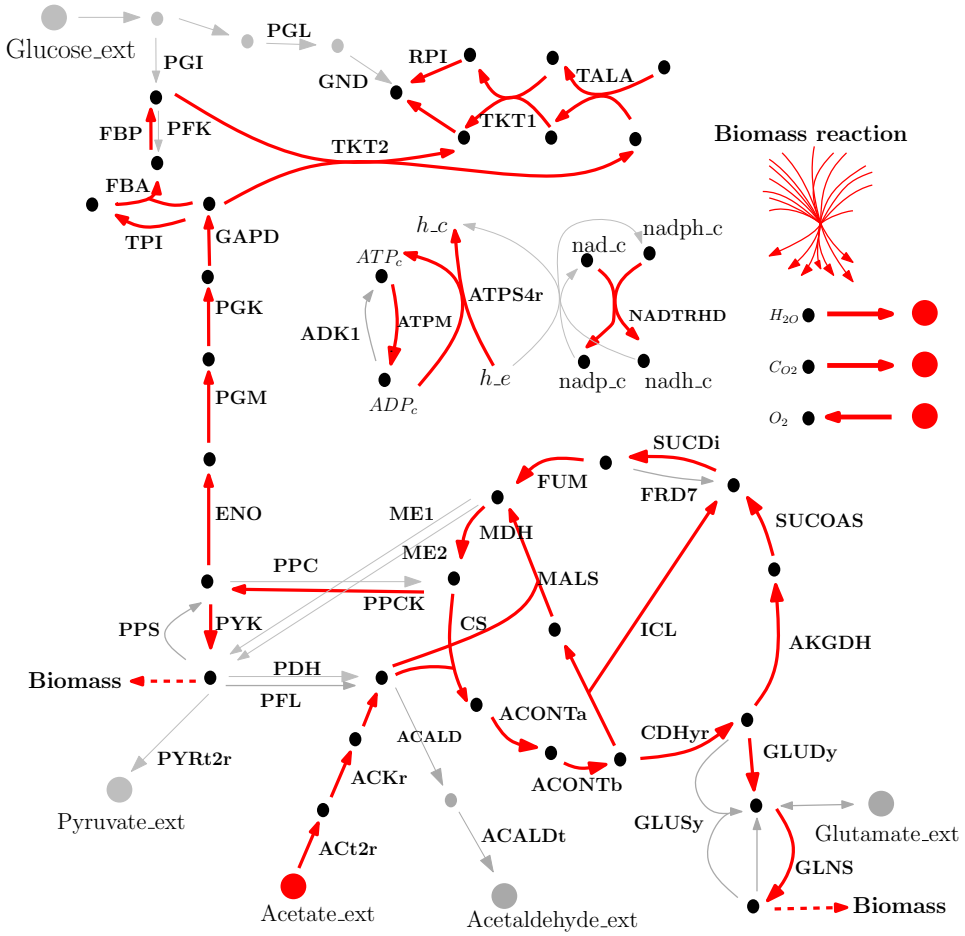


Figure 3.15: Vertex (c_3) of the optimal polyhedron during the growth phase (c). The three vertices are named in order. The difference between the three vertices (c_1), (c_2) and (c_3) exist in reactions: PYK, ME1, and ME2.

time interval. Biologically, it depends on the production or degradation/dilution of the proteins involved in the regulation. Hence, the time step for implementing rFBA should be given based on the rates of protein synthesis and degradation. In both case studies, the time interval is 0.25 hr, which is assumed to be the average protein synthesis/degradation rate at steady-state (Covert et al., 2001).

In conclusion, exploring the optimal solution space and looking for non-decomposable EFMs in rFBA enable us to comprehensively understand all the alternative pathways. Although the models implemented here are both small, we can still observe the changing optimal solution space with time and see how the EFMs switch during different growth phases. However, the flux variabilities of the reactions are limited in small metabolic models. We expect that by considering larger, possibly genome-scale metabolic reconstructions and multiple nutrients in the environment, we will observe much more variability in the optimal solutions,

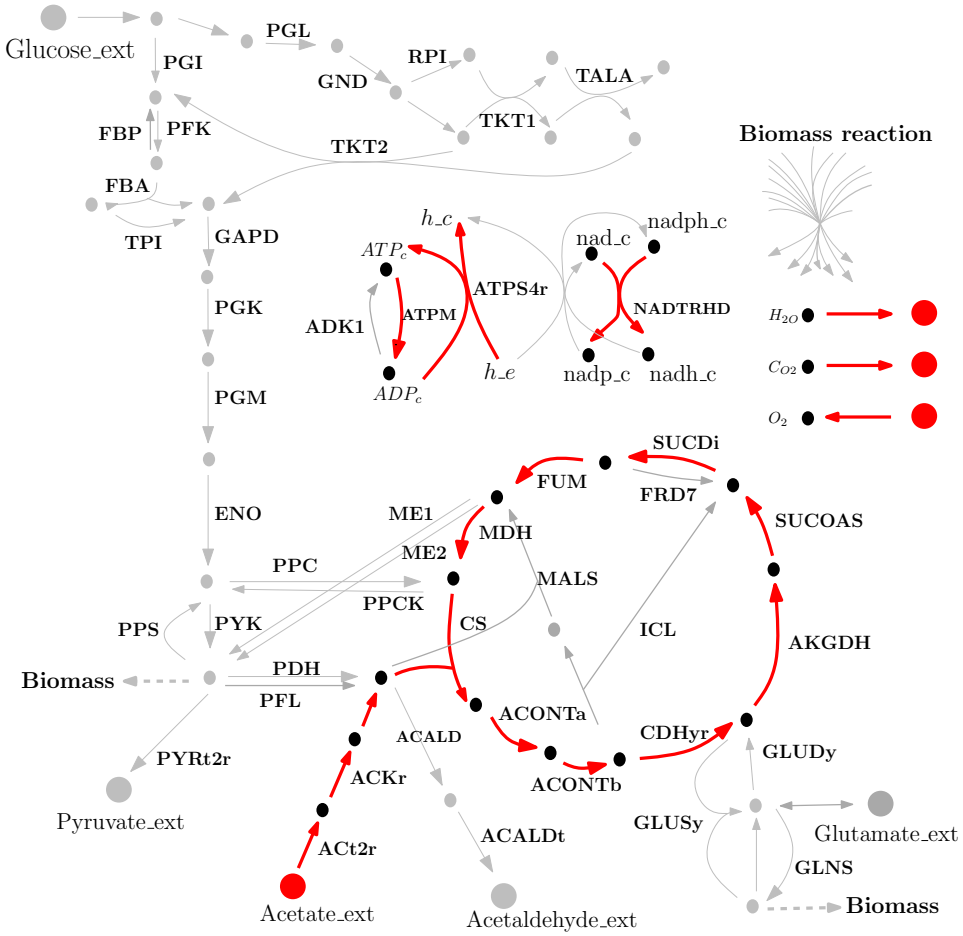


Figure 3.16: EFMb existing in each decomposition of vertices in growth phase (b) and phase (c).

which can be further explored by our approach.

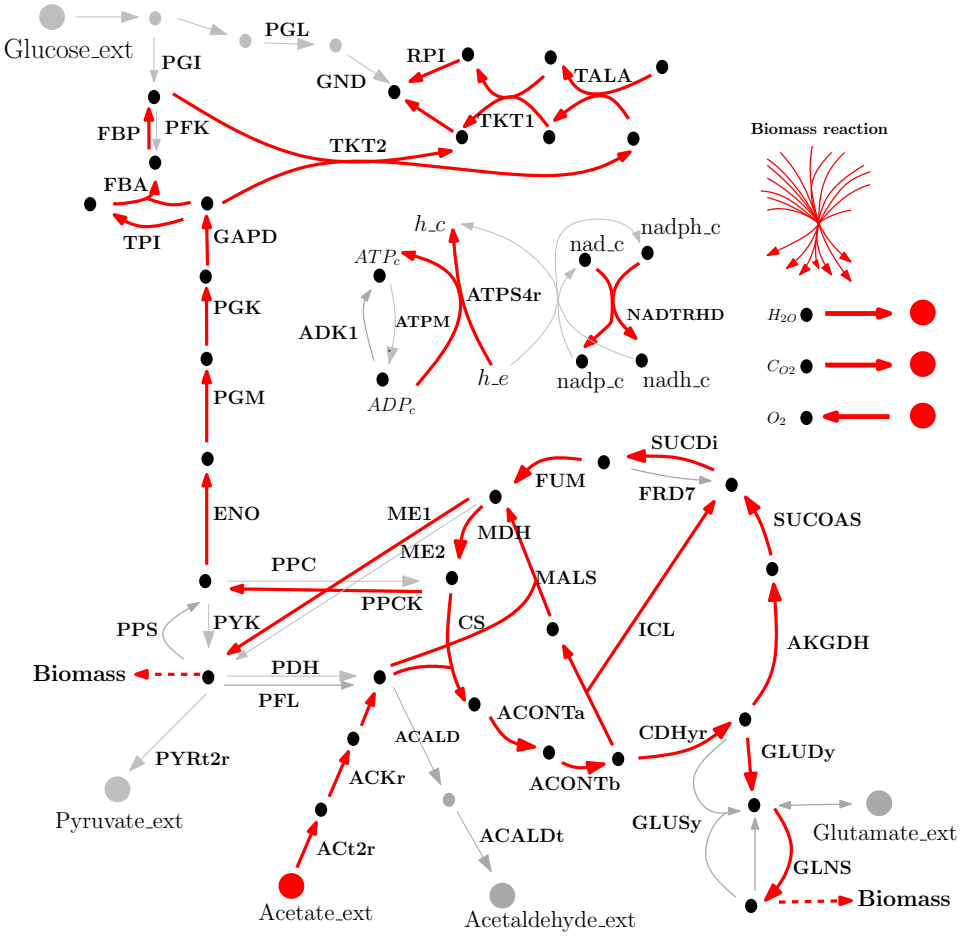


Figure 3.17: EFM c1 in the decomposition of vertex (c_1) besides EFMb.

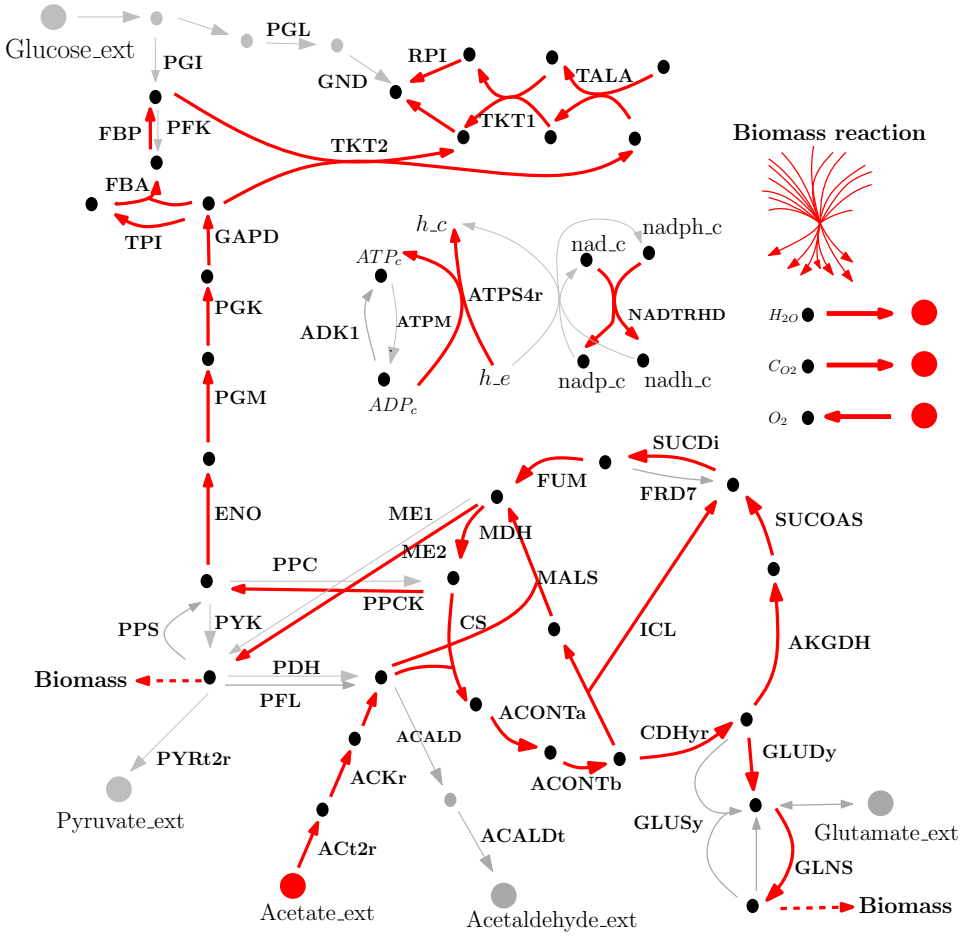


Figure 3.18: EFM c2 in the decomposition of vertex (c_2) besides EFMb.

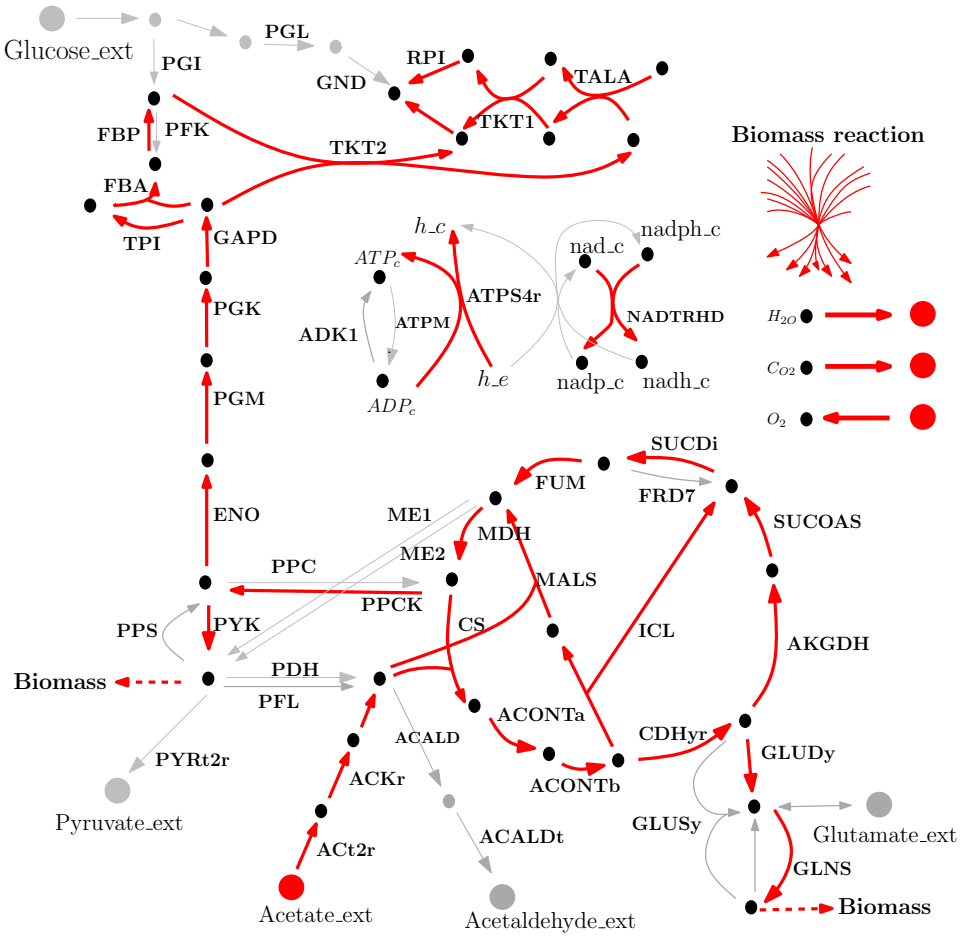


Figure 3.19: EFM c1 in the decomposition of vertex (c_3) besides EFMb. The difference between the three EFMs c1, c2 and c3 exists in reactions: PYK, ME1, and ME2.

Chapter 4

Iterating RBA incorporating regulatory rules

In this chapter, we present the early attempts of this PhD project to integrate resource allocation models with gene regulatory rules, while accounting for dynamic changes. First, we illustrate a strategy that iterates the resource allocation model to predict the dynamic fluxes and enzyme concentrations, which is called *iterative resource balance analysis* (iRBA). Then, *regulatory resource balance analysis* (riRBA) is proposed to additionally incorporate Boolean regulatory rules into iRBA.

4.1 Introduction

As introduced in Section 1.2.2, constraint-based modeling approaches have become a powerful tool in the analysis of genome-scale metabolic network reconstructions (Bordbar et al., 2014; Lewis et al., 2012). While standard FBA requires very few data, it is unable to capture more complex phenomena such as dynamics, resource allocation, or gene regulation. Extending the work by Palsson *et al.* (Varma and Palsson, 1994), Mahadevan *et al.* in 2002 introduced *dynamic flux balance analysis* (DFBA) (Mahadevan et al., 2002) to maximize biomass production over a time interval, taking into account the dynamics of extracellular metabolites and biomass. To incorporate the synthesis costs of macromolecules, Goelzer *et al.* developed *resource balance analysis* (RBA) (Goelzer et al., 2011, 2015), which allows predicting an optimal resource allocation for maximizing the steady-state growth rate. Lerman *et al.* introduced *ME-models* (Lerman et al., 2012; O'Brien et al., 2013), a related approach for integrating metabolism and gene expression at steady-state. To combine these two ways of extending FBA, dynamics and resource allocation, several frameworks have been developed during the last years,

which include *dynamic enzyme-cost FBA* (deFBA) (Waldherr et al., 2015), *conditional FBA* (cFBA) (Rügen et al., 2015; Reimers et al., 2017a), *dynamic resource balance analysis* (dRBA) (Jeanne et al., 2018) and *dynamicME* (Yang et al., 2019).

Concerning integrated modeling of metabolism and regulation, there exist approaches such as *regulatory flux balance analysis* (rFBA) (Covert et al., 2001) and *FlexFlux* (Marmiesse et al., 2015), which combine Boolean or multi-valued logical rules for the transcriptional regulation with a steady-state stoichiometric model of metabolism. Like the SOA variant of DFBA (Varma and Palsson, 1994; Mahadevan et al., 2002), these techniques iterate *flux balance analysis* by splitting the growth phase into discrete time steps. At each time step, the updated regulatory states are imposed as bounds on the reaction fluxes, while ignoring the costs for enzyme production. *Integrated FBA* (iFBA) (Covert et al., 2008) allows combing rFBA with a differential equation model for a specific subnetwork, while *integrated dynamic FBA* (idFBA) (Lee et al., 2008) brings together metabolism, regulation, and also signal transduction.

In addition to these iterative methods, there are also static approaches for combining metabolism and gene regulation. *Steady-state regulatory flux balance analysis* (SR-FBA) (Shlomi et al., 2007) aims at studying the steady-state behaviors of a metabolic-regulatory network by adding Boolean rules to the linear constraints of FBA, resulting in a mixed-integer linear program (Shlomi et al., 2007). *Probabilistic regulation of metabolism* (PROM) (Chandrasekaran and Price, 2010) makes use of microarray data sets to constrain the reaction upper bounds within a certain percentage of the maximal upper bound.

Approaches	No regulations		Regulations included	
	No enzyme costs	Enzyme costs included	No enzyme costs	Enzyme costs included
Static	FBA (1980s)	RBA (Goelzer and Fromion, 2011) ME models (Lerman et al., 2012)	SR-FBA (Shlomi et al., 2007) PROM (Chandrasekaran and Price, 2010)	
Iterative	DFBA (SOA)((Varma and Palsson, 1994))	dynamicME (Yang et al., 2019) iRBA (Chapter 4)	rFBA (Covert et al., 2001) iFBA (Covert et al., 2008) FlexFlux (Marmiesse et al., 2015)	idFBA(Lee et al., 2008) riRBA (Chapter 4)
Dynamic	DFBA (DOA)(Mahadevan et al., 2002)	deFBA (Waldherr et al., 2015) cFBA (Rügen et al., 2015) dRBA (Jeanne et al., 2018)		r-deFBA (Chapter 6)

Table 4.1: Constraint-based flux balance approaches.

To summarize, we classified in Table 4.1 existing flux balance approaches according to whether or not they include dynamics, macromolecule production costs, and gene regulation. As can be seen from Table 4.1, there has been no approach with the idea of iterating resource allocation models, such as RBA and ME-models, at the beginning of this PhD project. DynamicME applied the iterative idea to include the time-course of cellular metabolism and protein expression (Yang et al., 2019). It was developed based on the ME-models. Our iRBA has been done independently of dynamicME, which was published in January 2019. We already presented this work as a poster in the *17-th International Workshop on Bioinformatics and Systems Biology, in Jul. 2017 in Berlin, Germany*.

From the perspective of the three features in Table 4.1, the intuitive strategy of combining a resource allocation model, gene regulation, and dynamic growth consists in iterating RBA while additionally including gene regulatory rules. To implement this idea, *iterative resource balance analysis* (iRBA) is first introduced to predict the dynamics of metabolism, considering enzyme expression costs. Next, we add Boolean regulatory rules and propose *regulatory iterative resource balance analysis* (riRBA).

4.2 iRBA

Like in DFBA (SOA), see Section 2.1, the time course is discretized into n time intervals of length Δt in iRBA. At each time step, a resource allocation model is solved at the beginning to obtain the flux distribution and enzyme concentrations during this time step. In iRBA, we introduce RBA$(\mu)^k$ to represent the static optimization problem of the resource allocation at time step k for $k \in \{1, \ldots, n\}$. The biomass and external metabolite concentrations are iteratively updated based on the optimal growth rate and fluxes resulting from RBA$(\mu)^k$. Below, we formulate iRBA for the *metabolic-genetic network* model (see Figure 2.2) with the corresponding abbreviation and notations (see the details in Section 2.3.1).

4.2.1 Dynamics of biomass and external metabolites

We begin this section by shortly recalling the strategy used for the update of external metabolite and biomass concentrations, which was described in Section 2.1. Using the same strategy as DFBA, we also use the equations

$$\mathrm{Bio}^k = \mathrm{Bio}^{k-1} \cdot e^{\mu^k \cdot \Delta t}, \tag{2.3}$$

$$\mathbf{Y}^k = \mathbf{Y}^{k-1} - \frac{S_{\mathcal{Y},\mathcal{R}_\mathcal{Y}} \cdot \mathbf{v}^k_\mathcal{Y}}{\mu^k} \cdot \mathrm{Bio}^k \cdot (1 - e^{\mu^k \cdot \Delta t}), \tag{2.4}$$

to update the external metabolite and biomass concentrations in iRBA. μ^k is the optimal growth rate, and $\mathbf{v}^k_\mathcal{Y}$ is the set of exchange reaction fluxes at time step k resulting from RBA$(\mu)^k$, which is solved at the beginning of the time step k. $\mathbf{Y}^k$ represents the external metabolite concentrations at the end of time step k, and the nutrient concentrations $\mathbf{N}^k$ is a subvector of $\mathbf{Y}^k$. Bio^k represents the biomass concentration at the end of time step k. Notice that in Eq. 2.4, uptake fluxes are assumed to be negative, and exchange fluxes for secretions are assumed to be positive by convention. The initial concentrations of biomass and external species must be specified to initiate the iterative procedure.

4.2.2 Updating uptake fluxes of nutrients

Considering the fact that the metabolic fluxes are instantaneously affected by the changing external nutrient concentrations, we let the uptake fluxes of nutrients fol-

low Michaelis-Menten kientics

$$\mathbf{v}_i^k = -\frac{\mathbf{N}_i^{k-1} \cdot k_{cat}^{i,T}}{K_M^i + \mathbf{N}_i^{k-1}} \cdot T^k, \quad \text{for all } i \in \mathcal{R}_\mathcal{N},\ T \in \mathcal{T}, \tag{4.1}$$

where $k_{cat}^{i,T}$ is the catalytic rate of transporter T for uptake reaction i. K_M^i represents the Michaelis constant. $\mathbf{N}_i^{k-1}$ is the concentration of the nutrient at the end of time step $(k-1)$ and $\mathbf{v}_i^k$ is the uptake flux at time step k. T^k is the concentration of the transporter of the uptake reaction. Therefore, the uptake fluxes of nutrients are updated along with the dynamically changing nutrient concentrations.

4.2.3 Static non-linear optimization problem

As discussed above, iRBA iteratively solves RBA$(\mu)^k$ for $k \in \{1, \ldots, n\}$. It also assumes a steady-state, under which the metabolites maintain constant concentrations. Accordingly, RBA$(\mu)^k$ adopts the resource allocation formalism described in Section 2.3. Combining the resource allocation formalism with the iterative process, the static optimization at time step $k \in \{1, ..., n\}$ represented by RBA$(\mu)^k$ can be expressed as:

$$\begin{aligned}
\max_{\mathbf{v}^k, \mathbf{E}^k, \mu^k} \quad & \mu^k \\
s.t. \quad & S_{\mathcal{X},\mathcal{R}_\mathcal{Y}} \cdot \mathbf{v}_\mathcal{Y}^k + S_{\mathcal{X},\mathcal{R}_\mathcal{X}} \cdot \mathbf{v}_\mathcal{X}^k + S_{\mathcal{X},\mathcal{R}_\mathcal{P}} \cdot \mathbf{v}_\mathcal{P}^k = 0, \\
& S_{\mathcal{E},\mathcal{R}_\mathcal{E}} \cdot \mathbf{v}_\mathcal{E}^k - \mu^k \cdot \mathbf{E}^k = 0, \\
& S_{\mathcal{Q},\mathcal{R}_\mathcal{Q}} \cdot \mathbf{v}_\mathcal{Q}^k - \mu^k \cdot \mathbf{Q}^k = 0, \\
& \sum_{i \in cat(E)} \frac{|\mathbf{v}_i^k|}{k_{cat}^{i,E}} \leq E^k, \quad \text{for all } E \in \mathcal{E} \setminus \mathcal{T} \\
& \mathbf{v}_i^k = -\frac{\mathbf{N}_i^{k-1} \cdot k_{cat}^{i,T}}{K_M^i + \mathbf{N}_i^{k-1}} \cdot T^k, \quad \text{for all } i \in \mathcal{R}_\mathcal{N},\ T \in \mathcal{T} \\
& \mathbf{d}_\mathcal{P}^T \cdot \mathbf{P}^k \leq D, \\
& \mathbf{Q}^k = \mathbf{Q}_{fix}, \\
& \mathbf{lb} \leq \mathbf{v}^k \leq \mathbf{ub}, \\
& \mathbf{P}^k,\ \mu^k \geq 0.
\end{aligned} \tag{4.2}$$

RBA$(\mu)^k$ as presented in Eq. 4.2 is a non-linear optimization problem. This is because the term $\mu^k \cdot \mathbf{E}^k$ is not linear if we search for an optimal μ. To solve this problem, we use the bisection approach that is detailed in Algorithm 1, Section 2.3.4.

At each time step k, μ is initially fixed with a small value, and we get the relevant linear optimization program RBA$(\mu)^k$. Then, we keep doubling μ and solving RBA$(2\mu)^k$, until no feasible solution of RBA$(2\mu)^k$ can be found. The maximal

growth rate must be in the interval $[\mu, 2\mu]$. Next, we repeatedly bisect the interval and fix μ to the midpoint, and then check whether or not the linear program is feasible. The bisection process is continued until the interval is sufficiently small. In the end, we obtain the approximate maximal growth rate with a certain precision, see details in Section 2.3.4.

Therefore, our iRBA involves two computational loops. By converting the nonlinear optimization problem into a standard linear program, the inner loop indicates the bisection method detailed in Algorithm 1. At each time step, the maximal growth rate and the corresponding flux distribution are obtained from the inner loop computation. The outer loop iteratively solves the static optimization problem in Eq. 4.2 from the time step 1 to n. At each time step $k \in \{1, \ldots, n\}$ of the outer loop, the external metabolite concentrations $\mathbf{Y}^k$ and biomass Bio^k are updated by the optimal growth rate and the flux distribution. Thus, the dynamics of nutrients are accounted for in our iRBA, which simultaneously limits the uptake fluxes by Michaelis-Menten kinetics in Eq. 4.1.

4.2.4 Discussion

iRBA allows us to simulate the resource allocation with the consideration of the dynamics of external metabolites. Imagine a batch culture in a fixed bioactor. After the initial nutrients are given, the cells start to grow until all the nutrients are exhausted. At the same time, the biomass in the bioactor accumulates and the cell population may increase. In iRBA, $\text{RBA}(\mu)^k$ is solved to predict the flux distribution $\mathbf{v}^k$ and the maximal growth rate μ^k at time step k. The biomass and extracellular metabolite concentrations are updated with $\mathbf{v}^k$ and μ^k. The uptake fluxes are also updated at each time step according to the dynamically changing nutrients. Thus, $\text{RBA}(\mu)^k$ is iteratively solved and the concentrations of external species and biomass are correspondingly updated until the end of the culture.

Looking closer at the iRBA algorithm reveals that the only connection between the time steps is the dynamics of biomass and external metabolites calculated by Eq. 2.3 and Eq. 2.4. The protein abundances between time steps are independent. Their changes between time steps can be freely adjusted. Yet, the protein adjustment in a dynamic and changing environment should be constrained, even though the deterministic constraint may not be clear to us.

Hence, the plausibility of our iRBA is limited. It is worth mentioning that dynamicME solved this by introducing protein "inertia" constraints that account for the proteome dynamics (Yang et al., 2019). In dynamicME, they distinguish between two procedures, one that takes into consideration the protein dynamics between time steps and the other that does not. Their simulations, when discounting protein dynamics, only checks whether the nutrients are exhausted or newly available for constraining the uptake reaction fluxes. If not, they do not perform their ME-model. Otherwise the new ME-model is performed with the updated uptake reaction fluxes. Their second implementation removed the steady-state assumption in

macromolecule concentrations. Instead, the protein "inertia" constraints:

$$v_i - \mu \cdot p_i = \delta_i, \qquad \forall i \in Complex, \tag{4.3}$$

$$p_i = p_i^0 + \delta_i \cdot \Delta t, \qquad \forall i \in Complex \tag{4.4}$$

are added, where *Complex* is the set of protein complexes, p_i denotes a protein complex concentration, p_i^0 represents the protein complex concentration at the previous time step, μ is the growth rate, and v_i is its production rate (Yang et al., 2019). Thus, the accumulation or depletion indicated by δ_i is taken into account. If $\delta_i > 0$, the relevant protein complex is accumulated. Else if $\delta_i < 0$, it is depleted or diluted. δ_i is then used to update the protein abundance by Eq. 4.4. Regarding the proteome adjustment in response to the changing environment and perturbations, we also came up with the idea of minimizing the metabolic change between time steps. Nevertheless, we did not continue but instead, focused on the integration of the gene regulatory rules with iRBA in the following section.

4.3 riRBA

4.3.1 Regulatory constraints

Based on the iRBA and rFBA formalisms, we next introduce riRBA for embedding Boolean regulatory rules in the iterative procedure of iRBA. Hence, riRBA is also an iteratively dynamic approach.

As enzyme productions are included in the resource allocation models, we introduce a Boolean variable $\overline{p}^k \in \{\texttt{on}, \texttt{off}\}$ to represent the expression state of a gene encoding an enzyme p at time step k. This means that the `on`/`off` state of $\overline{p}^k$ indicates whether or not the gene encoding enzyme p can be expressed. If $\overline{p}^k = \texttt{on}$, the production rate of enzyme p is constrained by the original flux bound. If $\overline{p}^k = \texttt{off}$, the production rate is set to 0.

Similar to rFBA as formulated in Section 2.2, we also use a Boolean-valued function f_i that determines the expression activity of an enzyme p depending on the presence or absence of external metabolites and reaction fluxes at time step $(k-1)$. Therefore, for $k = 1, \ldots, n$, the regulatory constraints in riRBA can be formulated as:

$$\overline{p}^k = f_i(\mathbf{Y}^{k-1}, \mathbf{v}^{k-1}), \qquad p \in \mathcal{E} \tag{4.5}$$

$$\mathbf{lb}_p \cdot \overline{p}^k \leq \mathbf{v}_p^k \leq \mathbf{ub}_p \cdot \overline{p}^k, \qquad p \in \mathcal{E} \tag{4.6}$$

where $\overline{p}^k$ is a Boolean variable that can only be 1 corresponding to `on` or 0 resp. `off`. $\overline{p}^k$ represents the production state of an enzyme in $\mathcal{E}$ at time step k. $\mathbf{v}_p^k$ is the reaction flux that synthesizes enzyme p at time step k. Particularly, the expression state of protein $\overline{p}^k$ is updated according to the Boolean function in Eq. 4.5. Eq. 4.6 ensures that the reaction flux $\mathbf{v}_p^k$ can only be zero if $\overline{p}^k = 0$. However, it is bounded by the original upper and lower bounds $\mathbf{ub}_p, \mathbf{lb}_p$ when $\overline{p}^k = 1$.

4.3.2 Static non-linear optimization problem with regulatory constraints

Putting together the above regulatory constraints and the iRBA formalism, we obtain a non-linear optimization problem riRBA$(\mu)^k$ at time step $k \in \{1, \ldots, n\}$,

$$
\begin{aligned}
&\max_{\mathbf{v}^k, \mathbf{E}^k, \mu^k} \ \mu^k \\
&\quad s.t.\ S_{\mathcal{X},\mathcal{R}_\mathcal{Y}} \cdot \mathbf{v}^k_\mathcal{Y} + S_{\mathcal{X},\mathcal{R}_\mathcal{X}} \cdot \mathbf{v}^k_\mathcal{X} + S_{\mathcal{X},\mathcal{R}_\mathcal{P}} \cdot \mathbf{v}^k_\mathcal{P} = 0, \\
&\qquad S_{\mathcal{E},\mathcal{R}_\mathcal{E}} \cdot \mathbf{v}^k_\mathcal{E} - \mu^k \cdot \mathbf{E}^k = 0, \\
&\qquad S_{\mathcal{Q},\mathcal{R}_\mathcal{Q}} \cdot \mathbf{v}^k_\mathcal{Q} - \mu^k \cdot \mathbf{Q}^k = 0, \\
&\qquad \sum_{i \in cat(E)} \frac{|\mathbf{v}^k_i|}{k^{i,E}_{cat}} \leq E^k, \quad \text{for all } E \in \mathcal{E} \setminus \mathcal{T} \\
&\qquad \mathbf{v}^k_i = -\frac{\mathbf{N}^{k-1}_i}{K^i_m + \mathbf{N}^{k-1}_i} \cdot k^{i,T}_{cat} \cdot T^k, \quad \text{for all } i \in \mathcal{R}_\mathcal{N},\ T \in \mathcal{T} \\
&\qquad \mathbf{d}^T_\mathcal{P} \cdot \mathbf{P}^k \leq D, \\
&\qquad \mathbf{Q}^k = \mathbf{Q}_{fix}, \\
&\qquad \overline{p}^k = f_i(\mathbf{Y}^{k-1}, \mathbf{v}^{k-1}), \quad \forall\, p \in \mathcal{E} \\
&\qquad \mathbf{lb}_p \cdot \overline{p}^k \leq \mathbf{v}^k_p \leq \mathbf{ub}_p \cdot \overline{p}^k, \quad \forall\, p \in \mathcal{E} \\
&\qquad \mathbf{lb} \leq \mathbf{v}^k \leq \mathbf{ub}, \\
&\qquad \mathbf{P}^k,\ \mu^k \geq 0,\ \overline{p}^k \in \{0, 1\},
\end{aligned}
\tag{4.7}
$$

where $\mathbf{lb}, \mathbf{ub}$ denote the original lower and upper bounds on reaction fluxes.

In riRBA, the biomass and external metabolite concentrations are updated with Eq. 2.3 and Eq. 2.4. The resulting flux distribution and external metabolite concentrations are further used to determine the reaction activity states at the next time step. The initial medium is indicated by the initial values of the concentrations of external metabolites and biomass.

We note that the initial values of the Boolean variables $\overline{p}, p \in \mathcal{E}$ cannot be directly calculated using Eq. 4.5. This is because even though initial concentrations of external metabolites can be set as an environmental condition, the initial reaction fluxes at the starting time point are hard to determine. Therefore, when running riRBA, the Boolean variables also have to be initialized by the modeler. We suggest using a stable state of the gene regulatory network to avoid oscillation in the beginning of the simulation.

Solving riRBA also involves two computational loops, like in iRBA. The inner loop applies the bisection strategy to solve riRBA$(\mu)^k$ for obtaining the maximal growth rate μ^k and fluxes $\mathbf{v}^k$. In the outer loop, riRBA$(\mu)^k$ is iteratively solved from time step 1 to n. The biomass and external metabolite concentrations are updated in the outer loop by Eq. 2.3 and Eq. 2.4.

Unlike iRBA, riRBA additionally updates the gene states of regulated proteins according to the regulatory rules. As enzyme productions are included in the resource allocation models, riRBA controls the production of regulated enzymes. This is different from the strategy used in rFBA. Without considering protein expression, rFBA directly knocks out intermediate reactions when their associated genes are inhibited via the gene-reaction mapping. In riRBA, the enzyme production is inhibited when the state of its encoding gene is `off`. Finally, we note that the enzyme synthesis is constrained with the expression state $\overline{p}^k, p \in \mathcal{E}$. The production of regulatory proteins is not accounted for. Like in rFBA, we only use the gene states of the regulatory proteins to update the expression states of the enzymes, whose abundances further control the metabolic activities.

4.4 Comparison between DFBA, rFBA, iRBA and riRBA

4.4.1 A core carbon metabolic-genetic network and regulatory rules

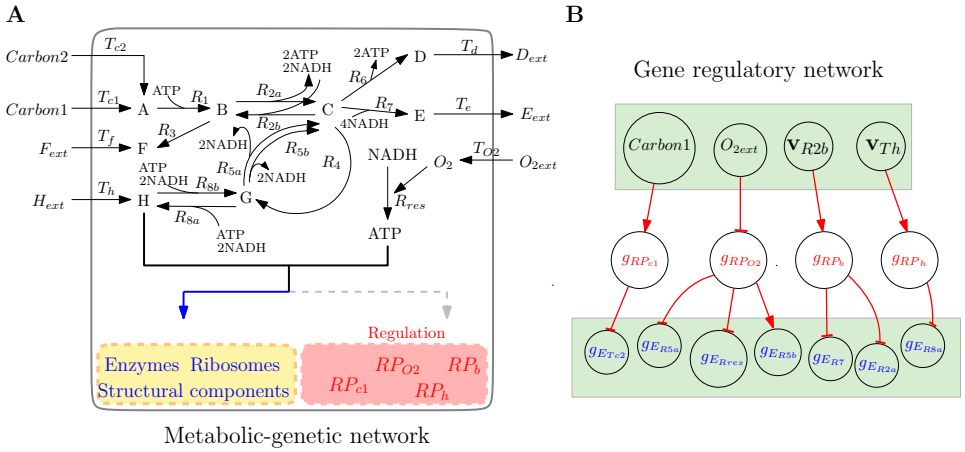


Figure 4.1: Metabolic-genetic network of core carbon model with gene regulatory rules. The gray-dashed arrow line directing to the regulatory proteins indicates that their production reactions are ignored in riRBA.

In this section, we illustrate the implementation of our iRBA and riRBA on the core carbon model. It was originally created to mimic the central carbon metabolism in (Covert et al., 2001), which was described in Section 3.3.1. To build the relevant metabolic-genetic network, reactions producing transporters, enzymes, ribosome, and also structural components were added following (Waldherr et al., 2015) (see Figure 4.1).

Catabolism of the core carbon network —the upper part of the metabolic-genetic network shown in Figure 4.1 —has already been described in Section 3.3.1. Besides catabolism, the anabolism is added to include the enzyme production (yellow box in Figure 4.1). The stoichiometric information and kinetic constants are given based on (Waldherr et al., 2015), see the details in Table 4.2 and Table 4.3. For

the density constraint, we define the density coefficients of macromolecules $\mathbf{d}_{\mathcal{P}}$ according to their length in amino acids, which are indicated by the numbers of H in the core carbon network.

With the notation from Section 2.3.1, we have the following molecular species:

$$\begin{aligned}
\mathcal{X} &= \{A, B, C, D, E, F, G, H, NADH, ATP, O_2\}, \\
\mathcal{Y} &= \{Carbon\ 1, Carbon\ 2, O_{2ext}, D_{ext}, E_{ext}, F_{ext}, H_{ext}\}, \\
\mathcal{E} &= \{E_{Tc1}, E_{Tc2}, E_{R1}, E_{R2a}, E_{R2b}, E_{R3}, E_{R4}, E_{R5a}, E_{R5b}, \\
&\quad E_{R6}, E_{R7}, E_{R8a}, E_{R8b}, E_{Tf}, E_{Th}, E_{Rres}, E_Q, Q, R\}, \\
\mathcal{T} &= \{E_{Tc1}, E_{Tc2}, E_{Tf}, E_{Th}, Q\} \subseteq \mathcal{E}, \\
\mathcal{Q} &= \{Q\}.
\end{aligned} \tag{4.8}$$

Reaction	Name i	Enzyme E	Turnover rate $k_{cat}^{i,E}$ [1/min]
Exchange reactions			
$Carbon\ 1 \rightarrow A$	T_{c1}	E_{Tc1}	3000
$Carbon\ 2 \rightarrow A$	T_{c2}	E_{Tc2}	2000
$F_{ext} \rightarrow F$	T_f	E_{Tf}	3000
$H_{ext} \rightarrow H$	T_h	E_{Th}	3000
$O_{2ext} \rightarrow O_2$	T_{O2}	Q	1000
$D \rightarrow D_{ext}$	T_d	Q	1000
$E \rightarrow E_{ext}$	T_e	Q	1000
Metabolic reactions			
$A + ATP \rightarrow B$	R_1	E_{R1}	1800
$B \rightarrow C + 2ATP + 2NADH$	R_{2a}	E_{R2a}	1800
$C + 2ATP + 2NADH \rightarrow B$	R_{2b}	E_{R2b}	1800
$C \rightarrow 2ATP + 3D$	R_6	E_{R6}	1800
$C + 4NADH \rightarrow 3E$	R_7	E_{R7}	1800
$B \rightarrow F$	R_3	E_{R3}	1800
$C \rightarrow G$	R_4	E_{R4}	1800
$G + ATP + 2NADH \rightarrow H$	R_{8a}	E_{R8a}	1800
$H \rightarrow G + ATP + 2NADH$	R_{8b}	E_{R8b}	1800
$G \rightarrow 0.8C + 2NADH$	R_{5a}	E_{R5a}	1800
$G \rightarrow 0.8C + 2NADH$	R_{5b}	E_{R5b}	1800
$O_2 + NADH \rightarrow ATP$	$Rres$	E_{Rres}	1800

Table 4.2: Metabolic reactions with corresponding enzymes and turnover rates.

To obtain reasonable flux bounds on reactions to describe the diffusive exchange across the plasma membrane, we define the structural component Q as the enzymatic macromolecule for the exchange reactions T_d, T_e and T_{O2}. However, we also classify Q as the quota compound Q whose concentration is fixed for running RBA.

Macromolecular synthesis reaction $\mathbf{n}_p H + \mathbf{m}_p ATP \rightarrow p$	Density coefficient $\mathbf{d}_p$	Turnover rate $k_{cat}^{p,R}$ [1/min]	Gene
Synthesis reactions of enzymes			
$400H + 1600ATP \rightarrow E_{Tc1}$	4	2.5	$g_{E_{Tc1}}$
$1500H + 6000ATP \rightarrow E_{Tc2}$	15	0.67	$g_{E_{Tc2}}$
$400H + 1600ATP \rightarrow E_{Tf}$	4	2.5	$g_{E_{Tf}}$
$400H + 1600ATP \rightarrow E_{Th}$	4	2.5	$g_{E_{Th}}$
$500H + 2000ATP \rightarrow E_{R1}$	5	2	$g_{E_{R1}}$
$500H + 2000ATP \rightarrow E_{R2a}$	5	2	$g_{E_{R2a}}$
$500H + 2000ATP \rightarrow E_{R2b}$	5	2	$g_{E_{R2b}}$
$1000H + 4000ATP \rightarrow E_{R6}$	10	1	$g_{E_{R6}}$
$1000H + 4000ATP \rightarrow E_{R7}$	10	1	$g_{E_{R7}}$
$2000H + 8000ATP \rightarrow E_{R3}$	20	0.5	$g_{E_{R3}}$
$500H + 2000ATP \rightarrow E_{R4}$	5	2	$g_{E_{R4}}$
$4000H + 16000ATP \rightarrow E_{R8a}$	40	0.25	$g_{E_{R8a}}$
$4000H + 16000ATP \rightarrow E_{R8b}$	40	0.25	$g_{E_{R8b}}$
$500H + 2000ATP \rightarrow E_{R5a}$	5	2	$g_{E_{R5a}}$
$500H + 2000ATP \rightarrow E_{R5b}$	5	2	$g_{E_{R5b}}$
$500H + 2000ATP \rightarrow E_{Rres}$	5	2	$g_{E_{Rres}}$
$500H + 2000ATP \rightarrow E_Q$	5	2	g_{E_Q}
$4500H + 21000ATP + 1500C \rightarrow R$	60	0.2	g_R
Synthesis reaction of structural components			
	$\mathbf{d}_Q$	k_{cat}^{Q,E_Q}	
$250H + 1500ATP + 250C + 250F \rightarrow Q$	7.5	3	

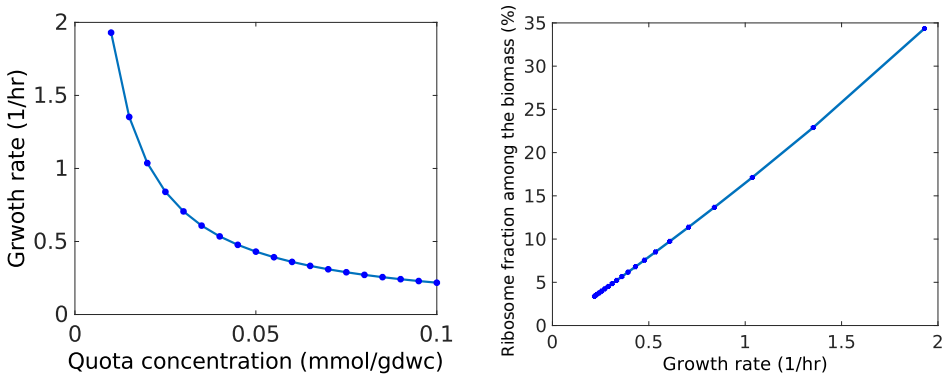
Table 4.3: Macromolecule synthesis reactions with corresponding genes, density coefficients, and turnover rates.

The corresponding gene regulatory network is shown in Figure 4.1B. The four genes in the middle layer are the genes that encode the four regulatory proteins RP_{o2}, RP_{c1}, RP_h and RP_b as described in Table 3.1. Regarding these regulatory rules, we briefly recall the role of the four regulatory proteins. The expression state of $g_{RP_{c1}}$ is activated by the presence of *Carbon* 1. It represses the expression state of $g_{E_{Tc2}}$ so that the model cannot uptake *Carbon* 2 if *Carbon* 1 is present in the medium. $g_{RP_{O2}}$ is `off` when oxygen O_{2ext} is in the environment. RP_{O2} then regulates the expression states of $g_{E_{R5a}}$ and $g_{E_{R5b}}$, which encode the isoenzymes for the aerobic and anaerobic pathways respectively. Lastly, g_{RP_b} is active when $\mathbf{v}_{R2b} > 0$, and g_{RP_b} then inhibits the expression states of $g_{E_{R2a}}$ and $g_{E_{R7}}$. The metabolic-genetic network and the regulation part are shown in Figure 4.1.

Note that we only consider the gene states of the regulatory proteins (the middle layer in Figure 4.1B) to further affect the enzyme synthesis. The production reactions of RP_{o2}, RP_{c1}, RP_h and RP_b are not included.

4.4.2 Investigating the relationship between fixed quota compound and maximal growth rate

The concentration of the structural component Q should be fixed for solving the RBA model, as explained in Section 2.3.3. However, in order to achieve a higher growth rate, the cell prefers to produce less quota compounds, since such macromolecules do not have any catalytic functionality but only cost resources. So, the value to which the quota compound should be fixed is important for the application of RBA. Here, we investigate the relationship between the fixed value of Q and the maximal growth rate in the core carbon network. To do so, we run RBA while changing the fixed value of Q from 0.005 to 0.1. The ways in which the maximal growth rate is affected by the fixed Q is shown in Figure 4.2a. We can see that as the fixed concentration of Q is increasing, the maximal growth rate resulting from RBA is decreasing. Therefore, when utilizing RBA, it is better to choose a precise quota concentration based on experiment in order to get a prediction closer to the real cells, cf. (Goelzer et al., 2011; Goelzer and Fromion, 2011; Goelzer et al., 2015).



(a) (b)

Figure 4.2: Relationships of the optimal growth rate with the fixed concentration of Q and how ribosome fraction among biomass is affected by the predicted optimal growth rate.

We also investigate how the ribosome concentration changes with the obtained growth rate while varying Q. The result in Figure 4.2b illustrates the linear relationship between ribosome fraction among biomass and growth rate, which was validated by several studies (Scott et al., 2014; Bosdriesz et al., 2015)

4.4.3 Predictions by riRBA, iRBA, rFBA and DFBA

In this section, we compare the predictions of riRBA, iRBA, rFBA, and DFBA and show the differences of the four approaches. Note that the metabolic models required for these four methods are different. Specifically, iRBA uses the metabolic-genetic network detailed in Table 4.2 and Table 4.3, which includes additional information about the protein expression as compared to the metabolic network

shown in Figure 3.2a. riRBA additionally integrates the gene regulatory network that is shown in Figure 4.1B. For DFBA, we use the metabolic network described in Table 3.1 without the regulatory rules required for applying rFBA.

We individually implemented riRBA, iRBA, rFBA and DFBA to predict the batch growth and compare the predictions. All the simulations aim at studying the diauxic shift between *Carbon* 1 and *Carbon* 2, which are set to 50 and 5 mM initially for the four methods. The oxygen concentration O_{2ext} is given as 1000 mM in excess and H_{ext}, F_{ext} are set to 0. So, initially ($Carbon$ 1, $Carbon$ 2, F_{ext}, H_{ext}, O_{2ext}, D_{ext}, E_{ext})(t_0) = (50, 5, 0, 0, 1000, 0, 0). The initial gene expression states for riRBA and rFBA are set according to (Covert et al., 2001). Using the same initial conditions, the dynamic reaction fluxes are predicted by riRBA, iRBA, rFBA and DFBA respectively.

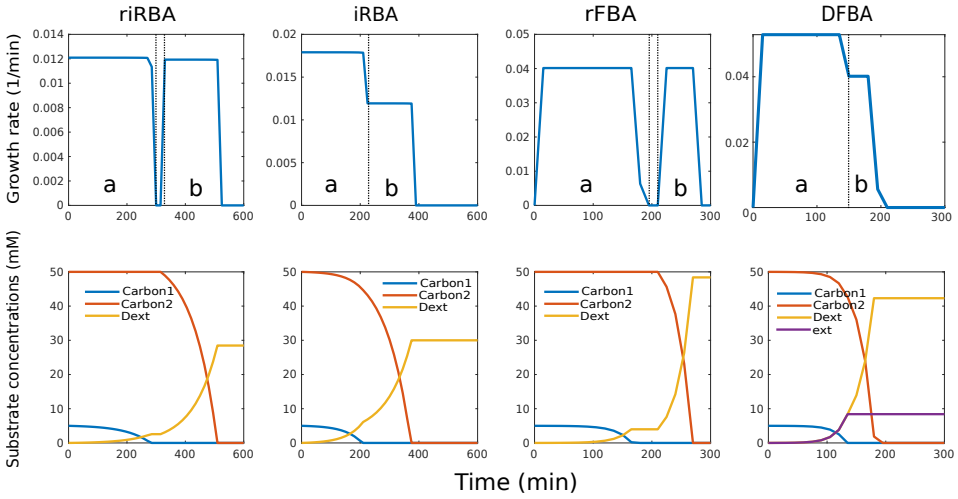


Figure 4.3: Optimal growth rate and external metabolite concentrations predicted by riRBA, iRBA, rFBA and DFBA.

All methods are able to predict the dynamics of metabolism during a simulation period. In Figure 4.3, each growth phase is divided into phase (a) and (b). We can see that the lag phase, the significant phase of diauxic growth, can only be predicted by riRBA and rFBA but not by iRBA and DFBA. Because of the regulatory rules in riRBA and rFBA, the model cannot metabolize *Carbon* 2 initially since the activity of the exchange reaction of *Carbon* 2 is repressed by the presence of *Carbon* 1 through regulatory protein RP_{c1}. After *Carbon* 1 has been used up, the cell needs a short period to disappear RP_{c1} and produce E_{Tc2} to uptake *Carbon* 2. This leads to a lag phase, in which the cells do not grow. In iRBA and DFBA, gene regulation is not considered and the model uses both *Carbon* 1 and *Carbon* 2 in phase (a). After *Carbon* 1 is exhausted the cell can absorb *Carbon* 2. This is growth phase (b) of the predictions in iRBA and DFBA, as shown in Figure 4.3 and Figure 4.5. Using both *Carbon* 1 and *Carbon* 2 during growth phase (a), the model of DFBA reaches an

optimal growth rate 0.05 1/min approximately. However, the optimal growth rate in growth phase (a) of rFBA prediction is around 0.04 1/min. As compared with the rFBA prediction, the cell would grow faster in DFBA result while the carbon sources are exhausted earlier, see Figure 4.3.

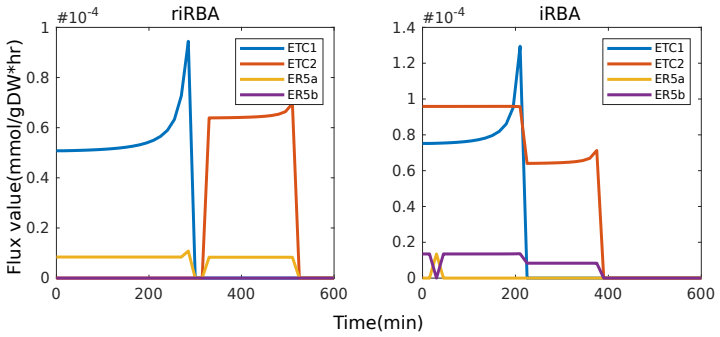


Figure 4.4: Production rates of E_{Tc1}, E_{Tc2}, E_{R5a}, E_{R5b} predicted by riRBA and iRBA.

From Figure 4.4, we can see that transporter E_{Tc2} has a higher concentration than E_{Tc1} in growth phase (a) in the prediction of iRBA. Consistent with the transporter concentration, both carbon sources are used in growth phase (a). However, it is inconsistent with the gene regulation. Unlike DFBA, the enzyme production costs are covered in iRBA. The model prefers to use *Carbon* 1 in iRBA because building up E_{Tc2} consumes more energy than E_{Tc1} (see Table 4.3). During growth phase (a), iRBA gets a growth rate that is larger than the one of riRBA as the iRBA model takes up both *Carbon* 1 and *Carbon* 2 in parallel, see Figure 4.3.

The resource allocation can be predicted by riRBA and iRBA in contrast to rFBA and DFBA. Figure 4.4 shows four reaction fluxes producing E_{Tc1}, E_{Tc2}, E_{R5a}, E_{R5b}. We can see that in the result of riRBA, the model produces E_{Tc1} first and only starts to produce E_{Tc2} after *Carbon* 1 is used up. E_{R5a} is produced during the whole growth phase since O_{2ext} is excess in this case. Yet, the result of iRBA is not consistent with the regulation without the consideration of regulatory rules. Both E_{Tc1} and E_{Tc2} are produced first in the iRBA prediction. In fact, reaction R_{5a} and R_{5b} have the same role for the model. Hence, E_{R5a} and E_{R5b} can be chosen alternatively from the perspective of optimization.

Figure 4.5 displays the activity states of the reactions. We can see how the active reactions shift from growth phase (a) to phase (b) and the differences between the four methods. According to the regulatory rules, reactions T_{c2} and R_{5b} are repressed by external *Carbon* 1 and O_{2ext} in growth phase (a), which is only predicted by riRBA and rFBA (red crosses in Figure 4.5). From the results of iRBA and DFBA in Figure 4.5, we can see that T_{c2} is active in phase (a) and reactions R_{5a} and R_{5b} are picked, which is in contradiction to the regulation. DFBA uses R_{5b} for respiration during the whole growth course, which is actually the fermentation reaction. In iRBA, there is also no regulation to ensure that the model chooses R_{5a}

for respiration. R_{5a} is active first, then switches to R_{5b} during growth phase (b) (see Figure 4.5).

Moreover, we see in the comparison whether or not the approaches include enzyme costs and regulation. As described above, the regulatory constraints are only included in riRBA and rFBA. The enzyme production costs are considered in riRBA and iRBA. The differences are shown in Figure 4.5. The gene regulation is marked in red, as shown in Figure 4.5, and the protein costs reactions are marked in blue.

4.5 Conclusions and discussion

In this chapter, we presented an intuitive way to extend RBA while accounting for dynamical growth and gene regulation, which resulted in two methods: iRBA and riRBA. Both successfully predict the dynamics of metabolism during the simulation period, based on the optimal principle. Using the core carbon network, the diauxic shift between two carbon sources has been investigated and compared with the results of DFBA and rFBA. The metabolic dynamics resulting from iRBA is due to the consideration of the changing nutrients and resource costs for the optimal growth rate. In riRBA, metabolism is additionally controlled by the gene regulatory rules, which guarantees a consistency between metabolism and gene regulation.

The differences between riRBA, iRBA, rFBA, and DFBA can be seen from the comparison in Figure 4.5. Without covering the enzyme cost reactions, DFBA and rFBA are incapable of predicting the resource allocation. DFBA and iRBA fail to predict correctly the active reactions consistent with gene regulation as the regulatory events are not considered. The situation is even worse for DFBA and iRBA when there are alternative reactions that play the same role in the network. Using only optimization without any gene regulatory information, they cannot guarantee to choose the right pathway.

Although regulation and enzyme costs are both considered in riRBA, such processes are not integrated seamlessly within a continuous dynamic framework. The regulatory protein expressions are also not included. Besides the steady-state assumption for intermediate metabolites, an assumption must be made in the application of iRBA and riRBA, which is that the protein abundances between time steps are independent and can be adjusted freely from one time step to the next. This is why we obtain the iRBA and riRBA formulas in Eq. 4.2 and Eq. 4.7. Regarding this problem, as mentioned before, Yang *et al.* introduced protein "inertia" constraints in dynamicME (Yang et al., 2019). Yet, dynamicME is still an iterative optimization-based framework, see Table 4.1. Furthermore, gene regulation is not included. As a more powerful mathematical framework, we propose in the next chapters a hybrid discrete-continuous system to unify the metabolic reactions, macromolecule synthesis, and transcriptional regulation.

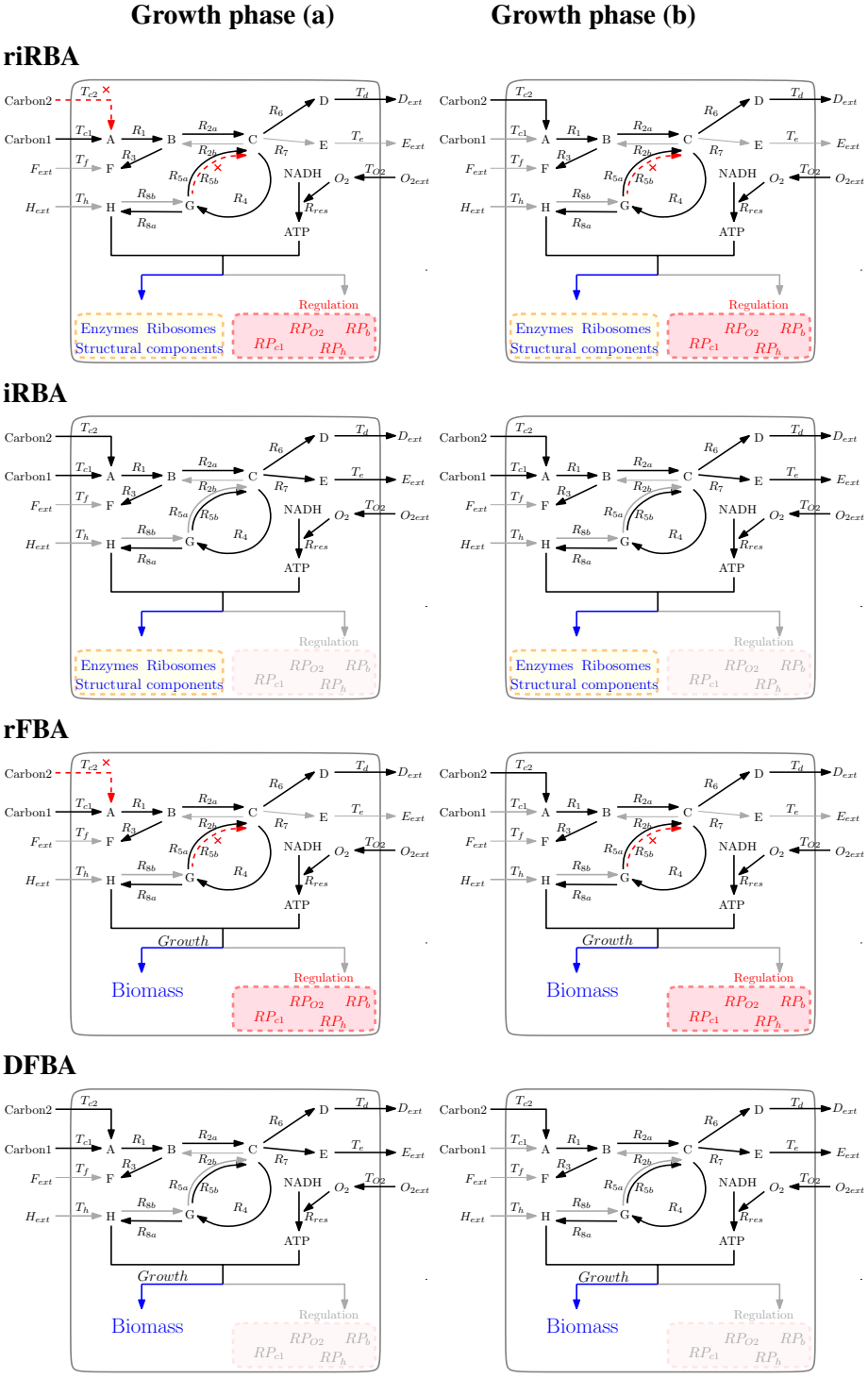


Figure 4.5: Metabolic activities predicted by riRBA, iRBA, rFBA and DFBA. The growth phase (a) and growth phase (b) are the growth phases shown in Figure 4.3. Grey lines represent the inactive reactions. The red crosses and red dotted lines are the reactions repressed according to the regulatory rules. The gray 'regulation' means the gene regulation is not included in the relevant method.

Chapter 5

Formalizing metabolic-regulatory networks by hybrid automata

In this chapter, we present a *metabolic-regulatory network* model (MRN) that allows integrating metabolism with transcriptional regulation, macromolecule production and enzyme resources. Using this model, we show that the dynamic interplay between these different cellular processes can be formalized by a hybrid automaton, combining continuous dynamics and discrete control. To validate the formalization, we build a hybrid automaton for the carbon catabolite repression to model the diauxie. The formalization in this chapter provides a theoretical basis for the following chapters. The work in this chapter has been done in collaboration with Alexander Bockmayr and material is from: 'Liu, L., Bockmayr, A. Formalizing Metabolic-Regulatory Networks by Hybrid Automata. Acta Biotheor 68, 73–85 (2020).' It is available in https://doi.org/10.1007/s10441-019-09354-y (Liu and Bockmayr, 2019a,b)

5.1 Introduction

Computational approaches in systems biology have become a powerful tool for understanding the fundamental mechanisms of cellular metabolism and regulation. However, the interplay between the regulatory and the metabolic system is still poorly understood. In particular, there is a need for formal mathematical frameworks that allow analyzing metabolism together with dynamic enzyme resources and regulatory events.

Concerning integrated modeling of metabolism and regulation, as summarized in Table 4.1 there exist approaches such as regulatory flux balance analysis (rFBA) (Covert et al., 2001) and FlexFlux (Marmiesse et al., 2015) that combine Boolean

or multi-valued logical rules for transcriptional regulation with a steady-state stoichiometric model of metabolism. These techniques iterate flux balance analysis (FBA) by splitting the growth phase into discrete time steps. At each time step, the updated regulatory states are imposed as bounds on the reaction fluxes while ignoring the costs for enzyme production. At a different level, there exist methods to predict metabolic resource allocation considering enzyme-catalytic relationships, either at steady-state (RBA (Goelzer et al., 2011), ME-models (Lerman et al., 2012)) or in a dynamic setting (deFBA (Waldherr et al., 2015), cFBA (Rügen et al., 2015; Reimers et al., 2017a), dRBA (Jeanne et al., 2018), dynamicME (Yang et al., 2019)). But, regulation is not included in these approaches. Besides Boolean logic and stoichiometric models, piecewise-linear differential equations (Ropers et al., 2006; Chaves et al., 2019) and other types of hybrid systems (Bockmayr and Courtois, 2002; De Jong et al., 2003; Bortolussi and Policriti, 2008) have also been used to study the dynamics of metabolic-genetic networks. Most of these studies, however, merely consider metabolism and regulation, and do not combine these with macromolecule production and enzymatic relationships.

In the present chapter, we introduce a *metabolic-regulatory network* model (MRN) extending the self-replicator system proposed in (Molenaar et al., 2009). Our modeling framework allows integrating metabolism with transcriptional regulation, macromolecule production, enzyme resources, and structural building blocks. Using this framework, we show that the dynamic interplay between cellular metabolism, macromolecule production and regulation can be formalized by a hybrid automaton, combining continuous dynamics and discrete control. In this formalization, the amounts of molecular species are represented by continuous variables. The discrete states of the system are composed of all gene expression states for the regulated proteins, which include regulatory proteins and regulated enzymes. In each discrete state, the continuous variables evolve according to a system of differential equations that is specific for this state. The guard conditions for the state transitions depend on the amounts of the molecular species and associated thresholds.

Our formalization makes it possible to apply hybrid system tools for analyzing metabolic-regulatory cellular processes. Compared to the approaches mentioned above, this will allow us including regulation, macromolecule production and enzyme resources into the prediction of the dynamics of cellular metabolism.

To validate our approach, we present a hybrid automaton for a simplified model of the diauxic shift in bacteria, which is inspired by the work in (Molenaar et al., 2009; Covert et al., 2001). We illustrate three possible dynamic trajectories of the model using the syntax and semantics of hybrid automata. The results demonstrate that our formalization in this chapter extends the mathematical frameworks of diauxie, as reviewed in paper (Kremling et al., 2018), which integrates the continuous dynamics of metabolism with the discrete regulation and resource allocation.

5.2 Construction of the MRN model

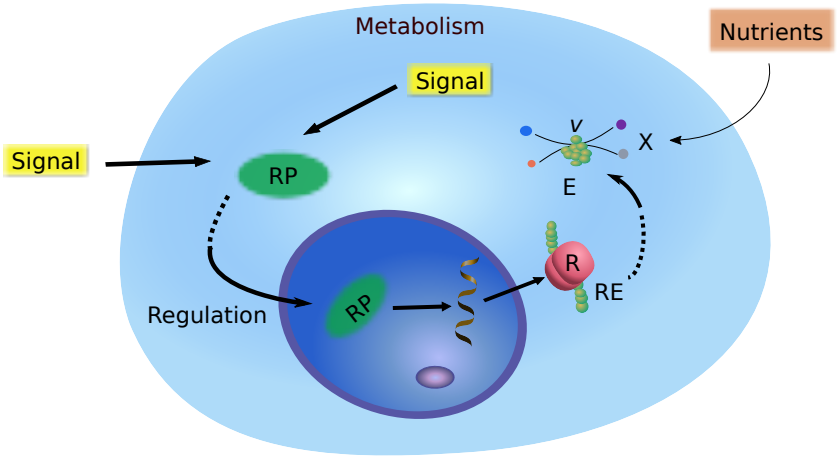


Figure 5.1: Typical metabolic-regulatory processes in response to the external/internal signals, with regulatory proteins *RP*, ribosomes *R*, regulated enzymes *RE*, enzymes *E*, metabolites *X*, and fluxes *v*.

In Figure 5.1, we show the typical regulatory processes in a cell in response to external/internal signals (Simmons et al., 2009). Regulatory proteins $\mathcal{RP}$ first sense extra- or intracellular signals and transmit these to the gene expression machinery, which then alters the production of proteins, in particular enzymes $\mathcal{E}$. Changes of enzyme amounts will affect the metabolite levels which in turn provide feedback as internal signals. Furthermore, via the process of breaking nutrients into energy and building blocks (catabolism), metabolism influences the level of signaling by providing the precursors of regulatory proteins.

We formalize the interactions between metabolism and regulation by a metabolic-regulatory network (MRN) that is given in Figure 5.2. Regarding metabolism, $\mathcal{Y}$ represents the set of all external metabolites and $\mathcal{R}_\mathcal{Y}$ is the set of reactions for the conversion between external species $\mathcal{Y}$ and intermediate metabolites $\mathcal{X}$. $\mathbf{Y}, \mathbf{X}$ are vectors of the associated amounts of $\mathcal{Y}$ and $\mathcal{X}$. $\mathcal{R}_\mathcal{X}$ denotes the set of reactions for the conversion between intermediate metabolites $\mathcal{X}$. $\mathbf{v}_\mathcal{X}, \mathbf{v}_\mathcal{Y}$ denote fluxes of reactions $\mathcal{R}_\mathcal{X}$ and $\mathcal{R}_\mathcal{Y}$. The macromolecular production reactions $\mathcal{R}_{\mathcal{RP}}$, $\mathcal{R}_\mathcal{Q}$ and $\mathcal{R}_\mathcal{E} = \mathcal{R}_{\mathcal{RE}} \cup \mathcal{R}_{\mathcal{NRE}}$ use the intermediate metabolites $\mathcal{X}$ to build regulatory proteins $\mathcal{RP}$, non-catalytic macromolecules $\mathcal{Q}$, and enzymes $\mathcal{E}$. To keep the model simple, the set of enzymes $\mathcal{E}$ contains all catalytic molecules, including transporters and ribosomes. However, we distinguish between regulated enzymes $\mathcal{RE}$ and non-regulated enzymes $\mathcal{NRE}$, i.e., $\mathcal{E} = \mathcal{RE} \cup \mathcal{NRE}$. Non-catalytic macromolecules, termed as quota compounds $\mathcal{Q}$ (Reimers et al., 2017b), e.g. DNA and lipids, are included in the model because they are essential for growth and consume a lot of cellular resources.

In the metabolic part, our network is inspired from the self-replicator model by

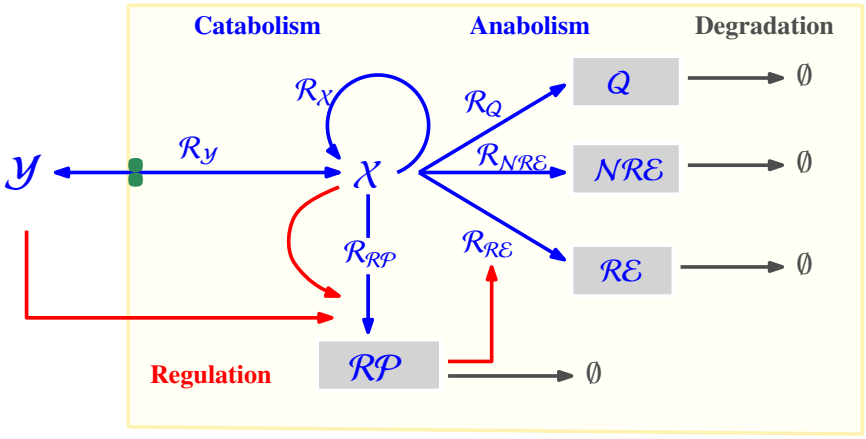


Figure 5.2: Schematic model of a metabolic-regulatory network (MRN). $\mathcal{Y}$ represents extracellular metabolites with the associated molar amounts $\mathbf{Y}$. $\mathcal{R}_\mathcal{Y}$ is the set of exchange reactions between extracellular species $\mathcal{Y}$ and intermediate metabolites $\mathcal{X}$. $\mathbf{X}$ represents molar amounts of intermediate species and $\mathcal{R}_\mathcal{X}$ is a set of intermediate reactions. Instead of classifying the macromolecules as transporters, enzymes and ribosomes, we use $\mathcal{E}$ to represent all the catalytic macromolecules here with molar amounts $\mathbf{E}$. $\mathcal{Q}$ denotes a set of non-catalytic macromolecules and $\mathbf{Q}$ is a vector of their molar amounts. In order to include regulation, $\mathcal{E}$ are classified as regulated enzymes $\mathcal{RE}$ and non-regulated enzymes $\mathcal{NRE}$ with associated amounts $\mathbf{RE}$, $\mathbf{NRE}$. That is $\mathcal{E} = \mathcal{RE} \cup \mathcal{NRE}$. The metabolism is represented by blue lines and the gene regulation by the red lines. Additionally, the degradation of macromolecules is shown by gray lines.

Molenaar *et al.* (Molenaar et al., 2009). Compared to the FBA-type models of metabolism, these authors include metabolic resource allocation. However, they do not consider regulation. Similar to (Covert et al., 2001), we focus here on transcriptional regulation, i.e., we do not model post-transcriptional modifications.

In the following, we will show how the dynamics of the metabolic-regulatory model, i.e., the interactions between metabolism and regulation can be naturally described by a hybrid automaton.

5.3 Hybrid discrete-continuous dynamics

5.3.1 Continuous variables

Kinetic modeling of metabolic networks by ordinary differential equations (ODEs) has a long history in systems biology. Based on our metabolic-regulatory network (see Figure 5.2), we define the set of molecular species

$$\mathcal{M} = \mathcal{Y} \cup \mathcal{X} \cup \mathcal{RP} \cup \mathcal{E} \cup \mathcal{Q} \tag{5.1}$$

as the union of all extracellular metabolites, intermediate metabolites, enzymes (including ribosome), regulatory proteins, and quota compounds. In a purely continuous modeling approach, the dynamics of the network would be described by a

system of ODEs

$$\dot{\mathbf{M}}(t) = \frac{d\mathbf{M}}{dt} = F(\mathbf{M}, \mathbf{K}, S, t), \tag{5.2}$$

where $\mathbf{K}$ is a vector of kinetic parameters, S is the stoichiometric matrix and t denotes time. Following (Waldherr et al., 2015; Reimers et al., 2017b), we assume throughout this chapter that $\mathbf{M}(t)$ denotes the molar amounts of the molecular species in $\mathcal{M}$ at time t. However, it would also be possible to use $\mathbf{M}(t)$ to model instead concentrations. The function F represents the kinetic laws that govern the dynamics, which could be mass action, Michaelis-Menten, Hill kinetics etc.

5.3.2 Discrete states

Continuous modeling of gene regulatory networks is known to be very difficult due to the lack of the necessary kinetic data. Therefore, we adopt a more qualitative approach to include regulation in our model. It is based on the logical modeling framework pioneered in the 1970's by L. Glass, S. Kauffman, R. Thomas and others, see (Abou-Jaoudé et al., 2016) for a recent review. We assume that for each regulated protein p there are two possible states `on` and `off`, describing whether the gene encoding p is expressed or not.

Formally, for all $p \in \mathcal{RP} \cup \mathcal{RE}$, we introduce a Boolean variable $\overline{p} = \overline{p}(t) \in \{0, 1\}$ and a logical function $f_p : \mathbb{R}^n \to \{0, 1\}$. Here, the Boolean value 0 corresponds to `off` and the value 1 to `on`. Each function f_p is defined as a Boolean combination (using the Boolean operations $\neg$ (not), $\wedge$ (and), $\vee$ (or)) of atomic formulae $x_i \geq \theta_i$, where x_i is a real variable and θ_i is a constant. As an example, consider $f : \mathbb{R}^2 \to \{0, 1\}$, $f(x_1, x_2) = (x_1 \geq 1) \wedge \neg(x_2 \geq 2)$, for which we get $f(1, 1) = 1$ and $f(1, 2) = 0$. Overall, the regulation of our MRN is then formalized by a system of Boolean equations of the form

$$\overline{p}(t) = f_p(\mathbf{RP}(t), \mathbf{Y}(t), \mathbf{X}(t)), \qquad \text{for all } p \in \mathcal{RP} \cup \mathcal{RE}. \tag{5.3}$$

Here, f_p describes how the expression state of the gene encoding the regulated protein p depends on the current amounts of regulatory proteins, external metabolites, and intermediate metabolites.

5.4 Combining discrete and continuous dynamics in a hybrid automaton

Combining metabolism and regulation in this way leads to a hybrid discrete-continuous system. Here, all amounts of molecular species are modeled by continuous variables. However, the evolution of regulated proteins p is controlled by the expression state $\overline{p}$ of the corresponding genes. Thus, depending on the discrete state $\overline{p}$, there are two different continuous dynamics. The system will jump from one discrete state to the other if some regulatory event occurs, see Figure 5.3 for illustration.

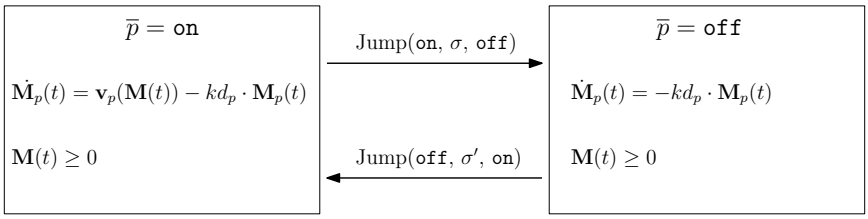


Figure 5.3: Graphic depiction of continuous evolution in the discrete states $\overline{p} = \texttt{on}$ and $\overline{p} = \texttt{off}$, for regulated proteins $p \in \mathcal{RP} \cup \mathcal{RE}$. In the on−state, protein synthesis and degradation are described by the kinetic law $\mathbf{v}_p(\mathbf{M}(t))$ resp. $-kd_p \cdot \mathbf{M}_p(t)$, with degradation constant kd_p, while in the off-state only degradation occurs. The invariant under both states is that the amounts of all molecular species cannot be negative, which is $\mathbf{M}(t) \geq 0$.

Composing the discrete states together with their continuous dynamics for all regulated proteins $p \in \mathcal{RP} \cup \mathcal{RE}$ leads to a hybrid automaton

$$H = (Loc, \Sigma, Edge, X, Init, Inv, Flow, Jump), \tag{5.4}$$

with the following components (Raskin, 2005; Henzinger, 2000):

- Loc is a finite set of discrete states or locations. Here, $Loc = \{0,1\}^{\mathcal{RP}\cup\mathcal{RE}}$ consists of all possible combinations of expression states $\overline{p} \in \{0,1\}$ of regulated proteins $p \in \mathcal{RP} \cup \mathcal{RE}$. In other words, for a MRN with n regulated proteins, there will be in total 2^n discrete states or locations in the hybrid automaton. However, not all of these have to be reachable from a given initial state.

- Σ is a finite set of events. In our case, these are given by the regulatory rules in Eq. 5.3. For instance, a regulatory event σ can be that the amount of a regulatory protein $\mathbf{RP}_i$ is larger than a certain threshold θ_i, which is expressed mathematically as $\sigma = (\mathbf{RP}_i \geq \theta_i)$.

- $Edge \subseteq Loc \times \Sigma \times Loc$ is the set of possible transitions from one location to another, which are labeled by an event from Σ.

- X is a finite set of real variables. In our case, $X = \mathcal{M} = \mathcal{Y} \cup \mathcal{X} \cup \mathcal{RP} \cup \mathcal{E} \cup \mathcal{Q}$. In each location l, these continuous variables evolve according to a specific dynamics depending on l, which is specified by the predicate $Flow(l)$.

- $Init, Inv, Flow$ are functions that assign logical predicates to each location $l \in Loc$:

  1. $Init(l)$ is a predicate which describes the possible initial values for the continuous variables when the automaton starts its execution in state l.
  2. $Inv(l)$ is a predicate which describes the possible values of the continuous variables when the control of the automaton lies in l.

3. *Flow*(l) is a predicate which describes the possible continuous evolutions when the control of the automaton is in l, for example by a system of ODEs.

- *Jump* is a function that assigns to each $e \in Edge$ a predicate *Jump*(e) describing when the discrete change modeled by e is possible and what the possible updates of the continuous variables are when this change is made.

For a formal specification of the discrete-continuous dynamics of the hybrid automaton H we refer to (Raskin, 2005; Henzinger, 2000). In the next section, we explain the main principles by an illustrative example.

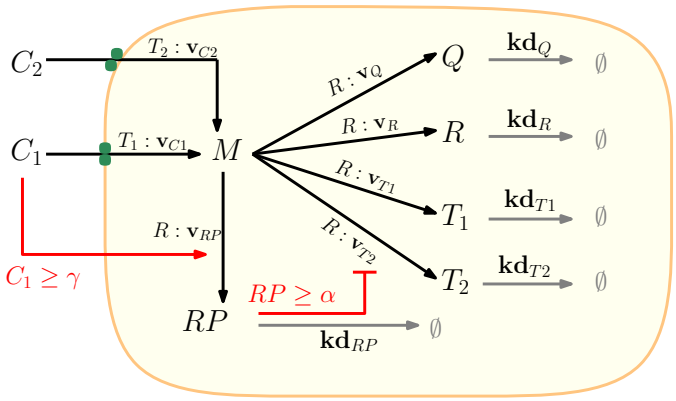


Figure 5.4: A specific model for modeling CCR with two regulatory rules. C_1, C_2 are the two carbon sources. T_1, T_2 are the enzymes for converting carbon sources into precursor M. Q denotes a house-keeping protein. RP is a regulatory protein. R represents the ribosome catalyzing the protein production. γ, α are the thresholds of the two regulatory rules.

5.5 Biological application

Carbon catabolite repression (CCR) is a common phenomenon in bacteria, especially in *Escherichia coli* (Görke and Stülke, 2008). While these bacteria are able to grow on different carbon sources, they do not consume these in parallel, but one after the other. Mathematical modeling of diauxie has played an important role in understanding these phenomena (Kremling et al., 2015, 2018). To illustrate our approach, we present here a new hybrid discrete-continuous model for CCR, which integrates continuous dynamics of metabolism with discrete regulation and resource allocation.

5.5.1 MRN model of the diauxic shift

Starting from the generic model in Figure 5.2, we consider two alternative carbon sources C_1, C_2 and a regulatory protein RP to build the MRN shown in Fig-

Parameter	Description	Value	Unit	References
k_{cat1}	Max import rate of T_1	3000	min^{-1}	
k_{cat2}	Max import rate of T_2	2000	min^{-1}	
K_T	Enzymatic threshold	1000	mmol	
kr	Max elongation rate	1260	aa min^{-1}	(Bremer et al., 1996)
Kr	Elongation threshold	7	aa	(Weiße et al., 2015)
$\mathbf{n}_Q, \mathbf{n}_{RP}$	Average length of non-ribosomal enzymes	300	aa	(Brandt et al., 2009)
$\mathbf{n}_{T1}$	Length of T_1	400	aa	
$\mathbf{n}_{T2}$	Length of T_2	1500	aa	
$\mathbf{n}_R$	Length of ribosome	7459	aa	(Keseler et al., 2010)
w	Average molar weight of amino acids	100	mg mmol^{-1}	
$\mathbf{kd}_Q, \mathbf{kd}_R, \mathbf{kd}_{T1}, \mathbf{kd}_{T2}$	Enzyme degradation rate	0.01	min^{-1}	(Kennell and Riezman, 1977)
$\mathbf{kd}_{RP}$	Regulatory protein degradation rate	0.2	min^{-1}	

Table 5.1: Model parameters. Here, aa stands for amino acids, which correspond to the precursor M in the coarse-grained CCR model of Figure 5.4.

ure 5.4. The two regulated proteins in our model are RP and T_2, with corresponding Boolean variables $\overline{RP}$ and $\overline{T}_2$. As shown by the red lines, the regulatory rules are the following: If the external amount of C_1 is above the threshold γ, the gene encoding for RP is activated. If the amount of RP inside the cell is above the threshold α, the gene encoding for T_2 is repressed. More formally:

$$\overline{RP} = 1 \Leftrightarrow (C_1 \geq \gamma) \quad \text{and} \quad \overline{T}_2 = 0 \Leftrightarrow (RP \geq \alpha) \tag{5.5}$$

Together, these regulations ensure that C_1 is the preferred carbon source for the model.

5.5.2 Hybrid automaton model of the diauxic shift

Next we construct the hybrid automaton H_{diaux} for the metabolic-regulatory network in Figure 5.4. We get the continuous variables $X_{diaux} = \{C_1, C_2, M, RP, T_1, T_2, R, Q\}$ describing time-dependent molar amounts (in mmol) and the discrete locations $Loc_{diaux} = \{(\overline{RP}, \overline{T}_2) \mid \overline{RP}, \overline{T}_2 \in \{\texttt{on}, \texttt{off}\}\}$. To specify the dynamics of H_{diaux}, we use the graphical representation in Figure 5.5, which is more intuitive than the formal definition according to Eq. 5.4.

The four nodes in Figure 5.5 correspond to the discrete states or locations. Within each node, we specify the continuous dynamics of the molecular species by a set of ODEs and some invariants. The arcs represent the discrete jumps between different states. They are labeled by the guards that the system has to satisfy in order to perform the corresponding transition.

For the uptake of C_1, C_2 we assume a Michaelis-Menten kinetics

$$\mathbf{v}_{C_1} = \frac{k_{cat1} \cdot C_1 \cdot T_1}{K_T + C_1}, \quad \mathbf{v}_{C_2} = \frac{k_{cat2} \cdot C_2 \cdot T_2}{K_T + C_2}. \tag{5.6}$$

All the parameters are listed in Table 5.1. Regarding the synthesis rate $\mathbf{v}_p$ of a protein p consisting of $\mathbf{n}_p$ amino acids, we assume like in (Faizi et al., 2018) a

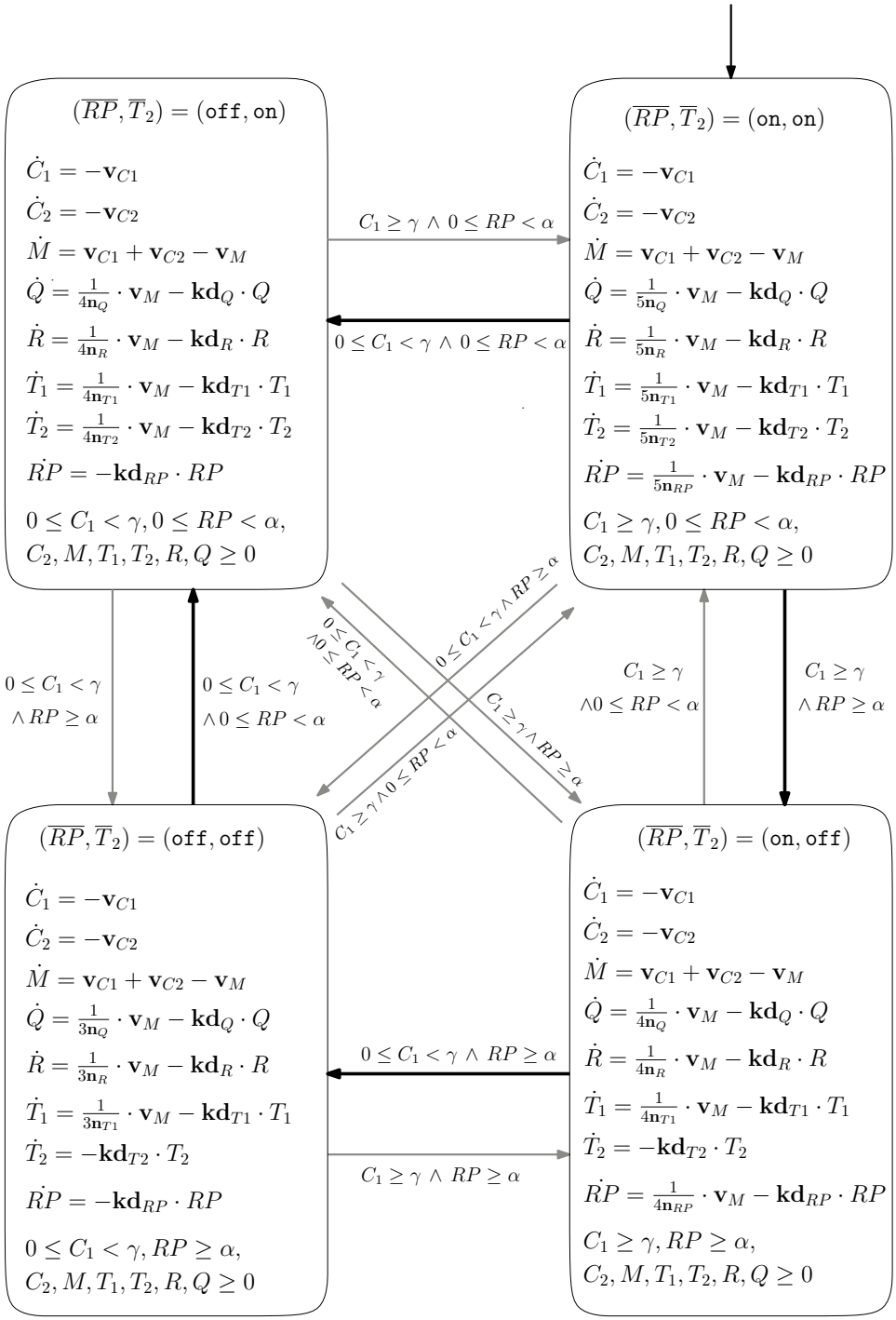


Figure 5.5: Graphic illustration of the hybrid automaton H_{diaux}.

Michaelis-Menten type kinetics of the form

$$\mathbf{v}_p = \frac{\beta_p}{\mathbf{n}_p} \cdot \mathbf{v}_M, \text{ with } \mathbf{v}_M = \frac{kr \cdot M \cdot R}{Kr + M} \text{ and } p \in \{RP, T_1, T_2, R, Q\}. \tag{5.7}$$

The weights $\beta_p \geq 0$ denote the fraction of cellular resources allocated to protein p, with $\sum_p \beta_p = 1$. For each location of the hybrid automaton shown in Figure 5.5, we assume for simplicity that the cellular resources are shared equally between the proteins that are expressed at this location. In other words, $\beta_p = 1/3$ in location $(\texttt{off}, \texttt{off})$, $\beta_p = 1/4$ in location $(\texttt{on}, \texttt{off})$ resp. $(\texttt{off}, \texttt{on})$, and $\beta_p = 1/5$ in location $(\texttt{on}, \texttt{on})$. The total biomass $Biomass(t)$ is defined as the sum of the molecular masses

$$Biomass(t) = w \cdot M(t) + \sum_{p \in \{RP, T_1, T_2, R, Q\}} w \cdot \mathbf{n}_p \cdot \mathbf{M}_p(t). \tag{5.8}$$

If $\overline{RP} = \texttt{on}$, the regulatory protein RP is built from the precursor M, catalyzed by the ribosome R. Whenever $C_1 < \gamma$ is reached, $\overline{RP}$ switches to $\texttt{off}$ and RP is degraded with rate $\mathbf{kd}_{RP}$. Similarly, the enzyme T_2 is produced if $\overline{T}_2 = \texttt{on}$, which is equivalent to $RP < \alpha$, and degraded if $\overline{T}_2 = \texttt{off}$. Due to mass balance, the dynamics of the precursor M also depends on the location and the regulatory control. The dynamics of the other variables C_1, C_2, T_1, R is not directly controlled by the discrete states, but depends on them as well via the shared variables.

5.5.3 Exploring the dynamics of H_{diaux}

Given the hybrid automaton H_{diaux}, we explore the dynamics of the system for three simulations, see Figure 5.6, Figure 5.7, and Figure 5.9. In all the three simulations, the kinetic parameters are the same and the initial state is $(\texttt{on}, \texttt{on})$. Following (Weiße et al., 2015), we assume that the cell starts growing with some positive amounts of precursors and ribosomes, but without any enzymes. Thus, the initial enzyme amounts T_1, T_2 are set to 0, so that the system first has to produce them before carbon uptake can start.

In Simulation 1 and Simulation 2, we use the hybrid automaton illustrated in Figure 5.5. In each location, the cellular resources are shared equally between the proteins whose expression states are $\texttt{on}$ at this location. The only difference between Simulations 1 and Simulation 2 is that the initial amount of C_1 is decreased from $C_1 = 500$ mmol in Simulation 1 to $C_1 = 50$ mmol in Simulation 2.

Simulation 3 has the same initial condition as Simulation 1, including initial values of continuous variables and initial discrete state. Differently, we specify β_p with different values for proteins that are expressed in each state in Simulation 3. For the specific β_p in Simulation 3, see Figure 5.8 in detail.

Simulation 1

For Simulation 1, we see in Figure 5.6 that both RP and T_2 initially increase because $(\overline{RP}, \overline{T}_2) = (\texttt{on}, \texttt{on})$. C_1, C_2 are consumed very slowly in the beginning, because the model is initialized with $T_1 = T_2 = 0$. When the amount of the regulatory protein RP reaches the threshold $\alpha = 0.03$ mmol (corresponding to 900 mg)

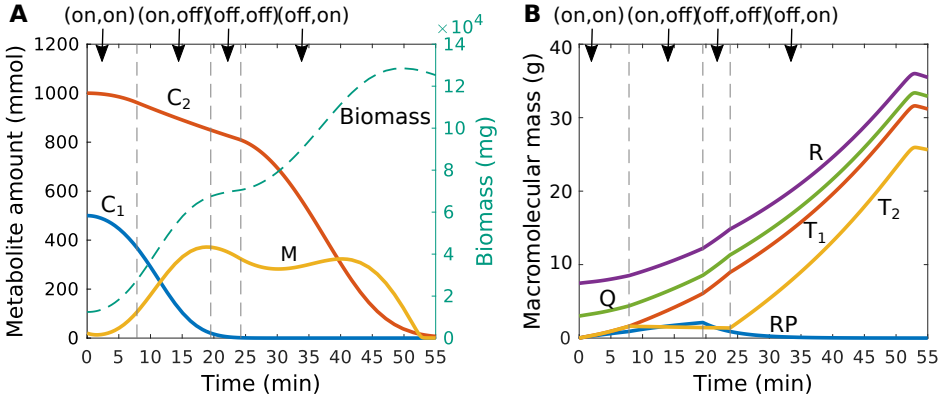


Figure 5.6: Time course for Simulation 1: (**A**) Metabolites C_1, C_2, M (left axis) and total biomass (right axis). The initial location is $(\overline{RP}, \overline{T}_2) = (\mathtt{on}, \mathtt{on})$ with initial values (C_1, C_2, M, RP, T_1, T_2, R, Q) = (500, 1000, 20, 0, 0, 0, 0.01, 0.1). Discrete state transitions are indicated by dashed lines. The thresholds are $\gamma = 20$ and $\alpha = 0.03$. (**B**) Macromolecular masses RP, T_1, T_2, Q, R.

and C_1 is still larger than $\gamma = 20$ mmol, $\overline{T}_2$ gets inactivated and the system will jump to the location $(\overline{RP}, \overline{T}_2) = (\mathtt{on}, \mathtt{off})$. Now, the enzyme T_2 stops being produced and, with T_2 getting close to 0, uptake of C_2 is not possible anymore. Thus, the system can only use C_1. The location $(\overline{RP}, \overline{T}_2) = (\mathtt{on}, \mathtt{off})$ therefore indicates a first growth phase on the preferred carbon source C_1. Once the guard condition $C_1 < 20, RP \geq 0.03$ is satisfied, the system will switch to the new state $(\mathtt{off}, \mathtt{off})$. This location represents the lag phase during diauxie, in which C_1 is exhausted while the utilization of C_2 is still repressed. Furthermore, the synthesis of the regulatory protein RP is inhibited and its amount decreases with the degradation rate $\mathbf{kd}_{RP}$. When the amount of RP reaches the threshold $\alpha = 0.03$, $\overline{T}_2$ is turned on again and we get to the final state $(\mathtt{off}, \mathtt{on})$. Now, the cell can produce T_2 and will consume carbon source C_2 until this is finally also exhausted.

Simulation 2

In Simulation 2, the initial amount of C_1 is decreased to 50 mmol. Starting again from the location $(\mathtt{on}, \mathtt{on})$, RP and T_2 will increase and both carbon sources are assimilated. Due to the smaller initial value of C_1, the event $C_1 < 20 \wedge RP < 0.03$ is triggered first in this case. Hence the system directly jumps to $(\mathtt{off}, \mathtt{on})$ by skipping $(\mathtt{on}, \mathtt{off})$ and $(\mathtt{off}, \mathtt{off})$. As we can see in Figure 5.7, there are no clear carbon switch and lag phase in this case. The regulatory protein RP does not reach the critical threshold α before C_1 is depleted. From the biological viewpoint, this means that the amount of the preferred carbon source C_1 is too small to inhibit the uptake of C_2.

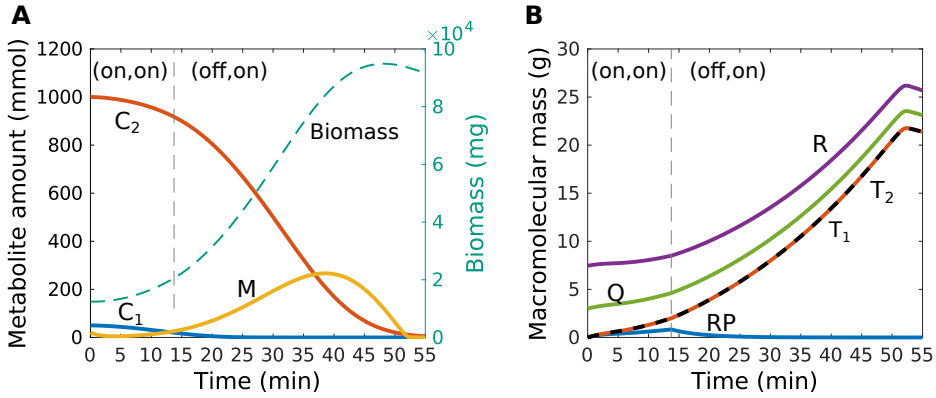


Figure 5.7: Time courses for Simulation 2: **(A)** Dynamics of metabolites C_1, C_2, M (left axis) and total biomass (right axis). The data are the same as in Simulation 1, except the initial value $C_1 = 50$. Note that there is no lag phase. The model directly jumps from $(\texttt{on}, \texttt{on})$ to $(\texttt{off}, \texttt{on})$, since the initial amount of C_1 is too small to inhibit the uptake of C_2 via RP. **(B)** Macromolecular masses RP, T_1, T_2, Q, R.

Simulation 3

Simulation 3 aims to investigate the effects of resource allocation on biomass accumulation. Specifically, the continuous dynamics in each discrete location is listed by the set of ODEs in Figure 5.8. The results are shown in Figure 5.9. Clearly, the carbon sources are exhausted earlier at nearly 35 min in this simulation while they are depleted at around 50 min in Simulation 1 (see Figure 5.6). The maximal biomass obtained is much larger than the maximal biomass obtained in Simulation 1. This simulation demonstrates that averaging the resources for all proteins expressed in each state is not an optimal strategy for the maximization of the biomass. It is probably not optimal for cellular growth. Therefore, how the cell allocates its resources significantly influences the biomass accumulation. This directly refers to the topic of predicting optimal resource allocation in cells. In order to explore the optimal control strategies for the hybrid automata representing our MRNs, we propose a constraint-based approach in the next chapter to predict the optimal resource allocation besides the dynamic metabolism and discrete state transitions.

The three simulations illustrated above exhibit just three possible behaviors of H_{diaux}. Figure 5.5 and Figure 5.8 show all state transitions that are possible based on our regulatory rules and the specific resource allocation. According to our model, the cellular behavior during diauxie depends on the whole metabolic-regulatory network. For example, in our model, the house-keeping protein Q also influences the discrete state, as it shares the precursor pool and competes with other regulated proteins for the limited ribosomes.

Regarding the diauxie phenomenon, it has been experimentally observed that the longer the cells grow in the preferred carbon source, the longer the lag phase is

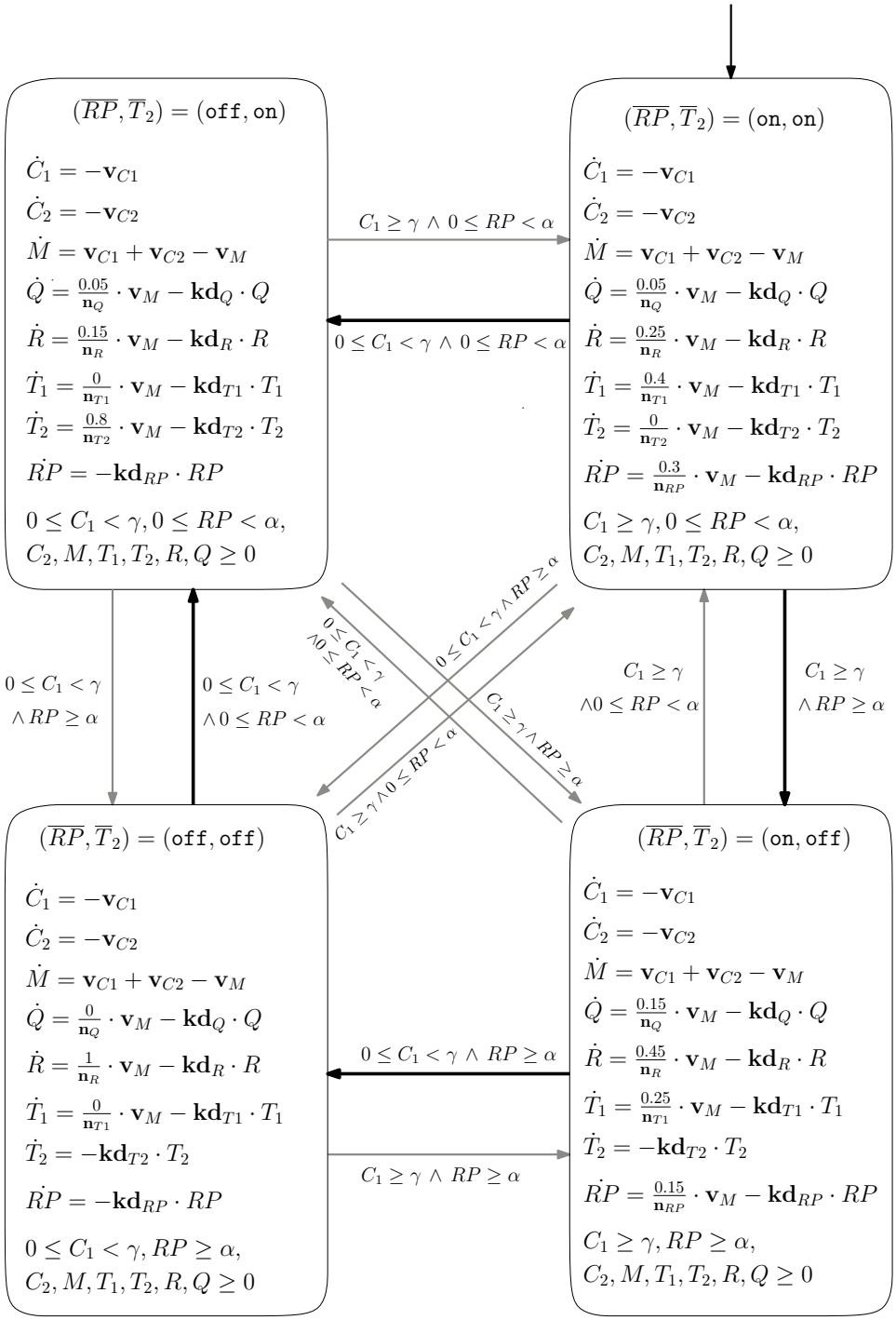


Figure 5.8: Graphic illustration of H_{diaux} for Simulation 3. Particularly, the resource fractions β_p, $p \in \{RP, T_1, T_2, R, Q\}$ are specified differently for proteins that are expressed in each location.

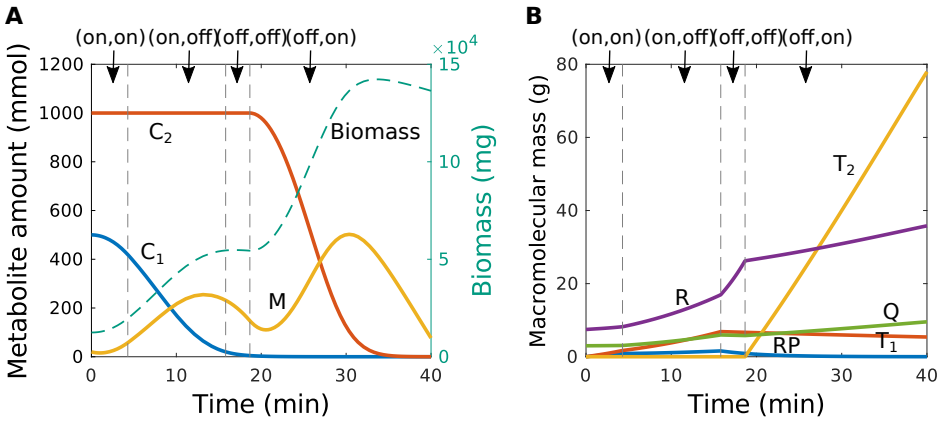


Figure 5.9: Time course for Simulation 3. **(A)** Metabolites C_1, C_2, M (left axis) and total biomass (right axis). The setting is the same as in Simulation 1, except $\beta_p, p \in \{Q, R, RP, T_1, T_2\}$ that are specified differently, see Figure 5.8. **(B)** Macromolecular masses RP, T_1, T_2, Q, R.

(New et al., 2014). Our Simulation 2 shows there is even no lag phase when the initial amount of C_1 is very small. The reason is still unclear. Yet our simulations suggest that when staying longer in the preferred carbon source, more regulatory proteins will be accumulated. Thus it will take more time to activate the second carbon source C_2. The comparison between Simulation 1 and Simulation 3 shows that the cellular resource allocation among proteins likewise decides the cellular performance. Thus indicates that there is a need for algorithms to predict the optimal dynamics of the hybrid automata representing our *metabolic-regulatory networks*.

5.6 Conclusion

In this chapter, we have proposed a new hybrid discrete-continuous modeling framework for metabolic-regulatory network that integrates metabolism, transcriptional regulation, macromolecule production and enzyme resources. Compared to classical logic-based gene regulatory networks (Thomas, 1973), we consider the expression states of proteins as discrete states or locations in a hybrid discrete-continuous system. Under each location, proteins evolve with a specific set of differential equations based on the cellular metabolic system. Thus, the time delays demanded for transmitting states are quantified by the dynamics of regulatory proteins, which is impossible in the framework of classical Boolean gene regulatory networks.

In the biological application, we illustrated the approach on a small self-replicator model and applied it to study the diauxic shift in bacteria. Compared to classic regulatory flux balance analysis (rFBA) (Covert et al., 2001), our model quantifies the synthesis and degradation of the regulatory protein, which determines the length of the lag phase. In contrast to kinetic modeling, we consider a discrete regulatory control for the continuous evolution of the molecular species. Using this formal-

ization, we can use theoretical concepts and software tools for hybrid systems, e.g. reachability analysis or model checking, to study the dynamic interplay between metabolism and regulation. We particularly introduce an approach that allows the exploration of the optimal control strategy for the hybrid system representing our MRN and predicting the optimal resource allocation in Chapter 6.

Chapter 6

Regulatory dynamic enzyme-cost flux balance analysis

In this chapter, we introduce a new constraint-based modeling framework named *regulatory dynamic enzyme-cost flux balance analysis* (r-deFBA), which unifies dynamic modeling of metabolism, cellular resource allocation and transcriptional regulation in a hybrid discrete-continuous setting. With r-deFBA, we can predict the discrete regulatory states together with the continuous dynamics of reaction fluxes, external substrates, enzymes, and regulatory proteins needed to achieve a cellular objective such as maximizing the biomass over a time interval. The dynamic optimization problem underlying r-deFBA can be reformulated as a mixed-integer linear optimization problem, for which there exist efficient solvers. The work has been done with Alexander Bockmayr (Liu and Bockmayr, 2019c), and is available as a bioRxiv preprint at `https://doi.org/10.1101/802249`.

6.1 Introduction

As can be seen from Table 4.1, there is currently no approach that integrates the three features: dynamics, macromolecule production costs and gene regulation in a unifying framework. In Chapter 5, we introduced *metabolic-regulatory networks* (MRNs) to formalize the interplay of metabolism, macromolecule synthesis and gene regulation. To specify the dynamics of MRNs, we used a hybrid automata framework, combining continuous dynamics of metabolism with discrete control by regulatory events. In this formalization, the amounts of molecular species are represented by continuous variables. The discrete states of the system correspond to gene expression states of regulated proteins, which include regulatory proteins and regulated enzymes. In each discrete state, the continuous variables evolve according to a system of differential equations that is specific for this state. The

guard conditions for the discrete state transitions depend on the amounts of the molecular species and associated thresholds.

In the present chapter, we look at dynamic optimization or optimal control of the hybrid automata representing MRNs, which leads us to a new constraint-based modeling framework called *regulatory dynamic enzyme-cost flux balance analysis* (r-deFBA). Like in other flux balance approaches, we apply a steady-state assumption for the intermediate metabolites. The resulting dynamic optimization problem can be transformed into a mixed-integer linear optimization problem (MILP), for which there exist efficient solvers.

The organization of this chapter is as follows: We start in Section 6.2 by shortly recalling the definition of MRNs and the hybrid modeling framework described in Chapter 5. In Section 6.3, we formally introduce r-deFBA by formulating the metabolic constraints, the regulatory constraints, and the resulting dynamic optimization problem. In Section 6.4, we give a possible way to transform the dynamic optimization into a MILP. To illustrate our approach, we consider two biological applications. In Section 6.5 we analyze the carbon catabolite repression model (CCR) already considered in Chapter 5. Finally, in Section 6.6, we apply our approach to a model of core carbon metabolism, inspired from (Covert et al., 2001) and (Waldherr et al., 2015).

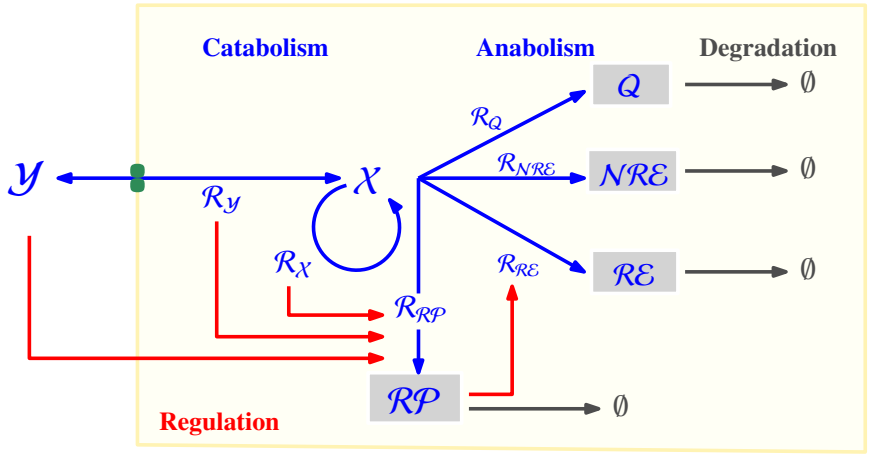


Figure 6.1: Schematic model of a metabolic-regulatory network (MRN) for presenting r-deFBA. We distinguish three types of molecular species: extracellular species $\mathcal{Y}$, intermediate metabolites $\mathcal{X}$, and macromolecules $\mathcal{P}$. $\mathbf{Y}, \mathbf{X}, \mathbf{P}$ are vectors of their molar amounts. $\mathcal{R}_{\mathcal{Y}}, \mathbf{v}_{\mathcal{Y}}$ are the exchange reactions and fluxes. $\mathcal{R}_{\mathcal{X}}$ is a set of intermediate reactions with the associated fluxes $\mathbf{v}_{\mathcal{X}}$. The macromolecules $\mathcal{P} = \mathcal{Q} \cup \mathcal{RE} \cup \mathcal{NRE} \cup \mathcal{RP}$ are classified into quota compounds $\mathcal{Q}$, non-regulated enzymes $\mathcal{NRE}$, regulated enzymes $\mathcal{RE}$, and regulatory proteins $\mathcal{RP}$ with their amounts $\mathbf{Q}, \mathbf{NRE}, \mathbf{RE}, \mathbf{RP}$. $\mathcal{R}_{\mathcal{RE}}, \mathcal{R}_{\mathcal{NRE}}, \mathcal{R}_{\mathcal{Q}}, \mathcal{R}_{\mathcal{RP}}, \mathbf{v}_{\mathcal{RE}}, \mathbf{v}_{\mathcal{NRE}}, \mathbf{v}_{\mathcal{Q}}, \mathbf{v}_{\mathcal{RP}}$ are the corresponding production reactions and fluxes. The intracellular signals are indicated by reaction fluxes $\mathbf{v}_{\mathcal{Y}}$ and $\mathbf{v}_{\mathcal{X}}$ in r-deFBA.

6.2 Hybrid dynamics of metabolic-regulatory networks

In Figure 6.1, we illustrate the schematic structure of a MRN model for presenting r-deFBA. It has the same structural components, abbreviations and notation as the metabolic-regulatory network model described in Section 5.2 shown in Figure 5.2. Therefore, the set of molecular species is defined as

$$\mathcal{M} = \mathcal{Y} \cup \mathcal{X} \cup \mathcal{P} = \mathcal{Y} \cup \mathcal{X} \cup (\mathcal{Q} \cup \mathcal{E} \cup \mathcal{RP}) = \mathcal{Y} \cup \mathcal{X} \cup (\mathcal{Q} \cup \mathcal{RE} \cup \mathcal{NRE} \cup \mathcal{RP}). \quad (6.1)$$

6.2.1 Continuous dynamics

In a purely continuous modeling approach, the dynamics of the network would be described by a system of ordinary differential equations

$$\dot{\mathbf{M}}(t) = \frac{d\mathbf{M}(t)}{dt} = F(\mathbf{M}, \mathbf{K}, S, t). \quad (6.2)$$

Following (Waldherr et al., 2015; Reimers et al., 2017b; Liu and Bockmayr, 2019b), we assume that $\mathbf{M}(t)$ denotes the molar amounts of the molecular species in $\mathcal{M}$ at time t. Furthermore, $\mathbf{K}$ is a vector of kinetic parameters, S is the stoichiometric matrix, and t denotes time. The function F represents the kinetic laws that govern the dynamics, which could be mass action, Michaelis-Menten, Hill kinetics etc.

6.2.2 Discrete control

In this context, regulatory interactions, as illustrated by the red arrows in Figure 6.1, refer to transcriptional regulation, i.e., we do not consider post-translational modifications. We assume that for each regulated protein $p \in \mathcal{RP} \cup \mathcal{RE}$ there are two possible states `on` and `off`, describing whether at a particular time t the gene encoding p is expressed or not. Formally, we introduce a Boolean variable $\overline{p} = \overline{p}(t) \in \{0, 1\}$ and a logical function $f_p : \mathbb{R}^n \to \{0, 1\}$. Here, the Boolean value 0 corresponds to `off` and the value 1 to `on`. Each function f_p is defined as a Boolean combination (using the Boolean operations $\neg$ (not), $\wedge$ (and), $\vee$ (or)) of atomic formulas of the form $x \geq \theta$, where x is a real variable and θ a threshold value.

The set of discrete states is composed of the expression states of regulated proteins $\mathcal{RE}$ and $\mathcal{RP}$ that is $Loc = \{0, 1\}^{\mathcal{RP} \cup \mathcal{RE}}$. Note that in contrast to the metabolic-regulatory networks described in Chapter 5, amounts of intermediate metabolites $\mathbf{X}(t)$ have been replaced with the flux values $\mathbf{v}_{\mathcal{X}}(t)$ and $\mathbf{v}_{\mathcal{Y}}(t)$. This is due to the steady-state assumption for intermediate metabolites, which is typical for constraint-based modeling approaches, see also in (Covert et al., 2001). Hence, the regulation of our MRN is then formalized by a system of Boolean equations of the form

$$\overline{p}(t) = f_p(\mathbf{RP}(t), \mathbf{Y}(t), \mathbf{v}_{\mathcal{Y}}(t), \mathbf{v}_{\mathcal{X}}(t)), \qquad \text{for all } p \in \mathcal{RP} \cup \mathcal{RE} \quad (6.3)$$

in r-deFBA. The logical function f_p indicates how the expression state of the regulated protein p depends on the current amounts of regulatory proteins, extracellular metabolites, and the reaction fluxes.

6.2.3 Hybrid discrete-continuous system

Combining the continuous dynamics of metabolism in Eq. 6.2 with the discrete logical control in Eq. 6.3 leads to a hybrid discrete-continuous system, which we further explore in Section 6.3.

6.3 Formalization of r-deFBA

The r-deFBA framework that we propose in this chapter aims at predicting from some initial conditions the continuous dynamics of metabolism and resource allocation together with discrete state transitions coming from genetic regulation. Compared with our earlier approach deFBA (Waldherr et al., 2015), regulatory logical constraints are included in addition to the metabolic constraints. Based on the schematic MRN model in Figure 6.1 and the notation in Section 5.2, the metabolic and regulatory constraints of r-deFBA will now be described in detail.

6.3.1 Metabolic constraints

We start by recalling the constraints on metabolism, which are derived from dynamic enzyme-cost flux balance analysis (deFBA) (Waldherr et al., 2015; Reimers et al., 2017b). deFBA is formalized in Section 2.4. Here, we shortly list these individual constraints:

- Dynamics of external metabolites $\mathcal{Y}$ (nutrients and by-products): the dynamics of the extracellular metabolites is modeled by a system of differential equations

$$\dot{\mathbf{Y}}(t) = S_{\mathcal{Y},\mathcal{R}_\mathcal{Y}} \cdot \mathbf{v}_\mathcal{Y}(t). \tag{6.4}$$

- Dynamics of macromolecules $\mathcal{P}$: the synthesis and the degradation of macromolecules are described by a system of differential equations

$$\dot{\mathbf{P}}(t) = S_{\mathcal{P},\mathcal{R}_\mathcal{P}} \cdot \mathbf{v}_\mathcal{P}(t) - \mathbf{kd}_\mathcal{P} \circ \mathbf{P}(t), \tag{6.5}$$

  where the vector $\mathbf{kd}_\mathcal{P}$ contains the degradation rates of macromolecules $\mathcal{P} = \mathcal{RP} \cup \mathcal{RE} \cup \mathcal{NRE} \cup \mathcal{Q}$ and $\circ$ denotes the component-wise product of vectors. The initial amounts of macromolecules can be specified by modeler, i.e.,

$$\mathbf{P}(t_0) = \mathbf{P}_0, \tag{6.6}$$

where $\mathbf{P}_0$ represents the given initial amounts of macromolecules. Alternatively, the initial biomass can be constrained within certain values, e.g.

$$\mathbf{b}_{\mathcal{P}}^T \cdot \mathbf{P}(t_0) = \mathrm{Bio}(t_0), \tag{6.7}$$

where $\mathrm{Bio}(t_0)$ is a constant value.

- Steady state for intermediate metabolites: for the intermediate metabolites $\mathcal{X}$ we assume that they are at steady-state, i.e., the rate of production is equal to the rate of consumption. This leads to a system of algebraic equations

$$\dot{\mathbf{X}}(t) = S_{\mathcal{X},\mathcal{R}_\mathcal{Y}} \cdot \mathbf{v}_\mathcal{Y}(t) + S_{\mathcal{X},\mathcal{R}_\mathcal{X}} \cdot \mathbf{v}_\mathcal{X}(t) + S_{\mathcal{X},\mathcal{R}_\mathcal{P}} \cdot \mathbf{v}_\mathcal{P}(t) = 0, \tag{6.8}$$

with stoichiometric matrices $S_{\mathcal{X},\mathcal{R}_\mathcal{Y}}, S_{\mathcal{X},\mathcal{R}_\mathcal{X}}, S_{\mathcal{X},\mathcal{R}_\mathcal{P}}$ and fluxes $\mathbf{v}_\mathcal{Y}, \mathbf{v}_\mathcal{X}, \mathbf{v}_\mathcal{P}$.

- Biomass composition constraint: in order to guarantee a sufficient production of non-catalytic macromolecules $\mathcal{Q}$ such as lipids and DNA, which are indispensable for cell growth and proliferation, we require that the mass of these quota compounds has to be at least a given fraction of the total biomass. Mathematically,

$$\mathbf{b}_\mathcal{Q}^T \cdot \mathbf{Q}(t) \geq \Phi_\mathcal{Q} \cdot \mathbf{b}_\mathcal{P}^T \cdot \mathbf{P}(t), \tag{6.9}$$

where $\mathbf{b}_\mathcal{P}$ is a vector with the molecular weights of all the macromolecules $\mathcal{P}$, $\mathbf{b}_\mathcal{Q}$ is the subvector of the molecular weights of the quota compounds $\mathcal{Q}$, the operation $\cdot^T$ denotes transposition, and $0 < \Phi_\mathcal{Q} < 1$ is a constant.

- Enzymatic and translational capacity constraints: fluxes through enzyme-catalyzed reactions are bounded by the amount of the corresponding enzymes. If an enzyme catalyzes more than one reaction, the sum of all the reaction fluxes is limited by the enzyme amount. Formally, we get

$$\sum_{i \in cat(E)} \frac{|\mathbf{v}_i(t)|}{k_{cat}^{i,E}} \leq E(t), \quad \text{for all } E \in \mathcal{E}, \tag{6.10}$$

where $cat(E)$ is the set of all reactions i catalyzed by enzyme E and $k_{cat}^{i,E}$ is the corresponding turnover rate. Note that this constraint also holds for protein translation and the ribosome, which is considered to be a special enzyme in our framework (see Section 5.2).

6.3.2 Regulatory logical control constraints

Extending the existing approaches for dynamic metabolic resource allocation such as deFBA, we now add two types of regulatory constraints. The first one describes the control of the discrete state transitions by the continuous variables, which corresponds to the triggering of the discrete jumps in the hybrid system. The second one is the control of the evolution of the continuous variables depending on the discrete state. Taken together, these two types of regulatory constraints specify the interplay between cellular regulation and metabolism.

Control of discrete state jumps

The key to the discrete dynamics of a hybrid system is how discrete state transitions are triggered by the continuous variables. According to Section 6.2, the expression state $\overline{p}(t)$ of all regulated proteins p is determined by a logical function $f_p : \mathbb{R}^n \to \{0, 1\}$ depending on the amounts of regulatory proteins, extracellular metabolites and reaction fluxes, i.e.,

$$\overline{p}(t) = f_p(\mathbf{RP}(t), \mathbf{Y}(t), \mathbf{v}_{\mathcal{Y}}(t), \mathbf{v}_{\mathcal{X}}(t)), \tag{6.3}$$

for all $p \in \mathcal{RP} \cup \mathcal{RE}$.

Control of the continuous dynamics by the discrete states

While the regulatory constraints in Eq. 6.3 describe how the discrete states depend on the continuous variables, we also have to specify how the continuous dynamics depends on the discrete state.

For a regulated protein p, the value of $\overline{p}$ determines whether protein p is expressed or not. Therefore, if $\overline{p}(t) = 1$, we impose as constraint that the production flux $\mathbf{v}_p(t)$ should be at least ε_p, while we require $\mathbf{v}_p(t)$ to be zero in the case $\overline{p}(t) = 0$. More formally, we obtain for all $p \in \mathcal{RP} \cup \mathcal{RE}$ the implications

$$\overline{p}(t) = 1 \quad \Rightarrow \quad \mathbf{v}_p(t) \geq \varepsilon_p, \tag{6.11}$$

$$\overline{p}(t) = 0 \quad \Rightarrow \quad \mathbf{v}_p(t) = 0, \tag{6.12}$$

where $\varepsilon_p > 0$ is a lower bound for the production rate $\mathbf{v}_p(t)$ of p in state `on`. We could also allow $\varepsilon_p = 0$ if we want to relax the model and determine the values of $\mathbf{v}_p(t)$ by optimization.

Note that the values of the parameters ε_p significantly influence the dynamics of the system. Since the lower bounds constrain the production rates, they directly affect the abundances of the regulatory proteins and the regulated enzymes. Conversely, p is strictly constrained to be degraded whenever $\overline{p}(t) =$ `off`.

6.3.3 Formulating r-deFBA as a dynamic optimization problem

Up to now, we have specified the metabolic constraints defining a dynamic solution space for cellular metabolism. In addition, we introduced the regulatory constraints to incorporate the dynamic interplay between gene regulation and metabolism. In order to predict how the cell can achieve optimal growth under these constraints, we formulate r-deFBA as a dynamic optimization problem, see Eq. 6.13. The objective is to compute time courses $\mathbf{v}(t), \mathbf{P}(t), \mathbf{Y}(t), \overline{p}(t)$ that maximize the total

biomass production over a given time interval $[t_0, t_f]$.

$$
\begin{aligned}
\max_{\mathbf{v}(t),\mathbf{P}(t),\mathbf{Y}(t),\overline{p}(t)} \quad & \int_{t_0}^{t_f} \mathbf{b}_{\mathcal{P}}^T \cdot \mathbf{P}(t)dt \\
s.t. \quad & \dot{\mathbf{Y}}(t) = S_{\mathcal{Y},\mathcal{R}_{\mathcal{Y}}} \cdot \mathbf{v}_{\mathcal{Y}}(t), \\
& \dot{\mathbf{P}}(t) = S_{\mathcal{P},\mathcal{R}_{\mathcal{P}}} \cdot \mathbf{v}_{\mathcal{P}}(t) - \mathbf{kd}_{\mathcal{P}} \circ \mathbf{P}(t), \\
& S_{\mathcal{X},\mathcal{R}_{\mathcal{Y}}} \cdot \mathbf{v}_{\mathcal{Y}}(t) + S_{\mathcal{X},\mathcal{R}_{\mathcal{X}}} \cdot \mathbf{v}_{\mathcal{X}}(t) + S_{\mathcal{X},\mathcal{R}_{\mathcal{P}}} \cdot \mathbf{v}_{\mathcal{P}}(t) = 0, \\
& \mathbf{b}_{\mathcal{Q}}^T \cdot \mathbf{Q}(t) \geq \Phi_{\mathcal{Q}} \cdot \mathbf{b}_{\mathcal{P}}^T \cdot \mathbf{P}(t), \\
& \sum_{i \in cat(E)} \frac{|\mathbf{v}_i(t)|}{k_{cat}^{i,E}} \leq E(t), \text{ for all } E \in \mathcal{E} \\
& \overline{p}(t) = f_p(\mathbf{RP}(t), \mathbf{Y}(t), \mathbf{v}_{\mathcal{Y}}(t), \mathbf{v}_{\mathcal{X}}(t)), \text{ for all } p \in \mathcal{RP} \cup \mathcal{RE} \\
& (\overline{p}(t) = 1) \Rightarrow (\mathbf{v}_p(t) \geq \varepsilon_p), \text{ for all } p \in \mathcal{RP} \cup \mathcal{RE} \\
& (\overline{p}(t) = 0) \Rightarrow (\mathbf{v}_p(t) = 0), \text{ for all } p \in \mathcal{RP} \cup \mathcal{RE} \\
& \mathbf{lb} \leq \mathbf{v}(t) \leq \mathbf{ub}, \\
& \mathbf{Y}(t_0) = \mathbf{Y}_0, \ \overline{p}(t_0) = \overline{p}_0, \text{ for all } p \in \mathcal{RP} \cup \mathcal{RE}, \\
& \mathbf{P}(t), \mathbf{Y}(t), \mathbf{v}_{\mathcal{P}}(t) \geq 0, \\
& \overline{p}(t) \in \{0, 1\}, \text{ for all } p \in \mathcal{RE} \cup \mathcal{RP}.
\end{aligned}
\tag{6.13}
$$

By $\mathbf{Y}_0$ and $\overline{p}_0$ we denote the initial values of $\mathbf{Y}(t)$ and $\overline{p}(t)$ at time $t = t_0$. In Eq. 6.13, the initial amounts $\mathbf{P}(t_0)$ are variables whose values are determined by the dynamic optimization. Alternatively, initial values for $\mathbf{P}(t_0)$ could be precomputed using RBA.

Involving discrete and continuous variables, the r-deFBA problem in Eq. 6.13 can be reformulated as a mixed-integer linear optimization problem (MILP), for which exist efficient solvers. To solve r-deFBA numerically, the dynamic real and Boolean variables are discretized in time like in (Reimers et al., 2017a; Reimers, 2017). The Boolean equations Eq. 6.3 and the logical implications Eq. 6.11-Eq. 6.12 can be transformed into a system of linear 0-1 inequalities using a standard recursive substitution procedure (Shlomi et al., 2007; Jensen et al., 2011).

6.4 Numerically solving r-deFBA as a MILP

6.4.1 Transforming logical functions into linear inequalities

To solve the dynamic optimization problem in Eq. 6.13, the logical functions representing regulatory logical control constraints in Eq. 6.3, Eq. 6.11 and Eq. 6.12 can be converted into linear 0-1 inequalities (Duffin et al., 1956). However, extra Boolean variables must be added for the conversion, which are called indicator variables. Note that the discrete states in the hybrid automata of MRNs are not affected by these indicator variables.

Converting the control of discrete state jumps into linear inequalities

We introduce a vector of indicator variables $\bar{\mathbf{I}}_{\mathcal{RP}} \in \{0,1\}^{\mathcal{RP}}$ (resp. $\bar{\mathbf{I}}_{\mathcal{Y}} \in \{0,1\}^{\mathcal{Y}}$) to indicate whether or not the regulatory protein amounts **RP** (resp. the extracellular metabolite amounts **Y**) are above the associated thresholds $\theta_{\mathcal{RP}}$ (resp. $\theta_{\mathcal{Y}}$). Mathematically

$$rp(t) \geq \theta_{rp} \quad \Leftrightarrow \quad \bar{\mathbf{I}}_{rp}(t) = 1, \text{ for all } rp \in \mathcal{RP} \tag{6.14}$$

$$y(t) \geq \theta_{y} \quad \Leftrightarrow \quad \bar{\mathbf{I}}_{y}(t) = 1, \text{ for all } y \in \mathcal{Y}. \tag{6.15}$$

Eq. (6.14)–(6.15) can be transformed into linear inequalities by the reformulation

$$x \geq \theta \quad \Leftrightarrow \quad \overline{x} = 1 \quad \rightsquigarrow \quad l \cdot (1 - \overline{x}) \leq x - \theta \leq (u + \epsilon) \cdot \overline{x} - \epsilon, \tag{6.16}$$

where x is a real variable, $\overline{x} \in \{0,1\}$ is the corresponding indicator variable, θ is the threshold, l resp. u is a lower resp. upper bound for $x - \theta$, and ϵ is a small positive number.

By $\bar{\mathbf{I}}_{\mathbf{v}_{\mathcal{Y}}} \in \{0,1\}^{\mathcal{R}_{\mathcal{Y}}}$ and $\bar{\mathbf{I}}_{\mathbf{v}_{\mathcal{X}}} \in \{0,1\}^{\mathcal{R}_{\mathcal{X}}}$ we describe the activity states of the reaction fluxes $\mathbf{v}_{\mathcal{Y}}$ and $\mathbf{v}_{\mathcal{X}}$. If a reaction flux is not zero, it is assumed to be active, i.e.,

$$\mathbf{v}_r(t) \neq 0 \quad \Leftrightarrow \quad \bar{\mathbf{I}}_{\mathbf{v}_r}(t) = 1, \text{ for all } r \in \mathcal{R}_{\mathcal{X}} \cup \mathcal{R}_{\mathcal{Y}}. \tag{6.17}$$

We transform these logical relations into linear inequalities by the reformulation

$$v \neq 0 \quad \Leftrightarrow \quad \overline{I}_v = 1 \quad \rightsquigarrow \quad \begin{cases} l \cdot \overline{I}_v \leq v \leq u \cdot \overline{I}_v, \\ (l - \epsilon)(1 - \bar{I}_v^2) + \epsilon \leq v \leq (u + \epsilon)(1 - \bar{I}_v^1) - \epsilon, \\ \overline{I}_v = \bar{I}_v^1 + \bar{I}_v^2, \end{cases} \tag{6.18}$$

where v is a real variable, $\overline{I}_v \in \{0,1\}$ is the corresponding indicator variable, l resp. u is a lower resp. upper bound for v, ϵ is a small positive number, and $\bar{I}_v^1$, $\bar{I}_v^2 \in \{0,1\}$ are two auxiliary 0-1 variables.

After introducing the indicator variables $\bar{\mathbf{I}}_{\mathcal{RP}}, \bar{\mathbf{I}}_{\mathcal{Y}}, \bar{\mathbf{I}}_{\mathbf{v}_{\mathcal{Y}}}$ and $\bar{\mathbf{I}}_{\mathbf{v}_{\mathcal{X}}}$, the regulatory constraints (6.3) are converted into Boolean equations

$$\overline{p}(t) = \overline{f}_p(\bar{\mathbf{I}}_{\mathcal{RP}}(t), \bar{\mathbf{I}}_{\mathcal{Y}}(t), \bar{\mathbf{I}}_{\mathbf{v}_{\mathcal{Y}}}(t), \bar{\mathbf{I}}_{\mathbf{v}_{\mathcal{X}}}(t)), \qquad \text{for all } p \in \mathcal{RE} \cup \mathcal{RP}. \tag{6.19}$$

Similar to the original function f_p in (6.3), the Boolean function $\overline{f}_p$ is defined in terms of the indicator variables $\bar{\mathbf{I}}_{\mathcal{RP}}, \bar{\mathbf{I}}_{\mathcal{Y}}, \bar{\mathbf{I}}_{\mathbf{v}_{\mathcal{X}}}, \bar{\mathbf{I}}_{\mathbf{v}_{\mathcal{Y}}}$ and the operations $\neg$ (not), $\wedge$ (and), $\vee$ (or), such that $\overline{f}_p(\bar{\mathbf{I}}_{\mathcal{RP}}(t), \bar{\mathbf{I}}_{\mathcal{Y}}(t), \bar{\mathbf{I}}_{\mathbf{v}_{\mathcal{Y}}}(t), \bar{\mathbf{I}}_{\mathbf{v}_{\mathcal{X}}}(t)) = f_p(\mathbf{RP}(t), \mathbf{Y}(t), \mathbf{v}_{\mathcal{X}}(t), \mathbf{v}_{\mathcal{Y}}(t))$.

Eq. (6.19) can be transformed into a set of linear inequalities by recursively applying the rules

$$y = \neg x_1 \quad \rightsquigarrow \quad y = 1 - x_1 \tag{6.20}$$

$$y = x_1 \wedge x_2 \quad \rightsquigarrow \quad y \leq x_1, \;\; y \leq x_2, \;\; x_1 + x_2 \leq 1 + y \tag{6.21}$$

$$y = x_1 \vee x_2 \quad \rightsquigarrow \quad x_1 \leq y, \;\; x_2 \leq y, \;\; y \leq x_1 + x_2 \tag{6.22}$$

for variables $x_1, x_2, y \in \{0,1\}$ and by introducing additional 0-1 variables for the intermediate results.

Converting the control of the continuous dynamics into linear inequalities

For the logical control of the continuous dynamics, Eq. 6.11 and Eq. 6.12 are converted into linear inequalities by the reformulation

$$\left.\begin{array}{lll} \overline{p}(t) = 1 & \Rightarrow & \mathbf{v}_p(t) \geq \varepsilon_p \\ \overline{p}(t) = 0 & \Rightarrow & \mathbf{v}_p(t) = 0 \end{array}\right\} \quad \rightsquigarrow \quad \varepsilon_p \cdot \overline{p}(t) \leq \mathbf{v}_p(t) \leq u_p \cdot \overline{p}(t), \tag{6.23}$$

where $p \in \mathcal{RP} \cup \mathcal{RE}$ is a regulated protein, $\varepsilon_p > 0$ is lower bound on the flux $\mathbf{v}_p(t)$ if p is expressed, and u_p is an upper bound on $\mathbf{v}_p(t)$.

Using these transformations, the regulatory constraints can be reformulated as mixed 0-1 linear inequalities, see also (Shlomi et al., 2007; Jensen et al., 2011). Together with the metabolic constraints, the dynamic r-deFBA optimization problem can be solved as a MILP by discretizing the continuous and Boolean variables in time.

6.4.2 Discretizing the variables in time to solve r-deFBA

To solve numerically the dynamic optimization problem of r-deFBA, we discretize the variables in time, using the midpoint rule like in (Reimers et al., 2017a). The macromolecular amounts $\mathbf{P}$ and extracellular metabolite amounts $\mathbf{Y}$ are discretized at each time point $t_k, k \in \{0, \ldots, n\}$. The flux variables $\mathbf{v}$ and the derivatives $\dot{\mathbf{P}}, \dot{\mathbf{Y}}$ are discretized at the midpoint $\frac{t_k + t_{k-1}}{2}, k \in \{1, \ldots, n\}$. In order to control the protein production, whose fluxes $\mathbf{v}_p$ are discretized at midpoint $\frac{t_k + t_{k-1}}{2}$, the Boolean variables $\overline{p}, p \in \mathcal{RP} \cup \mathcal{RE}$, are discretized at $\frac{t_k + t_{k-1}}{2}$, too. Indicator variables $\bar{\mathbf{I}}_{\mathbf{v}_\mathcal{Y}}$ and $\bar{\mathbf{I}}_{\mathbf{v}_\mathcal{X}}$ are also considered at the midpoint, which is consistent with the reaction fluxes $\mathbf{v}_\mathcal{Y}$ and $\mathbf{v}_\mathcal{X}$. For $\bar{\mathbf{I}}_\mathcal{Y}$ and $\bar{\mathbf{I}}_{\mathcal{RP}}$, we calculate the corresponding amounts $\mathbf{Y}$ and $\mathbf{RP}$ at the time points t_k and t_{k-1} to determine the indicator values by

$$\frac{rp(t_k) + rp(t_{k-1})}{2} \geq \theta_{rp} \quad \Leftrightarrow \quad \bar{\mathbf{I}}_{rp}\left(\frac{t_k + t_{k-1}}{2}\right) = 1, \text{ for all } rp \in \mathcal{RP}, \tag{6.24}$$

$$\frac{y(t_k) + y(t_{k-1})}{2} \geq \theta_y \quad \Leftrightarrow \quad \bar{\mathbf{I}}_{y}\left(\frac{t_k + t_{k-1}}{2}\right) = 1, \text{ for all } y \in \mathcal{Y}. \tag{6.25}$$

The resulting mixed-integer linear program (MILP) is given in Figure 6.2. As we can see, all macromolecule amounts $\mathbf{P}$, the derivatives $\dot{\mathbf{P}}$ and all reaction fluxes $\mathbf{v}$ at each time point are continuous variables. The indicator variables of the extracellular species $\bar{\mathbf{I}}_\mathcal{Y}$, regulatory proteins $\bar{\mathbf{I}}_{\mathcal{RP}}$, reaction fluxes $\bar{\mathbf{I}}_{\mathbf{v}_\mathcal{Y}}$, $\bar{\mathbf{I}}_{\mathbf{v}_\mathcal{X}}$ and expression states of regulated proteins $\overline{p}$ involved in the regulation are 0-1 variables. The biomass integral over the simulation period $[t_0, t_f]$ is defined as the sum of all the macromolecule masses at each time point. Using this MILP, values for all the variables can be predicted using efficient MILP solvers such as Gurobi (`http://www.gurobi.com`), Cplex (`https://www.ibm.com/products/ilog-cplex-optimization-studio`) or Scip (`https://scip.zib.de/`).

While the reformulation of r-deFBA as an MILP problem is our current solution strategy, developing possible alternative solution approaches is a topic of further research.

$$
\begin{aligned}
\max\ & \sum_{k=0}^{n} \mathbf{b}_{\mathcal{P}}^{T} \cdot \mathbf{P}\,(t_k)\,, \\
s.t.\ & \dot{\mathbf{Y}}\left(\frac{t_k + t_{k-1}}{2}\right) = S_{\mathcal{Y},\mathcal{R}_{\mathcal{Y}}} \cdot \mathbf{v}_{\mathcal{Y}}\left(\frac{t_k + t_{k-1}}{2}\right), \\
& \dot{\mathbf{P}}\left(\frac{t_k + t_{k-1}}{2}\right) = S_{\mathcal{P},\mathcal{R}_{\mathcal{P}}} \cdot \mathbf{v}_{\mathcal{P}}\left(\frac{t_k + t_{k-1}}{2}\right) - \mathbf{kd}_{\mathcal{P}} \circ \frac{\mathbf{P}(t_k) + \mathbf{P}(t_{k-1})}{2}, \\
& S_{\mathcal{X},\mathcal{R}_{\mathcal{Y}}} \cdot \mathbf{v}_{\mathcal{Y}}\left(\frac{t_k + t_{k-1}}{2}\right) + S_{\mathcal{X},\mathcal{R}_{\mathcal{X}}} \cdot \mathbf{v}_{\mathcal{X}}\left(\frac{t_k + t_{k-1}}{2}\right) + S_{\mathcal{X},\mathcal{R}_{\mathcal{P}}} \cdot \mathbf{v}_{\mathcal{P}}\left(\frac{t_k + t_{k-1}}{2}\right) = 0, \\
& \sum_{i \in cat(E)} \frac{|\mathbf{v}_i\left(\frac{t_k + t_{k-1}}{2}\right)|}{k_{cat}^{i,E}} \leq \frac{E(t_k) + E(t_{k-1})}{2}, \text{ for all } E \in \mathcal{E}, \\
& \mathbf{b}_{\mathcal{Q}}^{T} \cdot \mathbf{Q}(t_k) \geq \Phi_{\mathcal{Q}} \cdot \mathbf{b}_{\mathcal{P}}^{T} \cdot \mathbf{P}(t_k), \\
& \mathbf{P}(t_k) = \mathbf{P}(t_{k-1}) + (t_k - t_{k-1}) \cdot \dot{\mathbf{P}}\left(\frac{t_k + t_{k-1}}{2}\right), \\
& \mathbf{Y}(t_k) = \mathbf{Y}(t_{k-1}) + (t_k - t_{k-1}) \cdot \dot{\mathbf{Y}}\left(\frac{t_k + t_{k-1}}{2}\right), \\
& \mathbf{v}_r\left(\frac{t_k + t_{k-1}}{2}\right) \neq 0 \ \Leftrightarrow\ \bar{\mathbf{I}}_{\mathbf{v}_r}\left(\frac{t_k + t_{k-1}}{2}\right) = 1, \text{ for all } r \in \mathcal{R}_{\mathcal{X}} \cup \mathcal{R}_{\mathcal{Y}}, \\
& \frac{rp(t_k) + rp(t_{k-1})}{2} \geq \theta_{rp} \ \Leftrightarrow\ \bar{\mathbf{I}}_{rp}\left(\frac{t_k + t_{k-1}}{2}\right) = 1, \text{ for all } rp \in \mathcal{RP}, \\
& \frac{y(t_k) + y(t_{k-1})}{2} \geq \theta_y \ \Leftrightarrow\ \bar{\mathbf{I}}_{y}\left(\frac{t_k + t_{k-1}}{2}\right) = 1, \text{ for all } y \in \mathcal{Y}, \\
& \overline{p}\left(\frac{t_k + t_{k-1}}{2}\right) = \overline{f}_p\left(\bar{\mathbf{I}}_{\mathcal{RP}}\left(\frac{t_k + t_{k-1}}{2}\right), \bar{\mathbf{I}}_{\mathcal{Y}}\left(\frac{t_k + t_{k-1}}{2}\right), \bar{\mathbf{I}}_{\mathbf{v}_{\mathcal{Y}}}\left(\frac{t_k + t_{k-1}}{2}\right), \bar{\mathbf{I}}_{\mathbf{v}_{\mathcal{X}}}\left(\frac{t_k + t_{k-1}}{2}\right)\right), \\
& \qquad \text{for all } p \in \mathcal{RP} \cup \mathcal{RE}, \\
& \overline{p}\left(\frac{t_k + t_{k-1}}{2}\right) = 1 \ \Rightarrow\ \mathbf{v}_p(\frac{t_k + t_{k-1}}{2}) \geq \varepsilon_p, \text{ for all } p \in \mathcal{RP} \cup \mathcal{RE}, \\
& \overline{p}\left(\frac{t_k + t_{k-1}}{2}\right) = 0 \ \Rightarrow\ \mathbf{v}_p(\frac{t_k + t_{k-1}}{2}) = 0, \text{ for all } p \in \mathcal{RP} \cup \mathcal{RE}, \\
& \mathbf{P}(t_k), \mathbf{Y}(t_k) \geq 0, \quad \mathbf{v}_{min} \leq \mathbf{v}\left(\frac{t_k + t_{k-1}}{2}\right) \leq \mathbf{v}_{max}, \\
& \bar{\mathbf{I}}_{\mathbf{v}_{\mathcal{X}}}\left(\frac{t_k + t_{k-1}}{2}\right), \bar{\mathbf{I}}_{\mathbf{v}_{\mathcal{Y}}}\left(\frac{t_k + t_{k-1}}{2}\right), \bar{\mathbf{I}}_{\mathcal{RP}}\left(\frac{t_k + t_{k-1}}{2}\right), \bar{\mathbf{I}}_{\mathcal{Y}}\left(\frac{t_k + t_{k-1}}{2}\right) \in \{0,1\}, \\
& \overline{p}\left(\frac{t_k + t_{k-1}}{2}\right) \in \{0,1\}, \text{ for all } p \in \mathcal{RP} \cup \mathcal{RE}, \quad \text{for all } k = 1, \ldots, n, \\
& \mathbf{Y}(t_0) = \mathbf{Y}_0, \quad \overline{p}(t_0) = \overline{p}_0, \text{ for all } p \in \mathcal{RP} \cup \mathcal{RE}.
\end{aligned}
$$

Figure 6.2: MILP for solving r-deFBA model

6.5 Biological Application 1 on CCR model

The carbon catabolite repression (CCR) model and the corresponding hybrid automaton have been described in Section 5.5. The parameters are listed in Table 5.1. To illustrate r-deFBA, we construct in Section 6.5.1 an r-deFBA model for this network. In Section 6.5.2 we compare the resulting dynamics for r-deFBA to standard deFBA and to the hybrid automata framework considered in Chapter 5.

The general workflow for building an r-deFBA model is illustrated in Figure 6.3. Starting from a metabolic and a transcriptional regulatory network, we first construct a metabolic-regulatory network (MRN), as presented in Section 6.2.

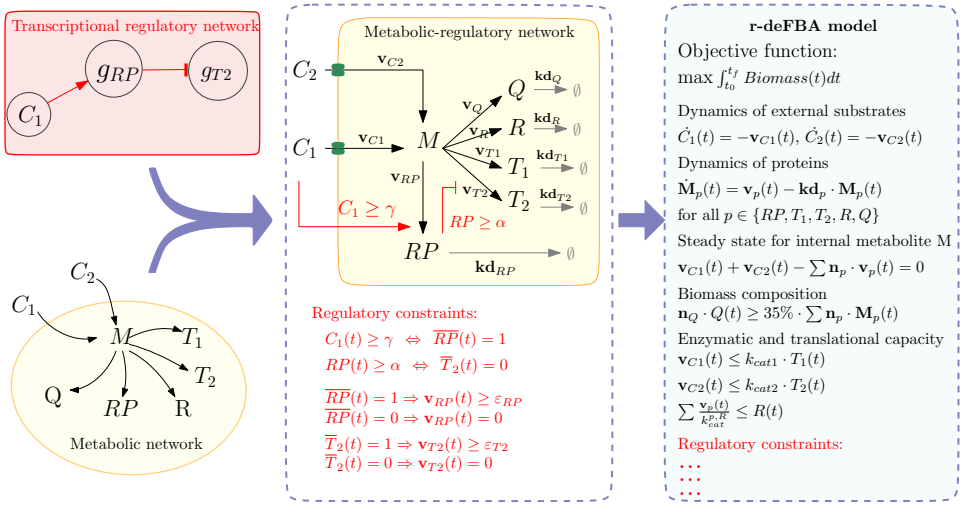


Figure 6.3: Workflow to build an r-deFBA model for the CCR model with two regulatory rules. The Boolean variables $\overline{RP}, \overline{T}_2$ describe the expression state of the genes g_{RP}, g_{T_2}, which determines the production of proteins RP, T_2. The thresholds $\varepsilon_{RP}, \varepsilon_{T_2}$ define the minimal expression levels for the regulated proteins RP, T_2 to be in state on

6.5.1 r-deFBA and deFBA model of CCR

In the metabolic network of Figure 6.3, we have two carbon sources $\mathcal{Y} = \{C_1, C_2\}$, which are converted into precursor molecules $\mathcal{X} = \{M\}$. For simplicity, we assume only two uptake reactions $C_1 \to M, C_2 \to M$, catalyzed by enzymes T_1 resp. T_2. The precursor molecules M are used to synthesize five types of macromolecules $\mathcal{P} = \{Q, R, T_1, T_2, RP\}$, which are the enzymes T_1, T_2, regulatory proteins RP, housekeeping proteins Q, and ribosomes R. The stoichiometry of the synthesis reactions and corresponding parameter values are given in Table 6.1. The total biomass $Biomass(t)$ is defined as the sum of the molecular masses

$$Biomass(t) = w \cdot M(t) + \sum_{p \in \mathcal{P}} w \cdot \mathbf{n}_p \cdot \mathbf{M}_p(t), \tag{6.26}$$

where w corresponds to the molar weight of one precursor molecule M and $\mathbf{n}_p$ is the number of precursor molecules needed to build one macromolecule p. By $M(t)$ and $\mathbf{M}_p(t)$ we denote again the molar amounts [mmol] of M resp. $p \in \mathcal{P}$ at time t.

In the regulatory network of Figure 6.3, g_{RP} and g_{T_2} denote the two genes encoding the regulated proteins RP and T_2. We assume that g_{RP} is activated by the presence of C_1 and that g_{T_2} is inhibited by g_{RP}. This leads to two regulatory rules

$$\overline{RP}(t) = 1 \quad \Leftrightarrow \quad C_1(t) \geq \gamma \quad \text{and} \quad \overline{T}_2(t) = 0 \quad \Leftrightarrow \quad RP(t) \geq \alpha, \tag{6.27}$$

with thresholds $\alpha, \gamma > 0$. The expression states $\overline{RP}, \overline{T}_2$ are linked to the flux variables v_{RP}, v_{T2} by the implications

$$\begin{array}{llll} \overline{RP}(t) = 1 & \Rightarrow & \mathbf{v}_{RP}(t) \geq \varepsilon_{RP}, & \overline{T}_2(t) = 1 \quad \Rightarrow \quad \mathbf{v}_{T2}(t) \geq \varepsilon_{T2}, \\ \overline{RP}(t) = 0 & \Rightarrow & \mathbf{v}_{RP}(t) = 0, & \overline{T}_2(t) = 0 \quad \Rightarrow \quad \mathbf{v}_{T2}(t) = 0, \end{array} \tag{6.28}$$

using thresholds $\varepsilon_{RP}, \varepsilon_{T2} > 0$.

The full r-deFBA model of the CCR network in Figure 6.3 reads in Eq. 6.29.

$$\begin{aligned} \max_{\mathbf{v}(t), C_1(t), C_2(t), \mathbf{M}(t)} \quad & \int_{t_0}^{t_f} \Big(\sum_{p \in \mathcal{P}} w \cdot \mathbf{n}_p \cdot \mathbf{M}_p(t) + w \cdot M(t_0) \Big)\, dt \\ s.t. \quad & \dot{C}_1(t) = -\mathbf{v}_{C1}(t), \ \dot{C}_2(t) = -\mathbf{v}_{C2}(t), \\ & \mathbf{M}_p(t) = \mathbf{v}_p(t) - \mathbf{kd}_p \cdot \mathbf{M}_p(t), \quad \text{for all } p \in \mathcal{P} \\ & \mathbf{v}_{C1}(t) + \mathbf{v}_{C2}(t) - \sum_{p \in \mathcal{P}} \mathbf{n}_p \cdot \mathbf{v}_p(t) = 0, \\ & \mathbf{n}_Q \cdot Q(t) \geq \Phi_Q \cdot \sum_{p \in \mathcal{P}} \mathbf{n}_p \cdot \mathbf{M}_p(t), \\ & \mathbf{v}_{C1}(t) \leq k_{cat1} \cdot T_1(t), \ \mathbf{v}_{C2}(t) \leq k_{cat2} \cdot T_2(t), \\ & \sum_{p \in \mathcal{P}} \frac{\mathbf{v}_p(t)}{k_{cat}^{p,R}} \leq R(t), \\ & C_1(t) \geq \gamma \Leftrightarrow \overline{RP}(t) = 1, \ RP(t) \geq \alpha \Leftrightarrow \overline{T}_2(t) = 0, \\ & \overline{RP}(t) = 1 \Rightarrow \mathbf{v}_{RP}(t) \geq \varepsilon_{RP}, \ \overline{RP}(t) = 0 \Rightarrow \mathbf{v}_{RP}(t) = 0, \\ & \overline{T}_2(t) = 1 \Rightarrow \mathbf{v}_{T2}(t) \geq \varepsilon_{T2}, \ \overline{T}_2(t) = 0 \Rightarrow \mathbf{v}_{T2}(t) = 0, \\ & (C_1, C_2, RP, T_1, T_2, R, Q)(t) \geq 0, \ \overline{RP}(t), \overline{T}_2(t) \in \{0, 1\}, \\ & (C_1, C_2, M, RP, T_1, T_2, R, Q)(t_0) = \\ & \qquad (1000, 500, 20, 0, 0.001, 0.001, 0.01, 0.15), \\ & (\overline{RP}, \overline{T}_2)(t_0) = (1, 1). \end{aligned} \tag{6.29}$$

The corresponding deFBA model is obtained by omitting the regulatory constraints. For the computations, we used the parameter values given in Table 6.1, Table 6.2, and Table 5.1. The r-deFBA and deFBA program detailed Eq. 6.29 can be solved by using optimization solvers such as Gurobi (http://www.gurobi.com).

Metabolic reaction	Flux	Enzyme	Turnover rate [1/min]	
$C_1 \to M$	$\mathbf{v}_{C1}$	T_1	$k_{cat1} = 3000$	
$C_2 \to M$	$\mathbf{v}_{C2}$	T_2	$k_{cat2} = 2000$	
Biomass reaction	Flux	Enzyme	Turnover rate	Degradation rate
$\mathbf{n}_p M \to p$	$\mathbf{v}_p$		$k_{cat}^{p,R} = k_r/\mathbf{n}_p$ [1/min]	$\mathbf{kd}_p$ [1/min]
$300M \to Q$	$\mathbf{v}_Q$	R	4.2	0.01
$7459M \to R$	$\mathbf{v}_R$	R	0.1689	0.01
$400M \to T_1$	$\mathbf{v}_{T1}$	R	3.15	0.01
$1500M \to T_2$	$\mathbf{v}_{T2}$	R	0.84	0.01
$300M \to RP$	$\mathbf{v}_{RP}$	R	4.2	0.2

Table 6.1: Metabolic and biomass reactions with corresponding parameters.

t_0	t_f	Φ_Q	α	γ	ε_{RP}	ε_{T2}	k_r
0	55	0.35	0.03	20	0.01	0.01	1260
min	min		mmol	mmol	mmol/min	mmol/min	1/min

Table 6.2: Additional parameters. Here, k_r denotes the elongation rate.

6.5.2 Comparing r-deFBA, deFBA, and the hybrid automaton

Next, we compare the dynamics of the CCR model obtained by r-deFBA, deFBA and the hybrid automaton, see Figure 6.4. The dynamics and discrete states of hybrid automaton H_{diaux} are shown in Figure 5.5, see the details in Section 5.5.2. In all three simulations, we use the same parameter values given in Table 5.1, Table 6.1 and Table 6.2, and the initial conditions from Eq. 6.29.

The diauxic shift is predicted successfully by all three approaches. However, the underlying principles are different. By maximizing the biomass production while taking into account only the metabolic constraints, deFBA shows that the diauxic shift is an optimal metabolic behavior. In contrast, r-deFBA computes an optimal trajectory for biomass production, taking into account both the metabolic and the regulatory constraints. Due to the additional regulatory constraints, r-deFBA produces less biomass than deFBA and needs more time to consume the available carbon resources, see Figure 6.4D. The continuous metabolic variables of the hybrid automaton evolve according to the Michaelis-Menten kinetics of equations in Figure 5.5. These kinetics depend on the current discrete state, which in turn is determined by the regulatory control, i.e., the jump conditions. As an optimal control strategy for the hybrid system representing the MRN, r-deFBA clearly gains more

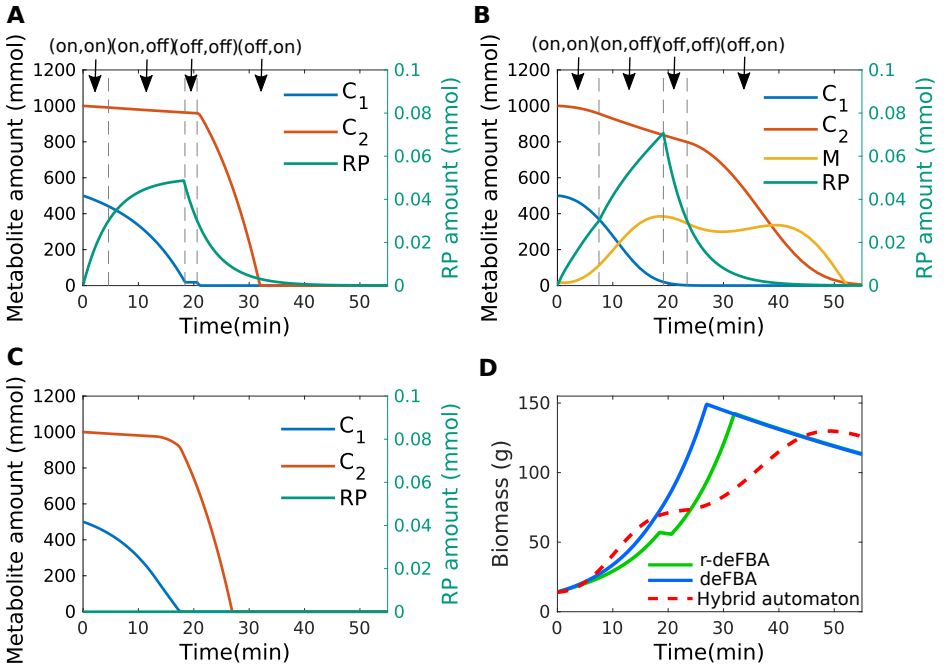


Figure 6.4: Time courses of C_1, C_2 (left axis),RP (right axis) predicted by r-deFBA (**A**), the hybrid automaton (**B**), deFBA (**C**) and corresponding biomass production (**D**). In all three simulations, the same parameter and initial values were used. For r-deFBA and the hybrid automaton, we also indicate the discrete states $(\overline{RP}, \overline{T}_2)$ with the transitions marked by vertical dashed lines. Due to the steady-state assumption, the molar amount of the precursor M remains constant in deFBA and r-deFBA.

biomass than the hybrid automaton, but less than deFBA, which does not include regulation.

Both r-deFBA and the hybrid automaton successfully predict the discrete state transitions during diauxie. In Figure 6.4A and Figure 6.4B, the time profiles are divided into three growth phases, corresponding to the discrete state transitions. The transitions of r-deFBA are consistent with those obtained by the hybrid automaton. In the first growth phase (a), expression of RP is activated and $\overline{RP} = \texttt{on}$ because initially $C_1 \geq \gamma$. We also have $\overline{T}_2 = \texttt{on}$ because RP is initialized by 0. Thus, the initial state of the network is $(\overline{RP}, \overline{T}_2) = (\texttt{on}, \texttt{on})$. As time goes on, C_1 is consumed while RP is synthesized and accumulated. When RP reaches the threshold α, the synthesis of T_2 is inhibited and the model jumps to the state $(\texttt{on}, \texttt{off})$. Next, when $C_1 < \gamma$, the discrete state changes to $(\texttt{off}, \texttt{off})$, which represents the lag phase during diauxie. In this phase, enzyme T_2 is still repressed until RP falls below its threshold α. Once this happens, the system switches to the final state $(\texttt{off}, \texttt{on})$, where $RP < \alpha$ and T_2 is produced to metabolize C_2. Overall, the interactions between discrete regulation and continuous metabolism are correctly incorporated in our r-deFBA. In deFBA, the regulatory protein RP remains at the initial value 0, see Figure 6.4C. From the optimization perspective, there is no benefit in produc-

ing *RP* because it is a non-catalytic protein and does not sufficiently contribute to biomass.

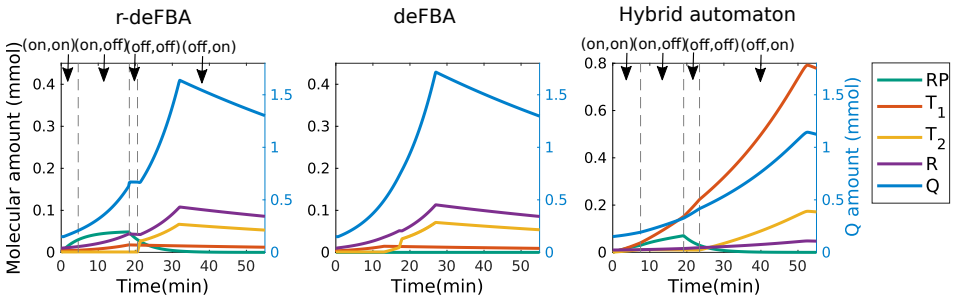


Figure 6.5: Time courses of RP, T_1, T_2, R (left axis) and Q (right axis) for r-deFBA, deFBA, and the hybrid automaton.

Another interesting point is the production of macromolecules, see Figure 6.5. In deFBA and r-deFBA, the dynamic optimization indicates that the production of T_1 should be stopped once C_1 is exhausted, although there is no regulatory control for T_1. Intuitively, T_1 is not needed anymore for uptake of C_1. In order to increase biomass, it is better to produce T_2 and R. When specifying the dynamics of the hybrid automaton in Figure 5.5, we equally share the available resources between all synthesis reactions that are active in the current location. In real cells, this is unlikely to happen and not optimal for biomass production, as can be seen from Figure 6.4D. Compared with the hybrid automaton, much more ribosome is produced in r-deFBA and deFBA, leading to a much larger biomass production, see Figure 6.5. Clearly, how the cell allocates its resources to different enzymes will affect significantly the cellular growth.

Since r-deFBA is more constrained than deFBA, the maximum biomass predicted by r-deFBA will always be less than or equal to the one for deFBA. However, the two maxima can get very close if the regulatory constraints are consistent with the objective in the dynamic optimization. In our simulation, deFBA successfully predicts the diauxic shift, even without regulatory control, showing that this is an optimal strategy for biomass production. However, deFBA fails to provide information about how the cell should be regulated in order to achieve this result. In contrast, r-deFBA allows predicting both the dynamic evolution of regulatory proteins and the discrete state transitions which together enable the cell to implement an optimal growth strategy.

6.6 Biological Application 2: core carbon metabolism

6.6.1 MRN model of the core carbon network

As before, we first construct a metabolic-regulatory network (MRN), see Figure 6.6. Here we combine a metabolic and a regulatory network for core carbon

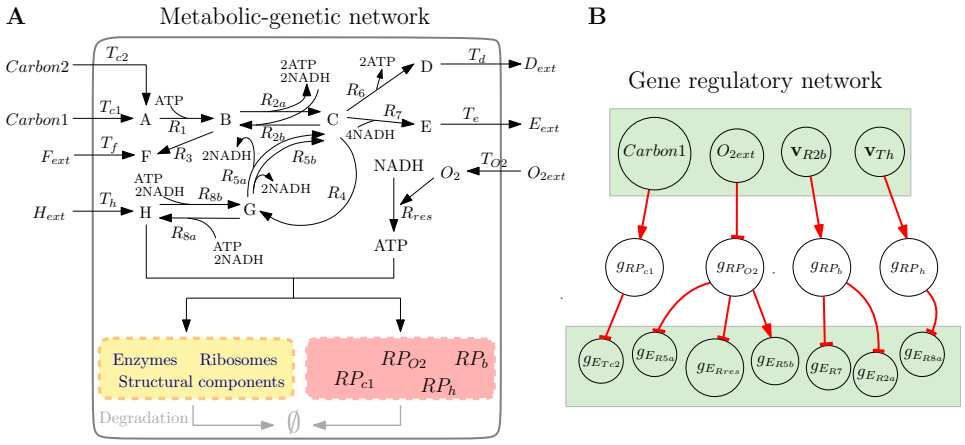


Figure 6.6: The metabolic-genetic network of core carbon metabolism and corresponding gene regulatory network.

Macromolecular synthesis reaction $\mathbf{n}_p H + \mathbf{m}_p ATP \rightarrow p$	Molar weight $\mathbf{b}_p$ [g/mmol]	Turnover rate $k_{cat}^{p,R}$ [1/min]	Regulatory rule
	Enzymes		
$400H + 1600ATP \rightarrow E_{Tc1}$	40	2.5	
$1500H + 6000ATP \rightarrow E_{Tc2}$	150	0.67	$\overline{E}_{Tc2} = \text{on} \Leftrightarrow RP_{c1} < \vartheta_{RP}$
$400H + 1600ATP \rightarrow E_{Tf}$	40	2.5	
$400H + 1600ATP \rightarrow E_{Th}$	40	2.5	
$500H + 2000ATP \rightarrow E_{R1}$	50	2.0	
$500H + 2000ATP \rightarrow E_{R2a}$	50	2.0	$\overline{E}_{R2a} = \text{on} \Leftrightarrow RP_b < \vartheta_{RP}$
$500H + 2000ATP \rightarrow E_{R2b}$	50	2.0	
$1000H + 4000ATP \rightarrow E_{R6}$	100	1.0	
$1000H + 4000ATP \rightarrow E_{R7}$	100	1.0	$\overline{E}_{R7} = \text{on} \Leftrightarrow RP_b < \vartheta_{RP}$
$2000H + 8000ATP \rightarrow E_{R3}$	200	0.5	
$500H + 2000ATP \rightarrow E_{R4}$	50	2.0	
$4000H + 16000ATP \rightarrow E_{R8a}$	400	0.25	$\overline{E}_{R8a} = \text{on} \Leftrightarrow RP_h < \vartheta_{RP}$
$4000H + 16000ATP \rightarrow E_{R8b}$	400	0.25	
$500H + 2000ATP \rightarrow E_{R5a}$	50	2.0	$\overline{E}_{R5a} = \text{on} \Leftrightarrow RP_{O2} < \vartheta_{RP}$
$500H + 2000ATP \rightarrow E_{R5b}$	50	2.0	$\overline{E}_{R5b} = \text{on} \Leftrightarrow RP_{O2} \geq \vartheta_{RP}$
$500H + 2000ATP \rightarrow E_{Rres}$	50	2.0	$\overline{E}_{Rres} = \text{on} \Leftrightarrow RP_{O2} < \vartheta_{RP}$
$500H + 2000ATP \rightarrow E_Q$	50	2.0	
$4500H + 21000ATP + 1500C \rightarrow R$	600	0.2	
	Regulatory proteins		
$300H + 1200ATP \rightarrow RP_{O2}$	30	3.33	$\overline{RP}_{O2} = \text{on} \Leftrightarrow O_{2ext} < \vartheta_Y$
$300H + 1200ATP \rightarrow RP_{c1}$	30	3.33	$\overline{RP}_{c1} = \text{on} \Leftrightarrow Carbon1 \geq \vartheta_Y$
$300H + 1200ATP \rightarrow RP_h$	30	3.33	$\overline{RP}_h = \text{on} \Leftrightarrow \mathbf{v}_{Th} \geq \vartheta_v$
$300H + 1200ATP \rightarrow RP_b$	30	3.33	$\overline{RP}_b = \text{on} \Leftrightarrow \mathbf{v}_{R2b} \geq \vartheta_v$
	Structural components		
Synthesis reaction	Molar weight $\mathbf{b}_Q$ [g/mmol]	Turnover rate k_{cat}^{Q,E_Q} [1/min]	
$250H + 1500ATP + 250C + 250F \rightarrow Q$	75	3.0	

Table 6.3: Macromolecular synthesis reactions with corresponding molar weights, turnover rates and regulatory rules

metabolism based on (Covert et al., 2001; Waldherr et al., 2015). The metabolic network in Figure 6.6A covers the major carbon pathways including glycolysis, TCA cycle, carbon storage, amino acid synthesis, pentose phosphate pathway, fermentation, and also the macromolecule synthesis.

Using the notation from Section 6.2, we have the following molecular species:

$$\begin{aligned}
\mathcal{X} &= \{A, B, C, D, E, F, G, H, NADH, ATP, O_2\},\\
\mathcal{Y} &= \{Carbon\ 1, Carbon\ 2, O_{2ext}, D_{ext}, E_{ext}, F_{ext}, H_{ext}\},\\
\mathcal{Q} &= \{Q\},\\
\mathcal{RE} &= \{E_{Tc2}, E_{R2a}, E_{R5a}, E_{R5b}, E_{R7}, E_{R8a}, E_{Rres}\},\\
\mathcal{NRE} &= \{E_{Tc1}, E_{Tf}, E_{Th}, E_{R1}, E_{R2b}, E_{R3}, E_{R4}, E_{R6}, E_{R8b}, E_Q, Q, R\},\\
\mathcal{RP} &= \{RP_{c1}, RP_{O2}, RP_b, RP_h\},\\
\mathcal{E} &= \mathcal{RE} \cup \mathcal{NRE}, \quad \mathcal{P} = \mathcal{Q} \cup \mathcal{E} \cup \mathcal{RP}.
\end{aligned} \tag{6.30}$$

The details on the different metabolic and biomass reactions are given in Table 4.2 and Table 6.3. To get reasonable flux bounds on reactions describing diffusive exchange across the plasma membrane, we define the structural component Q as the enzymatic macromolecule for these reactions, together with an appropriate rate constant for diffusion (Waldherr et al., 2015).

Regarding the regulatory network of Figure 6.6B, we identify again the state of a gene with the activity of the reaction producing the corresponding protein. For example, the gene state g_{RPc1} is identified with the activity state $\overline{RP}_{c1}$ of the reaction producing RP_{c1}. This means that the reaction synthesizing RP_{c1} will be active whenever the amount of external *Carbon* 1 exceeds a given threshold. Conversely, the reaction will be blocked if insufficient *Carbon* 1 is available, see the regulatory rule for RP_{c1} in Table 6.3. The roles of the four regulatory proteins RP_{c1}, RP_{O2}, RP_h, RP_b have been explained in Section 3.3.1. For additional details, we refer to (Covert et al., 2001).

6.6.2 r-deFBA vs deFBA model of core carbon network

The complete r-deFBA model reads in Eq. 6.31. The deFBA model of the core carbon network is the upper section of the dynamic optimization program, which merely includes the objective function and metabolic constraints while ignoring the regulatory control section. The initial values depend on the specific scenario and will be specified in the next section.

6.6.3 Comparing r-deFBA and deFBA

In total, there are 11 regulated proteins, which include 4 regulatory proteins and 7 regulated enzymes. The discrete state space thus contains 2^{11} states, which are difficult to explore by the hybrid automaton. In the following, we present two scenarios to show how r-deFBA can be used to predict the integrated dynamics

of metabolism and regulation even in a large state space. In each case, we compare r-deFBA with deFBA, which also models metabolism, but does not take into account the regulatory control.

$$
\begin{aligned}
\max_{\mathbf{v}(t),\mathbf{P}(t),\mathbf{Y}(t),\overline{p}(t)} \quad & \int_{t_0}^{t_f} \mathbf{b}_{\mathcal{P}}^T \mathbf{P}(t) dt \\
s.t. \quad & \dot{\mathbf{Y}}(t) = S_{\mathcal{Y},\mathcal{R}_\mathcal{Y}} \cdot \mathbf{v}_\mathcal{Y}(t), \\
& \dot{\mathbf{P}}(t) = S_{\mathcal{P},\mathcal{R}_\mathcal{P}} \cdot \mathbf{v}_\mathcal{P}(t) - \mathbf{kd}_\mathcal{P} \circ \mathbf{P}(t), \\
& S_{\mathcal{X},\mathcal{R}_\mathcal{Y}} \cdot \mathbf{v}_\mathcal{Y}(t) + S_{\mathcal{X},\mathcal{R}_\mathcal{X}} \cdot \mathbf{v}_\mathcal{X}(t) + S_{\mathcal{X},\mathcal{R}_\mathcal{P}} \cdot \mathbf{v}_\mathcal{P}(t) = 0, \\
& \mathbf{b}_Q^T \cdot Q(t) \geq \ \Phi_Q \cdot \mathbf{b}_\mathcal{P}^T \cdot \mathbf{P}(t), \\
& \mathbf{v}_i(t) \leq k_{cat}^{i,\mathcal{E}_j} \cdot \mathbf{E}_j(t), \text{ for all } \mathcal{E}_j \in \mathcal{E} \setminus \{Q, R\}, \\
& \sum_{i \in \{T_{O2}, T_d, T_e\}} \frac{\mathbf{v}_i(t)}{k_{cat}^{i,Q}} \leq Q(t), \quad \sum_{p \in \mathcal{P} \setminus Q} \frac{\mathbf{v}_p(t)}{k_{cat}^{p,R}} \leq R(t), \\
& \overline{E}_{Tc2} = \mathsf{on} \Leftrightarrow RP_{c1} < \vartheta_{RP}, \quad \overline{E}_{R2a} = \mathsf{on} \Leftrightarrow RP_b < \vartheta_{RP}, \\
& \overline{E}_{R7} = \mathsf{on} \Leftrightarrow RP_b < \vartheta_{RP}, \quad \overline{E}_{R8a} = \mathsf{on} \Leftrightarrow RP_h < \vartheta_{RP}, \\
& \overline{E}_{R5a} = \mathsf{on} \Leftrightarrow RP_{O2} < \vartheta_{RP}, \quad \overline{E}_{R5b} = \mathsf{on} \Leftrightarrow RP_{O2} \geq \vartheta_{RP}, \\
& \overline{E}_{Rres} = \mathsf{on} \Leftrightarrow RP_{O2} < \vartheta_{RP}, \\
& \overline{RP}_{O2} = \mathsf{on} \Leftrightarrow O_{2ext} < \vartheta_Y, \quad \overline{RP}_{c1} = \mathsf{on} \Leftrightarrow Carbon1 \geq \vartheta_Y \\
& \overline{RP}_h = \mathsf{on} \Leftrightarrow \mathbf{v}_{Th} \geq \vartheta_v, \quad \overline{RP}_b = \mathsf{on} \Leftrightarrow \mathbf{v}_{R2b} \geq \vartheta_v, \\
& \overline{p}(t) = 1 \Rightarrow \mathbf{v}_p(t) \geq \varepsilon_{RP}, \text{ for all } p \in \mathcal{RP} \\
& \overline{p}(t) = 1 \Rightarrow \mathbf{v}_p(t) \geq \varepsilon_E, \text{ for all } p \in \mathcal{RE} \\
& \overline{p}(t) = 0 \Rightarrow \mathbf{v}_p(t) = 0, \text{ for all } p \in \mathcal{RP} \cup \mathcal{RE} \\
& \mathbf{P}(t), \mathbf{Y}(t), \mathbf{v}(t) \geq 0, \quad \overline{p}(t) \in \{0, 1\}, \text{ for all } p \in \mathcal{RP} \cup \mathcal{RE}, \\
& \text{for all } t \in [t_0, t_f].
\end{aligned}
\tag{6.31}
$$

t_0	t_f^1	t_f^2	kd_E	kd_{RP}	ϑ_{RP}	ϑ_v	ϑ_Y	ε_E	ε_{RP}	Φ_Q
0 min	90 min	50 min	0.01 1/min	0.5 1/min	1.0e-3 mmol	0.1 mmol/min	1 mmol	1.0e-6 mmol/min	1.0e-3 mmol/min	0.35

Table 6.4: Parameter values for $E \in \mathcal{E}, RP \in \mathcal{RP}$ and end times t_f^1, t_f^2 for Scenario 1 and 2.

Scenario 1: diauxie on two carbon sources

Our first scenario focuses again on the diauxie phenomenon. Initially, we set *Carbon* 1 and *Carbon* 2 to 1000 resp. 500 mmol, oxygen is given in excess, all other extracellular metabolites are set to 0. We do not specify the initial amounts

of the macromolecules. Instead, these are computed by the optimization algorithm under the constraint that the initial biomass should be 1g. Thus, the initial values for $t = t_0$ are:

$Carbon\ 1$	$Carbon\ 2$	D_{ext}	E_{ext}	F_{ext}	H_{ext}	O_{2ext}	$Biomass$
1000	500	0	0	0	0	$+\infty$	1

(6.32)

The Boolean variables are initialized by the 0-1 values for growth phase (a) in Table 6.5.

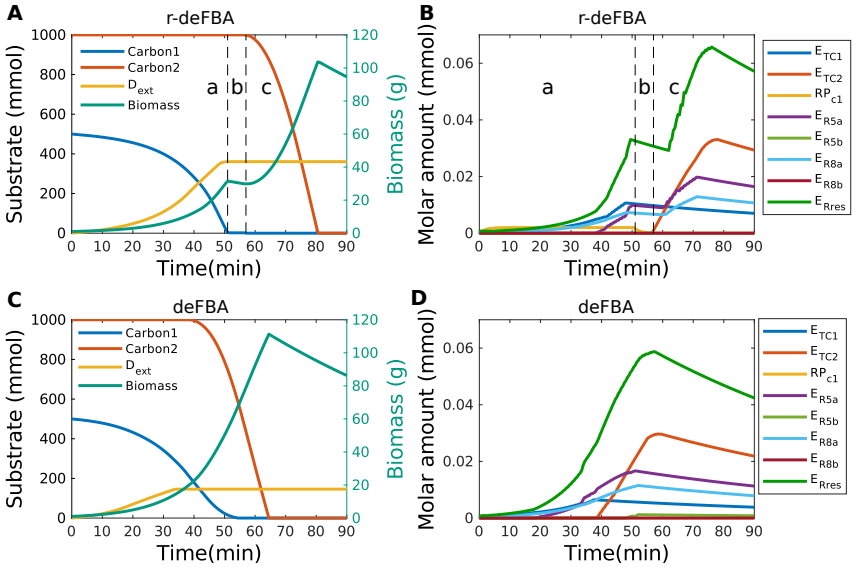


Figure 6.7: Dynamics of external metabolites (left axis), total biomass (right axis) predicted by r-deFBA (**A**) and deFBA (**C**), and key regulated proteins (**B** and **D**) in Scenario 1.

Comparing the results of r-deFBA and deFBA in Figure 6.7A resp. 6.7C, we note that in both approaches, *Carbon* 1 is metabolized first. Yet, the biphasic increase of biomass is predicted only by r-deFBA, and not by deFBA. Although most of the available *Carbon* 1 is utilized at the beginning, no lag phase is predicted by deFBA. The overall biomass production in the time interval $[t_0, t_f^1]$ predicted by deFBA amounts to 111.3g, which is 8% more than the 103.0g obtained by r-deFBA.

At the level of individual proteins, $\overline{RP}_{c1}$ is produced in growth phase (a) of r-deFBA, for which $Carbon\ 1 \geq \vartheta_Y$, see Figure 6.7B. Here, the expression of $\overline{E}_{Tc2}$ is inhibited since $\overline{RP}_{c1} \geq \vartheta_{RP}$. Thus, only *Carbon* 1 supports growth during this period. Once it is exhausted, the growth shifts to phase (b). The indicator variable $\overline{RP}_{c1}$ is triggered to be `off`, implying that RP_{c1} is degraded and not produced anymore. The expression of the transporter E_{Tc2} via $\overline{E}_{Tc2}$ is only activated when $RP_{c1} < \vartheta_{RP}$. Therefore, no carbon can be taken up during phase (b) and the total biomass production shows a lag phase. Finally, in growth phase (c), the transporter E_{Tc2} is produced and biomass production resumes based on *Carbon* 2.

Similarly in deFBA, E_{Tc2} is not synthesized as long as *Carbon* 1 supports a high growth rate. The protein dynamics for r-deFBA and deFBA in Figure 6.7B resp. Figure 6.7D are also relatively close. However, RP_{c1} is not produced at all in deFBA and there is no lag phase observed, see Figure 6.7D. In deFBA, the uptake of *Carbon* 2 starts well before *Carbon* 1 is exhausted, while in r-deFBA, *Carbon* 1 and *Carbon* 2 are metabolized one after the other, due to the regulatory control by RP_{c1}. The synthesis of RP_{c1} generates extra costs in r-deFBA, such that the total biomass in r-deFBA is smaller than in deFBA.

Growth phase	$\overline{RP}_{c1}$	$\overline{E}_{Tc2}$	$\overline{RP}_{O2}$	$\overline{E}_{5a}$	$\overline{E}_{5b}$	$\overline{E}_{Rres}$	$\overline{RP}_{b}$	$\overline{E}_{R2a}$	$\overline{E}_{R7}$	$\overline{RP}_{h}$	$\overline{E}_{R8a}$
a	1	0	0	1	0	1	0	1	1	0	1
b	0	0	0	1	0	1	0	1	1	0	1
c	0	1	0	1	0	1	0	1	1	0	1

Table 6.5: Discrete state transitions in Scenario 1.

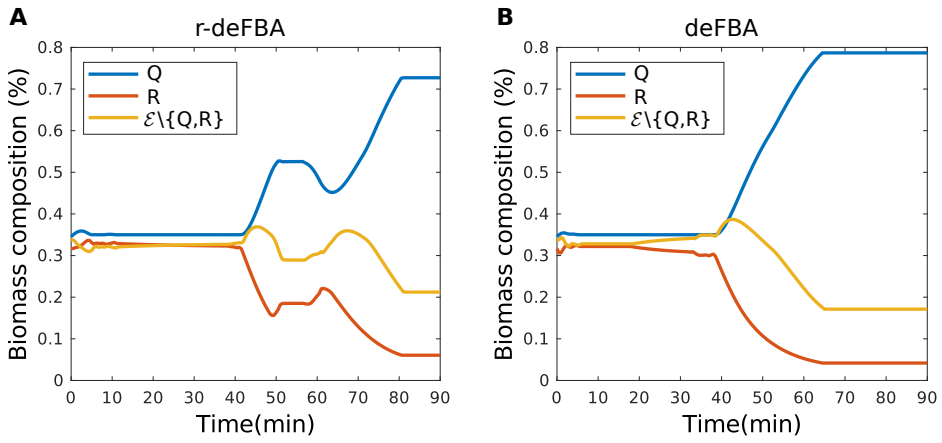


Figure 6.8: Dynamics of biomass composition in Scenario 1 with structural components Q, ribosomes R and enzymes $\mathcal{E} \setminus \{Q, R\}$.

The discrete state transitions for all the regulated proteins as predicted by r-deFBA are given in Table 6.5. Here, we group together each of the four regulatory proteins with the corresponding regulated enzymes. The expression of the enzymes regulated by RP_{O2} does not change since external oxygen is given in excess. Thus, $\overline{RP}_{O2}$ and $\overline{E}_{R5b}$ are always inhibited, while $\overline{E}_{R5a}$ and $\overline{E}_{Rres}$ remain activated. In Scenario 1, with no extracellular H_{ext} in the environment, reaction R_{2a} is constantly activated, consuming *Carbon* 1, *Carbon* 2, while T_h is inactive. Thus, the regulatory proteins $\overline{RP}_b$, $\overline{RP}_h$ are always `off`, and $\overline{E}_{R2a}$, $\overline{E}_{R7}$, $\overline{E}_{R8a}$ are on. Note that the r-deFBA framework allows computing an optimal regulatory strategy for maximizing growth even though the discrete state space is very large.

The resource allocation during the carbon switch can also be investigated. In Figure 6.8, we compare the dynamic biomass composition predicted by r-deFBA and

deFBA for three kinds of macromolecules: structural components, enzymes and ribosomes. At the beginning, both approaches exhibit a stable biomass composition. The fraction of structural components initially stays around 35%, which corresponds to the lower bound imposed by the biomass constraint in Eq. 6.9. As *Carbon* 1 is depleted, the structural components reach a rather high level, while the fractions of enzymes and ribosomes are decreasing in both predictions. Interestingly, in r-deFBA, the fractions of ribosomes and enzymes are increasing again while the structural components are going down at the outset of the second growth phase. This means that the cell must allocate more resources to the ribosomes to start the second growth phase. In the last step, r-deFBA predicts a high fraction of structural components and a low fraction of enzymes and ribosomes, which can also be validated by experiments (Fischer and Sauer, 2005). Overall, we obtain a biphasic resource allocation in r-deFBA, which is consistent with the two growth phases during diauxie. In deFBA, the quota fraction directly increases to about 80% of the total biomass and then keeps constant.

Scenario 2: growth on carbon and amino acid with amino acid in excess

Scenario 2 explores the dynamic growth on carbon and amino acid, with amino acid in excess. For $t = t_0$ we choose the initial values:

$Carbon\ 1$	$Carbon\ 2$	D_{ext}	E_{ext}	F_{ext}	H_{ext}	O_{2ext}	$Biomass$
100	0	0	0	0	250	$+\infty$	1

(6.33)

The Boolean variables are initialized by the 0-1 values for growth phase (a) in Table 6.6.

As can be seen from Figure 6.9, both r-deFBA and deFBA first predict a co-utilization of *Carbon* 1 and extracellular amino acid H_{ext}, followed by the utilization of H_{ext} once *Carbon* 1 has been exhausted. In the co-utilization phase, R_{2a} instead of R_{2b} is active to metabolize *Carbon* 1. Consequently, enzyme E_{R2a} is synthesized in this phase, but not E_{R2b}. When *Carbon* 1 gets almost exhausted, enzyme E_{R2b} starts being produced in order to activate reaction R_{2b}, see Figure 6.9B and Figure 6.9D. Now, B has to be generated from C, since B is needed for growth. The switch between R_{2a} and R_{2b} is predicted by both approaches because it benefits growth. Although in deFBA the expression of the regulatory protein RP_b is not triggered to inhibit the synthesis of E_{R2a}, the production of E_{R2a} stops in deFBA as well. Like E_{R2a}, enzyme E_{R8b} also has a similar dynamics in both approaches.

An interesting observation in the comparison is that deFBA activates enzyme E_{R5b} responsible for the anaerobic pathway, which is not consistent with the regulation. In contrast, r-deFBA produces enzyme E_{R5a} to catalyze reaction R_{5a}, in agreement with the regulation by RP_{O2}. Intuitively, R_{5a}, R_{5b} are two alternative reactions that play the same role in the network, one in the aerobic, the other in the anaerobic case. Using only optimization without any regulatory information, deFBA cannot guarantee to choose the right pathway. Since optimal solutions are not unique, the

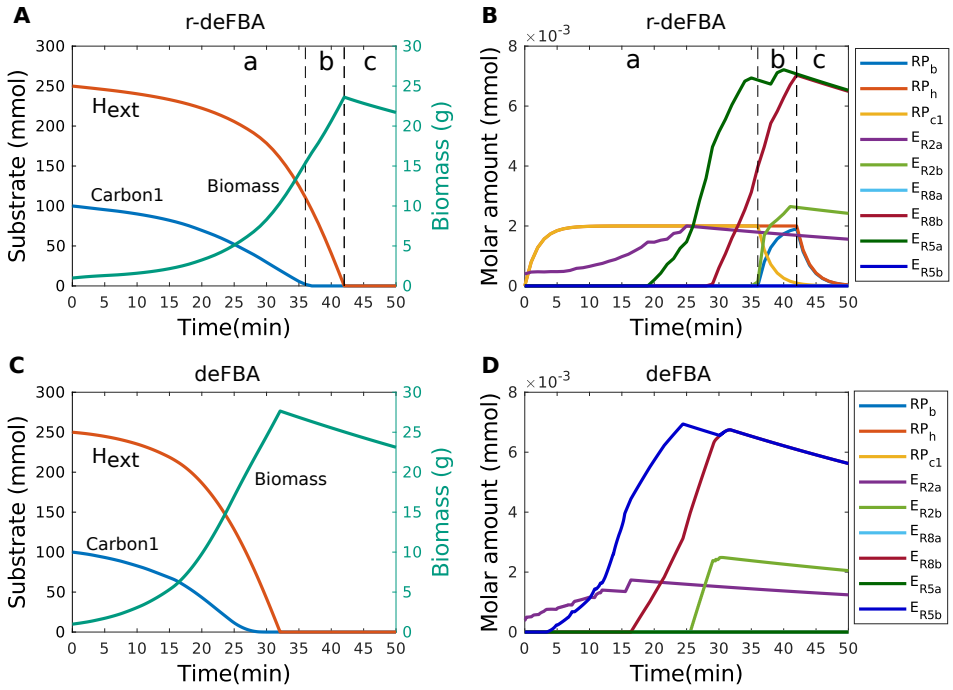


Figure 6.9: Dynamics of external metabolites (left axis) and biomass (right axis) predicted by r-deFBA(**A**) and deFBA (**C**), and key regulated proteins (**B** and **D**) in Scenario 2.

Growth phase	$\overline{RP}_{c1}$	$\overline{E}_{Tc2}$	$\overline{RP}_{O2}$	$\overline{E}_{5a}$	$\overline{E}_{5b}$	$\overline{E}_{Rres}$	$\overline{RP}_{b}$	$\overline{E}_{R2a}$	$\overline{E}_{R7}$	$\overline{RP}_{h}$	$\overline{E}_{R8a}$
a	1	0	0	1	0	1	0	1	1	1	0
b	0	0	0	1	0	1	1	0	0	1	0
c	0	1	0	1	0	1	0	0	0	0	0

Table 6.6: Discrete state transitions in Scenario 2.

solver can choose any of the two reactions or a combination thereof. Indeed, a small amount of E_{R5b} is produced by deFBA in the last phase of Scenario 1, even though this is not significant (see Figure 6.7D). Clearly, the consistency between metabolism and regulation cannot be ensured by deFBA without additional regulatory information. In contrast, the dynamics of metabolism in r-deFBA highly depends on the activity of the regulatory proteins, and the converse is also true.

Switching between active reactions by r-deFBA is illustrated in Figure 6.10. We can see in Figure 6.10A that in the beginning, *Carbon* 1 and H_{ext} are co-utilized and H is obtained only from H_{ext}. Since the model starts with a small biomass, even the TCA cycle is first inactive. Only when enzyme E_{R5a} has been synthesized is the TCA cycle activated after 18 minutes, see Figure 6.10B. Next in Figure 6.10C, reaction R_{8b} is activated to furnish the TCA cycle with amino acid H, while releasing ATP and NADH. Now, *Carbon* 1 is no longer sufficient to provide energy for growth. Finally, in Figure 6.10D, *Carbon* 1 has been exhausted, enzyme E_{R2b} is

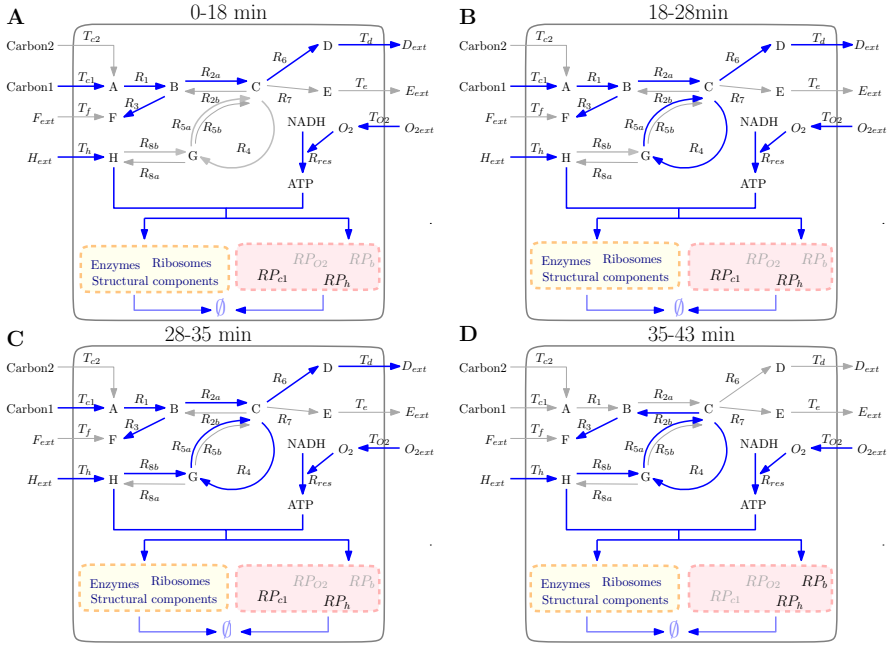


Figure 6.10: Patterns of active reactions predicted by r-deFBA in Scenario 2.

synthesized and R_{2b} is used to generate B.

Regarding the discrete state transitions, we divide the simulation period of r-deFBA again into three phases (a), (b), and (c), see Figure 6.9A and Figure 6.9B. During the last phase (c), there is no growth since all the nutrients are exhausted. The key regulatory pathways analyzed in Scenario 2 are operated by the regulatory proteins RP_b and RP_h. First, R_{2a} is active rather than R_{2b} for better metabolizing the carbon source. Hence, $\overline{RP}_b$ is `off` until *Carbon* 1 is used up. Soon after growth phase (a), R_{2b} has to be activated to use H, so that RP_b is triggered to be produced (see Figure 6.9B). The enzymes $\overline{E}_{R2a}$, $\overline{E}_{R7}$ then switch to `off`. Before H_{ext} has been exhausted, the expression state of RP_h is `on` because T_h has to be active for the uptake. Enzyme $\overline{E}_{R8a}$ is inhibited by RP_h. The state transitions related with RP_{O2} are the same as in Scenario 1 because external oxygen is given in excess during the whole period. *Carbon* 1 is given initially and exhausted at the end of growth phase (a). Although *Carbon* 2 is set to 0 and the cell cannot use it, the expression state of E_{Tc2} is activated when RP_{c1} is totally degraded, due to the regulatory constraints. In Scenario 2, this happens by chance at the time when H_{ext} is used up. So, $\overline{E}_{Tc2}$ is `on` in phase (c). Meanwhile, both *Carbon* 1 and H_{ext} have been used up and the two reaction signals $\mathbf{v}_{Th} > \vartheta_v$ and $\mathbf{v}_{R2b} > \vartheta_v$ are inactive. The indicator variables $\overline{RP}_b, \overline{RP}_h$ are turned `off` and the associated proteins RP_b, RP_h are degraded in the last phase, see Figure 6.9B.

6.7 Conclusion

Overall, r-deFBA computes optimal control strategies for hybrid automata representing metabolic-regulatory networks. Compared to previous approaches, in particular rFBA (Covert et al., 2001) and deFBA (Waldherr et al., 2015), r-deFBA allows for more realistic and accurate predictions by integrating the continuous dynamics of metabolism, including cellular resource allocation, with discrete regulatory control.

In purely discrete modeling frameworks for regulatory networks like the asynchronous logical formalism of R. Thomas (Thomas and Kaufman, 2001), it is not possible to quantify the time delay between discrete state transitions. The hybrid automata approach proposed in (Liu and Bockmayr, 2019b) solves this problem by using continuous variables for regulatory protein amounts together with thresholds that trigger the discrete jumps. However, exploring the dynamics of these hybrid automata is difficult due to the exponentially large discrete state space. By computing an optimal control strategy for the hybrid automaton, r-deFBA is able to predict, even in large state spaces, the quantitative dynamics of the regulatory proteins together with the sequence of discrete state transitions that are needed to achieve optimal growth.

In summary, r-deFBA allows predicting optimal cellular resource allocation in a dynamic environment by integrating metabolic reactions, enzyme-costs, quota compounds, and transcriptional regulation. Thus, r-deFBA considerably extends the predictive capabilities of current constraint-based modeling approaches as summarized in Table 4.1. Based on a hybrid discrete-continuous dynamics, r-deFBA is able to predict not only the continuous evolution of macromolecules and extracellular metabolites, but also the sequence of regulatory events needed to achieve an optimal growth. Finally, r-deFBA provides a solution for how to share enzymes between different reactions, which includes ribosome allocation in protein synthesis as a special case.

Chapter 7

Perspectives: Formalizing metabolic-regulatory networks at population-level by product automata

In this chapter, we illustrate a prospective application of the theoretical work in Chapter 5. In contrast to investigating individual-cell-level *metabolic-regulatory networks* (MRNs), the composition of hybrid automata is introduced to model the dynamics of metabolism and regulatory states at population-level. In particular, we formalize the population-level metabolic-regulatory networks by product automata. Such a product automaton indicates a cellular community, in which each cell is denoted by a hybrid automaton representing a MRN. To validate the formalization, we present two case studies in Section 7.4.

7.1 Introduction

Microbes are ubiquitously living in nature within communities. Mathematical models of metabolism and growth of microbial populations often lump the cells of a population into an aggregate model, which are so-called non-segregated models that do not distinguish the individual cells (De Jong et al., 2017). However, the effects of cell heterogeneity and the interactions between community members are lost in these non-segregated models. The microbial populations in nature or microbial cultures in industry are comprised of heterogeneous cells that may differ in size, intracellular state, metabolite concentrations, etc. The cells may be cooperating, communicating, competing, or regulating each other as a complex population.

For instance, there can be a competitive relationship in which the community members fight for the limited nutrients in an environment. It is also often the case in nature that one kind of cells enables the utilization of the by-products or secretions of another kind of cells. Such relationships are commensalism and mutualism. There are also signaling pathways that the cell could use to affect its neighboring cells, such as the Delta-Notch signaling pathway (Campos-Ortega, 1995; Appel et al., 2001). Such different cellular relationships may lead to completely diverse behaviors, structures, and dynamics of the community. For example, the microbial population in which the members could cross-feed would grow better than a community without such metabolically beneficial cooperation.

Studying and predicting the behaviors of a cellular community rather than of individual cells has important implications. For example, synthetic microbial consortia provide a platform to engineer population-level phenotypes that exhibit good robustness to the changing environment and are beneficial to the population only when cultured together (Brenner et al., 2008; Chen et al., 2015). Enabling the design of a microbial community with a particular purpose, such as the production of drugs, fuels or food, would be a highly promising application in the fields of biotechnology and synthetic biology. However, established constraint-based modeling approaches and resource allocation studies mostly focus on individual-based models. They fail to predict microbial community interactions (Gottstein et al., 2016; Succurro et al., 2017).

In Chapter 5, we formalized an individual-level *metabolic-regulatory network* (MRN) as a hybrid automaton. Such a hybrid automaton considers the integration of catabolism, anabolism, and transcriptional gene regulation, which is suitable for describing one cellular metabolic-regulatory system. Naturally, a microbial community which includes numerous cells can be modeled as a composition of hybrid automata denoting individual MRNs, which is called a product automaton.

In this chapter, we concentrate on exploring the population-level dynamics, including continuous metabolism and regulatory discrete states. Specifically, we model a cellular population as the composition of individual metabolic-regulatory networks. Mathematically, we formalize such a population with the theory of product automata. The cellular relationships that are considered in this formalization can be competitiveness, commensalism, and mutualism, in which the cells may interact by competing for the same nutrients or cross-feeding. Besides, considering that the cells may activate or inhibit each other by signaling molecules, we are also able to model the transcriptional response to the signals between cells. However, other cellular interactions and communications such as the physical interactions and signal transductions are hard to consider and are not covered in this chapter.

We first show that the microbial community composed of cells with relationships such as competition, commensalism and mutualism, can be considered as the composition of hybrid automata representing individual metabolic-regulatory networks. Such a microbial population is then formalized by a product automaton.

Finally, using the case studies in Section 7.4, we validate that the dynamics of continuous metabolism and regulatory discrete states of a cellular community can be analyzed by the product automata framework.

7.2 Hybrid system of the composition of MRNs

It is important to keep in mind that the cellular relationships considered here focus on metabolic interactions, particularly the metabolite exchange interactions, and regulations at the transcriptional level. The common kinds of cellular relationships in a population are competition, commensalism, and mutualism. We illustrate these relationships in Figure 7.1. In competition, the cells must compete for the shared nutrients in the environment. In commensalism, one cell may reuse the by-products secreted by another cell. There is also mutualism in which both cells could benefit from each other. However, other types of relationships, e.g. physical interactions, signal transductions and predation, are not discussed in this chapter.

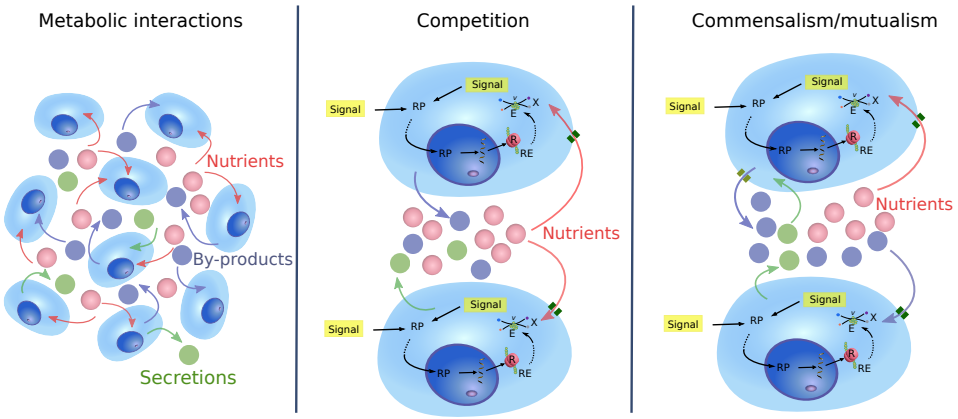


Figure 7.1: Metabolite exchange interactions between community members.

In Figure 7.2, we abstract a cellular community to a population-level metabolic-regulatory network, which considers the metabolic interactions or transcriptional regulations between community members. The abbreviations and notation are the same as explained in Section 5.2. Figure 7.2 presents a two-cellular community composed by cell 1 and cell 2, which are indicated by the two associated MRNs. However, it would also be possible to model a multi-cellular community by using the composition of multiple MRNs.

7.2.1 Continuous variables

Clearly, the cells within a community may share extracellular metabolites represented by $\mathcal{Y}$. Thus, we define the total extracellular species as the union of the external metabolites of each MRN, which is $\mathcal{Y}_{cell1} \cup \mathcal{Y}_{cell2}$ in the two-cellular case shown in Figure 7.2. The intracellular metabolite species of the community

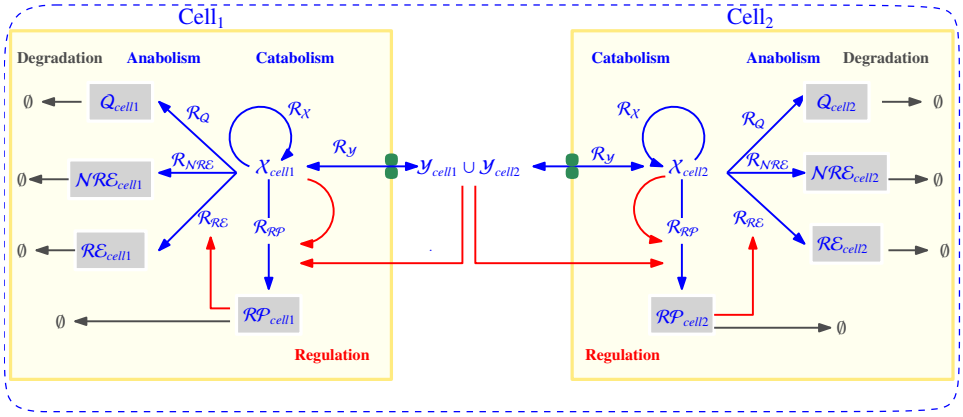


Figure 7.2: A two-cellular metabolic-regulatory network. The by-products or secretions belong to the extracellular species denoted by the union set $\mathcal{Y}_{cell1} \cup \mathcal{Y}_{cell2}$. We remark that the signaling molecules produced by one cell to regulate the other are also included in the extracellular species set. The individual MRN structure has been described in Section 5.2.

members are independent. Therefore, the continuous variables are all metabolite amounts, including the extracellular species and intracellular metabolites of all the cells in the community. Thus, we obtain a set of molecular species of the microbial community of cell 1 and cell 2:

$$\begin{aligned}\mathcal{M} = \mathcal{Y}_{cell1} \cup \mathcal{Y}_{cell2} \cup \mathcal{X}_{cell1} \cup \mathcal{X}_{cell2} \cup \mathcal{E}_{cell1} \cup \\ \mathcal{E}_{cell2} \cup \mathcal{Q}_{cell1} \cup \mathcal{Q}_{cell2} \cup \mathcal{RP}_{cell1} \cup \mathcal{RP}_{cell2}.\end{aligned} \tag{7.1}$$

The purely continuous dynamics of metabolites can be modeled by ODEs according to the Law of Mass Action (Waage and Guldberg, 1864), Michaelis-Menten kinetics (Menten and Michaelis, 1913), or Monod equation (Monod, 1949), which can be formulated by

$$\dot{\mathbf{M}}(t) = \frac{d\mathbf{M}}{dt} = F(\mathbf{M}, \mathbf{K}, S, t), \tag{7.2}$$

where $\mathbf{K}$ includes all the kinetic parameters, and S represents the stoichiometric coefficients of the metabolic reactions of the cellular community and t is time.

7.2.2 Discrete states

For the discrete states, we assume that for each regulated protein p in the population there are two possible states `on` and `off`, describing whether or not the gene encoding p is expressed at a particular time t, cf. Chapter 5. As shown in Figure 7.2, the cells in a microbial community only share extracellular metabolites, particularly nutrients, by-products, and signaling molecules. The intracellular metabolites are separated by cellular membranes. In other words, the MRNs that describe community cells have common continuous variables but do not share discrete states of transcriptional regulation.

Therefore, the whole discrete state space consists of all the possible combinations of expression states of regulated proteins in all community members, if we compose the relevant MRNs together as a population-level metabolic-regulatory network. Thus, we obtain the discrete state space $\{0,1\}^{\mathcal{RP}_{cell1}\mathcal{RE}_{cell1}\cup\mathcal{RP}_{cell2}\cup\mathcal{RE}_{cell2}}$ for the two-cellular community in Figure 7.2. Here, the Boolean value 0 corresponds to `off` and the value 1 to `on`. The regulatory rules for discrete state transitions in the community can be formalized as

$$\overline{p}(t) = f_p(\mathbf{RP}_{cell_i}, \mathbf{Y}_{cell1\cup cell2}(t), \mathbf{X}_{cell_i}(t)), \tag{7.3}$$

where for all $p \in \mathcal{RP}_{cell_i} \cup \mathcal{RE}_{cell_i}, i \in \{1,2\}$ and f_p is the logical function with Boolean combinations of atomic formulas of the form $x \geq \theta$, where x is a real continuous variable and θ denotes the relevant threshold value.

7.3 Combining discrete and continuous dynamics in a product automaton

In Figure 7.2, the two-cellular community consists of two interacting metabolic-regulatory networks. We model each as a hybrid automaton H_{cell1} and H_{cell2} respectively. The two hybrid automata interact with each other by shared metabolites and regulatory events.

Composing the discrete states together with continuous dynamics of hybrid automaton $H_{cell1} = \{Loc_{cell1}, \Sigma_{cell1}, Edge_{cell1}, X_{cell1}, Init_{cell1}, Inv_{cell1}, Flow_{cell1}, Jump_{cell1}\}$ and hybrid automaton $H_{cell2} = \{Loc_{cell2}, \Sigma_{cell2}, Edge_{cell2}, X_{cell2}, Init_{cell2}, Inv_{cell2}, Flow_{cell2}, Jump_{cell2}\}$ with $Loc_{cell1} \cap Loc_{cell2} = \emptyset$, leads to a product automaton

$$H = (Loc, \Sigma, Edge, X, Init, Inv, Flow, Jump) \tag{7.4}$$

with the the following components (Raskin, 2005):

- Loc is a finite set of all possible combinations of the expression states of all the regulated proteins inside the two cells, including regulatory proteins and regulated enzymes. So, $Loc = \{0,1\}^{\mathcal{RE}_{cell1}\cup\mathcal{RP}_{cell1}\cup\mathcal{RE}_{cell2}\cup\mathcal{RP}_{cell2}}$.
- $\Sigma = \Sigma_{cell1} \cup \Sigma_{cell2}$ is a finite set of events. The events are described by the regulatory rules represented by the logical function in Eq. 7.3.
- $Edge \subseteq Loc \times \Sigma \times Loc$ represents the possible state transitions, which are labeled by events from Σ. We have $((l^1_{cell1}, l^1_{cell2}), \sigma, (l^2_{cell1}, l^2_{cell2})) \in Edge$ iff either:
  1. $\sigma \in \Sigma_{cell1} \backslash \Sigma_{cell2}, (l^1_{cell1}, \sigma, l^2_{cell1}) \in Edge_{cell1}$ and $l^1_{cell2} = l^2_{cell2}$;
  2. $\sigma \in \Sigma_{cell2} \backslash \Sigma_{cell1}, (l^1_{cell2}, \sigma, l^2_{cell2}) \in Edge_{cell2}$ and $l^1_{cell1} = l^2_{cell1}$;
  3. $\sigma \in \Sigma_{cell1} \cap \Sigma_{cell2}, (l^1_{cell1}, \sigma, l^2_{cell1}) \in Edge_{cell1}$ and $(l^1_{cell2}, l^2_{cell2}) \in Edge_{cell2}$.

In condition 1 and condition 2, the two automata do not share events and the unshared events are interleaved. However, condition 3 expresses that the shared events have to occur synchronously in the two automata.

– $X = X_{cell1} \cup X_{cell2}$ represents a finite set of continuous variables, including all external species, intermediate metabolites, regulatory proteins, catalytic molecules and quota molecules of the two cells. That is $X = \mathcal{M} = \mathcal{Y}_{cell1} \cup \mathcal{Y}_{cell2} \cup \mathcal{X}_{cell1} \cup \mathcal{X}_{cell2} \cup \mathcal{RP}_{cell1} \cup \mathcal{RP}_{cell2} \cup \mathcal{E}_{cell1} \cup \mathcal{E}_{cell2} \cup \mathcal{Q}_{cell1} \cup \mathcal{Q}_{cell2}$.

– $Init((l_{cell1}, l_{cell2}))$ is a predicate that describes the possible initial values for the continuous variables when the automaton starts in state (l_{cell1}, l_{cell2}) with $Init((l_{cell1}, l_{cell2})) = Init_{cell1}(l_{cell1}) \wedge Init_{cell2}(l_{cell2})$ for any $(l_{cell1}, l_{cell2}) \in Loc$.

– $Inv((l_{cell1}, l_{cell2}))$ is a predicate that describes the possible values of the continuous variables when the control of the automaton lies in (l_{cell1}, l_{cell2}) with $Inv((l_{cell1}, l_{cell2})) = Inv_{cell1}(l_{cell1}) \wedge Inv_{cell2}(l_{cell2})$ for any $(l_{cell1}, l_{cell2}) \in Loc$.

– $Flow((l_{cell1}, l_{cell2}))$ is a predicate which describes the possible continuous evolutions when the control of the automaton stays in (l_{cell1}, l_{cell2}) with $Flow((l_{cell1}, l_{cell2})) = Flow_{cell1}(l_{cell1}) \wedge Flow_{cell2}(l_{cell2})$ for any $(l_{cell1}, l_{cell2}) \in Loc$.

– $Jump$ is a function that assigns to each edge $((l^1_{cell1}, l^1_{cell2}), \sigma, (l^2_{cell1}, l^2_{cell2})) \in Edge$ a predicate $Jump((l^1_{cell1}, l^1_{cell2}), \sigma, (l^2_{cell1}, l^2_{cell2}))$ describing when the discrete change modeled by $((l^1_{cell1}, l^1_{cell2}), \sigma, (l^2_{cell1}, l^2_{cell2}))$ is possible and what the possible updates for the continuous variables are when the transition is made. For any edge $((l^1_{cell1}, l^1_{cell2}), \sigma, (l^2_{cell1}, l^2_{cell2})) \in Edge$, we have that:

  1. $Jump((l^1_{cell1}, l^1_{cell2}), \sigma, (l^2_{cell1}, l^2_{cell2})) = Jump((l^1_{cell1}, \sigma, l^2_{cell1})) \wedge \bigwedge_{x \in X_{cell2} \setminus X_{cell1}} x' = x$ if $\sigma \in \Sigma_{cell1} \setminus \Sigma_{cell2}$;
  2. $Jump((l^1_{cell1}, l^1_{cell2}), \sigma, (l^2_{cell1}, l^2_{cell2})) = Jump((l^1_{cell2}, \sigma, l^2_{cell2})) \wedge \bigwedge_{x \in X_{cell1} \setminus X_{cell2}} x' = x$ if $\sigma \in \Sigma_{cell2} \setminus \Sigma_{cell1}$;
  3. $Jump((l^1_{cell1}, l^1_{cell2}), \sigma, (l^2_{cell1}, l^2_{cell2})) = Jump((l^1_{cell1}, \sigma, l^2_{cell1})) \wedge Jump((l^1_{cell2}, \sigma, l^2_{cell2}))$ if $\sigma \in \Sigma_{cell1} \cap \Sigma_{cell2}$.

  In conditions 1 and 2, the effect of discrete jumps that are local to one hybrid automaton is described by the jump function of that automaton. Condition 3 expresses that the effect is the conjunction of the effects of each discrete jump when the discrete changes are shared by the two automata.

The formalization above presents the composition of two hybrid automata H_{cell1} and H_{cell2}, which indicates a population consisting of cell 1 and cell 2. However, it would also be possible to model a multi-cellular community using the composition of multiple hybrid automata of MRNs. For example, we present a case study that models a three-cellular competition using a product automaton in Section 7.4.1.

7.4 Biological applications

7.4.1 Modeling competitiveness of cells having different β_R

Several studies have posited that cellular growth is a balance between production of ribosomal and non-ribosomal proteins (Maitra and Dill, 2015), and a result of how the cells allocate their resources to diverse functional proteins (Weiße et al., 2015). In this case, we consider three cells in a community, which allocate different fractions of resources to their ribosomes. Each cell is represented by the carbon catabolite repression network (CCR) shown in Figure 5.4. For ease of understanding, we illustrate the three-cellular community in Figure 7.3.

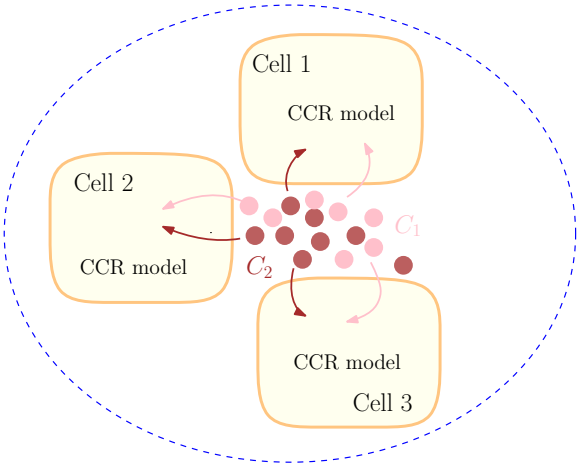


Figure 7.3: A three-cellular community composed of three CCR models described in Section 5.5.1.

In the dynamics of a hybrid automaton H_{diaux} describing the CCR model, β_R denotes the fraction of resources allocated to the production of ribosome R, see Section 5.5.1. In this section, we use a product automaton as the composition of three hybrid automata of type H_{diaux} to model the competition between three cells for the common carbon sources C_1 and C_2.

In each CCR network (see Figure 5.4), there are 5 intracellular metabolites M, Q, T_1, T_2, RP. The expression states of the two proteins RP, T_2 are regulated. Since C_1 and C_2 are shared by the three cells, the continuous variables of the three-cellular system are composed of the two extracellular carbon sources and 5×3 intracellular metabolites, which is 17 in total. The discrete expression state of the regulatory protein RP in each model is regulated by the same external carbon source C_1. The expression state of T_2 in each cell is decided by the regulatory protein RP in the same cell. Clearly, the three cells do not share discrete states. Thus, we obtain the set of discrete states $Loc = \{(\overline{RP}_{cell1}, \overline{T_2}_{cell1}, \overline{RP}_{cell2}, \overline{T_2}_{cell2}, \overline{RP}_{cell3}, \overline{T_2}_{cell3}) \mid \overline{RP}_{cell1}, \overline{T_2}_{cell1}, \overline{RP}_{cell2}, \overline{T_2}_{cell2}, \overline{RP}_{cell3}, \overline{T_2}_{cell3} \in \{0, 1\}\}$, which are 2^6 in total.

For the dynamics of the continuous variables, we shall consider intracellular

metabolites and extracellular substrates separately. The intracellular metabolites in different cells do not depend on each other, and the three cells also do not share discrete states. So, the dynamics of intracellular metabolites can be defined by the ODEs system that corresponds to each state as described in Section 5.5.1. Take the production rates of proteins in cell 1 as example. The synthesis rate $\mathbf{v}_p$ of a protein p in cell 1 that consists of $\mathbf{n}_p$ amino acids, is

$$\mathbf{v}_p = \frac{\beta_{p_{cell1}}}{\mathbf{n}_p} \cdot \mathbf{v}_M, \text{ with } \mathbf{v}_M = \frac{kr \cdot M_{cell1} \cdot R_{cell1}}{Kr + M_{cell1}}, \tag{7.5}$$

for all $p \in \{RP_{cell1}, T1_{cell1}, T2_{cell1}, R_{cell1}, Q_{cell1}\}$. The parameter values of $\mathbf{n}_p$, kr, and Kr are listed in Table 5.1.

The difference between the three CCR models is that we set β_R to relatively high, middle and low values. Technically, we individually fix $(\beta_{R_{cell1}}, \beta_{R_{cell2}}, \beta_{R_{cell3}}) = (0.4, 0.25, 0.1)$ for the three CCR models, regardless of the discrete states, and then average the resources left for the other proteins that are expressed in each state. Take cell 1 for example, $\beta_{p_{cell1}} = (1 - \beta_{R_{cell1}})/4$, $p \in \{Q_{cell1}, RP_{cell1}, T1_{cell1}, T2_{cell1}\}$ when $\overline{RP}_{cell1}, \overline{T2}_{cell1}$ are both on, and $\beta_{p_{cell1}} = (1 - \beta_{R_{cell1}})/2$, $p \in \{Q_{cell1}, T1_{cell1}\}$ when $\overline{RP}_{cell1}, \overline{T2}_{cell1}$ are both off. The other parameters and initial conditions are set to be the same values for each cellular model to eliminate the effects of other factors.

The extracellular substrates C_1 and C_2 provide carbon sources for all three cells. Thus, their dynamics depend on the transporters of the three cells given by

$$\dot{C}_1 = -\frac{kcat1 \cdot C_1 \cdot T1_{cell1}}{K_T + C_1} - \frac{kcat1 \cdot C_1 \cdot T1_{cell2}}{K_T + C_1} - \frac{kcat1 \cdot C_1 \cdot T1_{cell3}}{K_T + C_1}, \tag{7.6}$$

$$\dot{C}_2 = -\frac{kcat2 \cdot C_2 \cdot T2_{cell1}}{K_T + C_2} - \frac{kcat2 \cdot C_2 \cdot T2_{cell2}}{K_T + C_2} - \frac{kcat2 \cdot C_2 \cdot T2_{cell3}}{K_T + C_2}. \tag{7.7}$$

All the kinetic parameters are the same as in the CCR model of Section 5.5.1, see Table 5.1.

The threshold $\gamma = 20$ mmol while the $\alpha = 0.01$ mmol in this case. As initial condition, we set $C_1 = C_2 = 10^3$ mmol. The initial values of intracellular metabolites in the three cells are set to: $(C, M, RP, R, T_1, T_2, Q) = (200, 0, 0.01, 0, 0, 0.1)$ for each cell. As initially $C_1 = 1000$ and $RP = 0$ in this case, $\overline{RP}$ is on and $\overline{T}_2$ is on in all the three cells. Hence, the three-cellular system initially is in location (on, on, on, on, on, on).

The results are shown in Figure 7.4 and Figure 7.5. We can see in Figure 7.5 that C_2 is only used after C_1 is exhausted. From the dynamics of RP and T_2, we can see that the state transitions in the three models are (on, on) → (on, off) → (off, off) → (off, on), where the first component is $\overline{RP}$ and the second is $\overline{T}_2$. The discrete jumps of these three cells happen almost at the same time point. As we set $(\beta_{R_{cell1}}, \beta_{R_{cell2}}, \beta_{R_{cell3}}) = (0.4, 0.25, 0.1)$ for the three cells, the ribosome production in cell 1 is fastest in the three-cellular community.

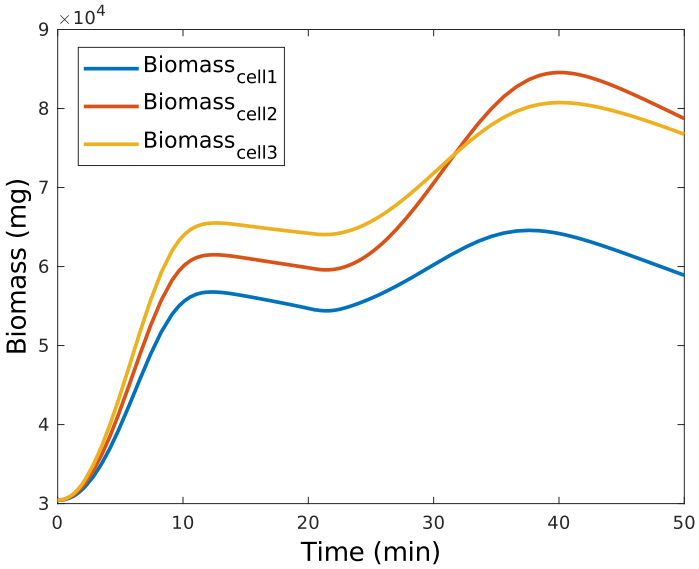


Figure 7.4: Biomass comparison.

Since the cell 1 allocates 40% of its resources M_{cell1} to ribosome R_{cell1}, it has the highest ribosomal production, see Figure 7.5. However, unexpectedly, the production of other proteins is not much faster as compared to cell 2 and cell 3. In particular, the $T2_{cell1}$ amount is even lower than the amount of $T2_{cell2}$ and $T2_{cell3}$. Looking at the biomass comparison in Figure 7.4, we can see the biomass of cell 1 is increasing the slowest. The reason is evident if we further look at the dynamics of the precursors M. A higher production of ribosome would consume so many precursors that the production of other enzymes is limited by the low level of the precursor pool. The situation gets even worse at around 25 min in that M_{cell1} gets close to 0. Therefore, the biomass of cell 1 increases comparatively slowly within this three-cellular community (see Figure 7.4).

Another interesting observation is that $Biomass_{cell2}$ exceeds $Biomass_{cell3}$ after around 31 min, see Figure 7.4. Cell 2 finally ends up with the maximum biomass among the three cells. The reason is also obvious. Allocating only 10% of the resources to ribosome in cell 3 results in the lowest level of ribosome R_{cell3}. Although the remaining precursors that are available for the non-ribosomal proteins are in excess inside cell 3, the production rates of proteins are limited by the lowest ribosome amount compared with the other two cells.

We conclude that allocating too many resources to the production of ribosomal proteins would shrink the precursor pool, which further slows down the protein expression level. Yet, allocating too few resources to the ribosomal proteins may result in a low ribosome level that also retards the overall protein expression rates. Therefore, the resource allocation to ribosomal and non-ribosomal proteins significantly affects the cellular growth, and the cell with the optimal resource allocation

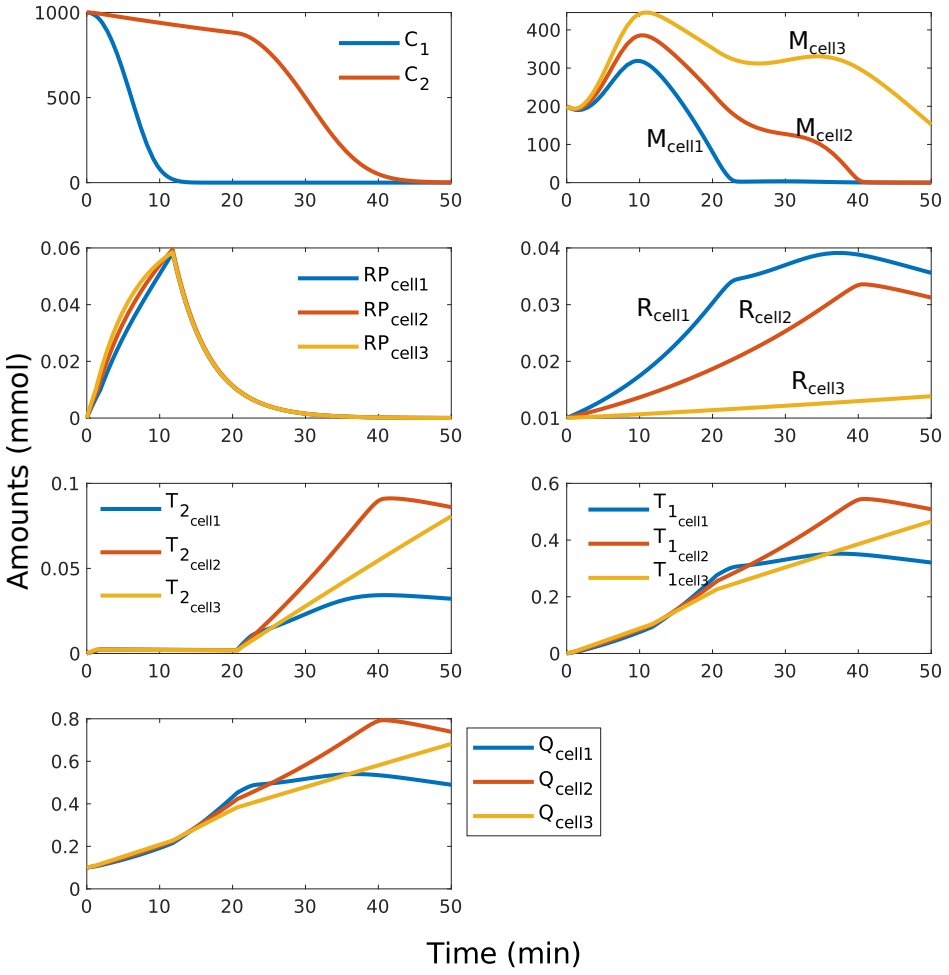


Figure 7.5: Dynamics of metabolite species in the three-cellular community.

would gain larger biomass and win in the competition within the microbial community.

7.4.2 A community consisting of activator and repressor strains

Constructing "activator" and "repressor" strains

In this section, we use our product automata framework to model a community consisting of two distinct microbial strains, inspired by the work in (Chen et al., 2015). First, an "activator" strain and a "repressor" strain are designed to create a microbial consortium, see Figure 7.6. Figure 7.6A shows that the "activator" strain increases the protein expression in both strains while the "repressor" strain decreases the protein expression in both strains. To model such activator-repressor popula-

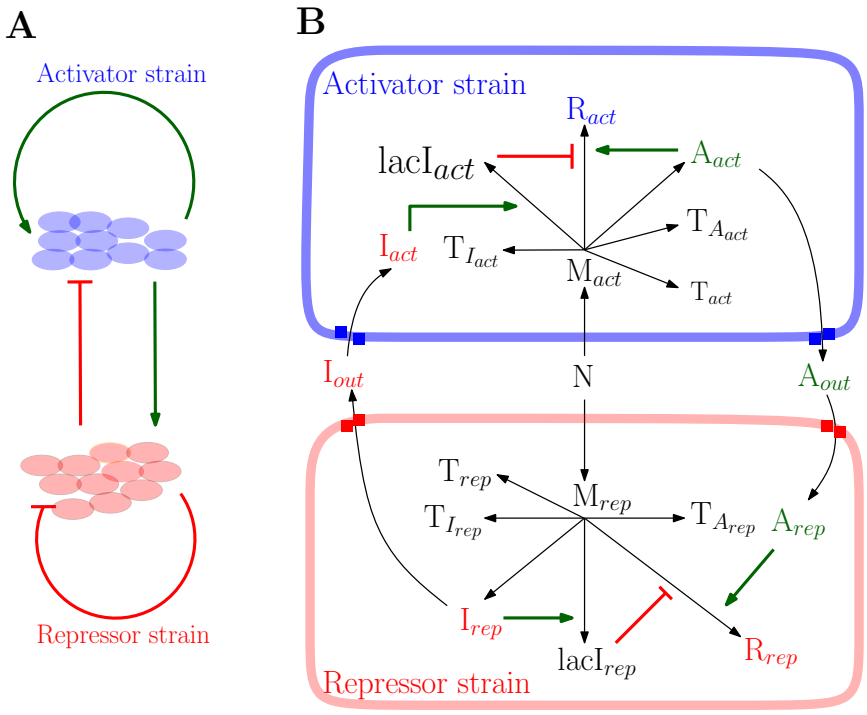


Figure 7.6: A community consisting of "activator" and "repressor" strains and their metabolic-regulatory networks. By regulating the expression state of ribosomes (R_{act} and R_{rep}), the "activator" strain increases the protein expression in both strains while the "repressor" strain decreases the protein expression in both strains.

tion with a product automaton, we design two metabolic-regulatory networks that enzymatically produce and transcriptionally respond to signaling molecules.

In detail, N represents the shared nutrients of the two strains. M_{act}, M_{rep} denote the precursors that are produced from nutrients N and catalyzed by enzymes T_{act} and T_{rep} respectively. The precursors M_{act}, M_{rep} are used to build proteins individually in each strain. R_{act} and R_{rep} are ribosomes. In both strains, we use the expression state of ribosomes, which is denoted by $\overline{R}_{act}, \overline{R}_{rep} \in \{\texttt{on}, \texttt{off}\}$, to indicate the level of protein expression. This is because all the translational rates of proteins are affected by ribosome abundance. If the production state of ribosome turns `off`, protein expression will correspondingly decline.

For the positive regulation, we assume that the activator strain enables the production of a signaling molecule A_{act} while the repressor strain does not. On one hand, A_{act} activates the expression state of ribosome $\overline{R}_{act}$ with threshold θ_A. $\overline{R}_{act}$ has two

states, either on or off, depending on A_{act}, $lacI_{act}$ and their thresholds, see Figure 7.6. On the other hand, A_{act} is secreted into A_{out} to the medium by transporter $T_{A_{act}}$. A_{out} is further uptaken into A_{rep} by transporter $T_{A_{cap}}$ of the repressor strain. This transportation is the only way to accumulate A_{cap} in the repressor strain as it does not produce it. A_{rep} also serves as the signaling molecule to activate the production state $\overline{R}_{rep}$ of R_{rep} with threshold θ_A. $\overline{R}_{rep}$ is either on or off, depending on A_{rep}, $lacI_{rep}$ and their thresholds. Thus, by producing the signaling molecule A_{act}, the activator strain activates the protein expression of both strains.

For the regulation of ribosome production, the regulatory proteins $lacI_{act}$ and $lacI_{rep}$ are additionally introduced to inhibit $\overline{R}_{act}$ and $\overline{R}_{rep}$ respectively, together with the signaling molecules A_{act} and A_{rep}. Therefore, we obtain the regulatory rules of $\overline{R}_{act}$ and $\overline{R}_{rep}$:

$$\overline{R}_{act} = \texttt{on} \;\Leftrightarrow\; (A_{act} \geq \theta_A \;\text{ and }\; lacI_{act} < \theta_{lacI}), \tag{7.8}$$

$$\overline{R}_{rep} = \texttt{on} \;\Leftrightarrow\; (A_{rep} \geq \theta_A \;\text{ and }\; lacI_{rep} < \theta_{lacI}). \tag{7.9}$$

where θ_A and θ_{lacI} are thresholds.

For a negative regulation, we introduce I_{rep} as a signaling molecule that is only produced in the repressor strain. When above its threshold θ_I, I_{rep} activates the production of regulatory protein $lacI_{rep}$. Above the threshold θ_{lacI}, $lacI_{rep}$ further inhibits the expression of R_{act}. Meanwhile, I_{rep} is secreted to the medium by transporter $T_{I_{rep}}$. I_{out} is then uptaken by transporter $T_{I_{act}}$ to accumulate I_{act} in the activator strain. I_{act} also activates the production of $lacI_{act}$, which further inhibits the expression of R_{act}. Therefore, we get the regulatory rules of $\overline{lacI}_{act}$ and $\overline{lacI}_{rep}$:

$$\overline{lacI}_{act} = \texttt{on} \;\Leftrightarrow\; (I_{act} \geq \theta_I), \tag{7.10}$$

$$\overline{lacI}_{rep} = \texttt{on} \;\Leftrightarrow\; (I_{rep} \geq \theta_I), \tag{7.11}$$

where θ_I is the threshold value.

For the signaling molecules A_{act} and I_{rep}, we assume they are small proteins composed of 100 amino acids produced from precursors M_{act} and M_{rep} respectively. Therefore, in the activator strain, the production of the proteins T_{act}, $T_{I_{act}}$, $T_{A_{act}}$, A_{act}, R_{act}, $lacI_{act}$ is catalyzed by ribosome R_{act}. In the repressor strain, it is R_{rep} that catalyzes the production of the proteins T_{rep}, $T_{I_{rep}}$, $T_{A_{rep}}$, I_{rep}, R_{rep}, $lacI_{rep}$. The full details of the metabolic-regulatory networks of the two strains are given in Table 7.1.

Modeling the activator-repressor consortium by a product automaton

In this section, we build a product automaton $H_{act-rep}$ for simulating the community of the activator strain and repressor strain as illustrated in Figure 7.6B. We obtain the continuous variables $X_{act-rep} = \{N,\ M_{act},\ A_{act},\ R_{act},\ T_{act},\ T_{A_{act}},\ T_{I_{act}},$

Activator strain				
Biomass reaction $\mathbf{n}_p M \to p$	Enzyme	Turnover rate $k_{cat}^{p,R} = k_r/\mathbf{n}_p$ [1/min]	Degradation rate $\mathbf{kd}_p$ [1/min]	Regulation
$400M_{act} \to T_{act}$	R_{act}	3.15	0.01	
$7459M_{act} \to R_{act}$	R_{act}	0.1689	0.01	$A_{act} \geq \theta_A$ & $lacI_{act} < \theta_{lacI}$
$1500M_{act} \to T_{A_{act}}$	R_{act}	0.84	0.01	
$1000M_{act} \to T_{I_{act}}$	R_{act}	1.26	0.01	
$300M_{act} \to lacI_{act}$	R_{act}	4.2	0.2	$I_{act} \geq \theta_I$
$100M_{act} \to A_{act}$	R_{act}	12.6	0.01	
Metabolic reaction	Enzyme	Turnover rate [1/min]		
$N \to M_{act}$	T_{act}	$k_{catN} = 3000$	0	
$A_{act} \to A_{out}$	$T_{A_{act}}$	$k_{catA} = 100$	0	
$I_{out} \to I_{act}$	$T_{I_{act}}$	$k_{catI} = 250$	0.05	
Repressor strain				
Biomass reaction $\mathbf{n}_p M \to p$	Enzyme	Turnover rate $k_{cat}^{p,R} = k_r/\mathbf{n}_p$ [1/min]	Degradation rate $\mathbf{kd}_p$ [1/min]	Regulation
$400M_{rep} \to T_{rep}$	R_{rep}	3.15	0.01	
$7459M_{rep} \to R_{rep}$	R_{rep}	0.1689	0.01	$A_{rep} \geq \theta_A$ & $lacI_{rep} < \theta_{lacI}$
$1500M_{rep} \to T_{A_{rep}}$	R_{rep}	0.84	0.01	
$1000M_{rep} \to T_{I_{rep}}$	R_{rep}	1.26	0.01	
$300M_{rep} \to lacI_{rep}$	R_{rep}	4.2	0.2	$I_{rep} \geq \theta_I$
$100M_{rep} \to I_{rep}$	R_{rep}	12.6	0.05	
Metabolic reaction	Enzyme	Turnover rate [1/min]		
$N \to M_{rep}$	T_{rep}	$k_{catN} = 3000$	0	
$A_{out} \to A_{rep}$	$T_{A_{rep}}$	$k_{catA} = 100$	0.01	
$I_{rep} \to I_{out}$	$T_{I_{rep}}$	$k_{catI} = 250$	0	

Table 7.1: Metabolic and protein production reactions with corresponding parameters.

$lacI_{act}$, I_{act}, A_{rep}, R_{rep}, T_{rep}, $T_{A_{rep}}$, $T_{I_{rep}}$, $lacI_{rep}$, I_{rep}, A_{out}, $I_{out}\}$ describing time-dependent molar amounts of all the metabolites. The discrete locations correspond to the different $\texttt{on} - \texttt{off}$ states of $\overline{R}_{act}, \overline{lacI}_{act}, \overline{R}_{rep}$ and $\overline{lacI}_{rep}$, which means there are $2^4 = 16$ discrete states. So, $Loc_{act-rep} = \{(\overline{R}_{act}, \overline{lacI}_{act}, \overline{R}_{rep}, \overline{lacI}_{rep}) \mid \overline{R}_{act}, \overline{lacI}_{act}, \overline{R}_{rep}, \overline{lacI}_{rep} \in \{\texttt{on}, \texttt{off}\}\}$.

We also use a set of Michaelis-Menten equations to specify the dynamics of $H_{act-rep}$ at each location. Nutrients N provide resources for both strains. Below, we illustrate the dynamics of the continuous variables in state $(\texttt{on}, \texttt{on}, \texttt{on}, \texttt{on})$ as an example:

1. $\dot{N} = -\frac{k_{catN} \cdot N \cdot T_{act}}{K_M + N} - \frac{k_{catN} \cdot N \cdot T_{rep}}{K_M + N}$,

2. $\dot{M}_{act} = \frac{k_{catN} \cdot N \cdot T_{act}}{K_M + N} - \frac{kr \cdot R_{act} \cdot M_{act}}{Kr + M_{act}}$,

3. $\dot{A}_{act} = \frac{1}{6} \cdot \frac{k_{cat}^{A_{act},R} \cdot R_{act} \cdot M_{act}}{K_r + M_{act}} - \frac{k_{catA} \cdot T_{A_{act}} \cdot A_{act}}{K_M + A_{act}} - \mathbf{kd}_{A_{act}} \cdot A_{act}$,

4. $\dot{T}_{act} = \frac{1}{6} \cdot \frac{k_{cat}^{T_{act},R} \cdot R_{act} \cdot M_{act}}{K_r + M_{act}} - \mathbf{kd}_{T_{act}} \cdot T_{act}$,

5. $\dot{T}_{A_{act}} = \frac{1}{6} \cdot \frac{k_{cat}^{T_{A_{act}},R} \cdot R_{act} \cdot M_{act}}{K_r + M_{act}} - \mathbf{kd}_{T_{Aact}} \cdot T_{A_{act}}$,

6. $\dot{T}_{I_{act}} = \frac{1}{6} \cdot \frac{k_{cat}^{T_{I_{act}},R} \cdot R_{act} \cdot M_{act}}{K_r + M_{act}} - \mathbf{kd}_{T_{Iact}} \cdot T_{I_{act}}$,

7. $\dot{I}_{act} = \frac{k_{catI} \cdot T_{I_{act}} \cdot I_{out}}{K_M + I_{out}} - \mathbf{kd}_{I_{act}} \cdot I_{act}$,

8. $\dot{R}_{act} = \frac{1}{6} \cdot \frac{k_{cat}^{R_{act},R} \cdot R_{act} \cdot M_{act}}{K_r + M_{act}} - \mathbf{kd}_{R_{act}} \cdot R_{act}$,

9. $\dot{lacI}_{act} = \frac{1}{6} \cdot \frac{k_{cat}^{lacI_{act},R} \cdot R_{act} \cdot M_{act}}{K_r + M_{act}} - \mathbf{kd}_{lacI_{act}} \cdot lacI_{act}$,

10. $\dot{M}_{rep} = \frac{k_{catN} \cdot N \cdot T_{rep}}{K_M + N} - \frac{kr \cdot R_{rep} \cdot M_{rep}}{Kr + M_{rep}}$,

11. $\dot{A}_{rep} = \frac{k_{catA} \cdot T_{A_{rep}} \cdot A_{out}}{K_M + A_{out}} - \mathbf{kd}_{A_{rep}} \cdot A_{rep}$,

12. $\dot{T}_{rep} = \frac{1}{6} \cdot \frac{k_{cat}^{T_{rep},R} \cdot R_{rep} \cdot M_{rep}}{K_r + M_{rep}} - \mathbf{kd}_{T_{rep}} \cdot T_{rep}$,

13. $\dot{T}_{A_{rep}} = \frac{1}{6} \cdot \frac{k_{cat}^{T_{A_{rep}},R} \cdot R_{rep} \cdot M_{rep}}{K_r + M_{rep}} - \mathbf{kd}_{T_{A_{rep}}} \cdot T_{A_{rep}}$,

14. $\dot{T}_{I_{rep}} = \frac{1}{6} \cdot \frac{k_{cat}^{T_{I_{rep}},R} \cdot R_{rep} \cdot M_{rep}}{K_r + M_{rep}} - \mathbf{kd}_{T_{I_{rep}}} \cdot T_{I_{rep}}$,

15. $\dot{I}_{rep} = \frac{1}{6} \cdot \frac{k_{cat}^{I_{rep},R} \cdot R_{rep} \cdot M_{rep}}{K_r + M_{rep}} - \frac{k_{catI} \cdot T_{I_{rep}} \cdot I_{rep}}{K_M + I_{rep}} - \mathbf{kd}_{I_{rep}} \cdot I_{rep}$,

16. $\dot{R}_{rep} = \frac{1}{6} \cdot \frac{k_{cat}^{R_{rep},R} \cdot R_{rep} \cdot M_{rep}}{K_r + M_{rep}} - \mathbf{kd}_{R_{rep}} \cdot R_{rep}$,

17. $\dot{lacI}_{rep} = \frac{1}{6} \cdot \frac{k_{cat}^{lacI_{rep},R} \cdot R_{rep} \cdot M_{rep}}{K_r + M_{rep}} - \mathbf{kd}_{lacI_{rep}} \cdot lacI_{rep}$,

18. $\dot{A}_{out} = \frac{k_{catA} \cdot T_{A_{act}} \cdot A_{act}}{K_M + A_{act}} - \frac{k_{catA} \cdot T_{A_{rep}} \cdot A_{out}}{K_M + A_{out}}$,

19. $\dot{I}_{out} = \frac{k_{catI} \cdot T_{I_{rep}} \cdot I_{rep}}{K_M + I_{rep}} - \frac{k_{catI} \cdot T_{I_{act}} \cdot I_{out}}{K_M + I_{out}}$.

w	θ_A	θ_I	θ_{lacI}	K_M	k_r	K_r
100 mg/mmol	0.3 mmol	0.2 mmol	0.1 mmol	1000	1260	7

Table 7.2: k_r denotes the elongation rate and K_r, K_M are Michaelis constants.

Dynamic oscillations

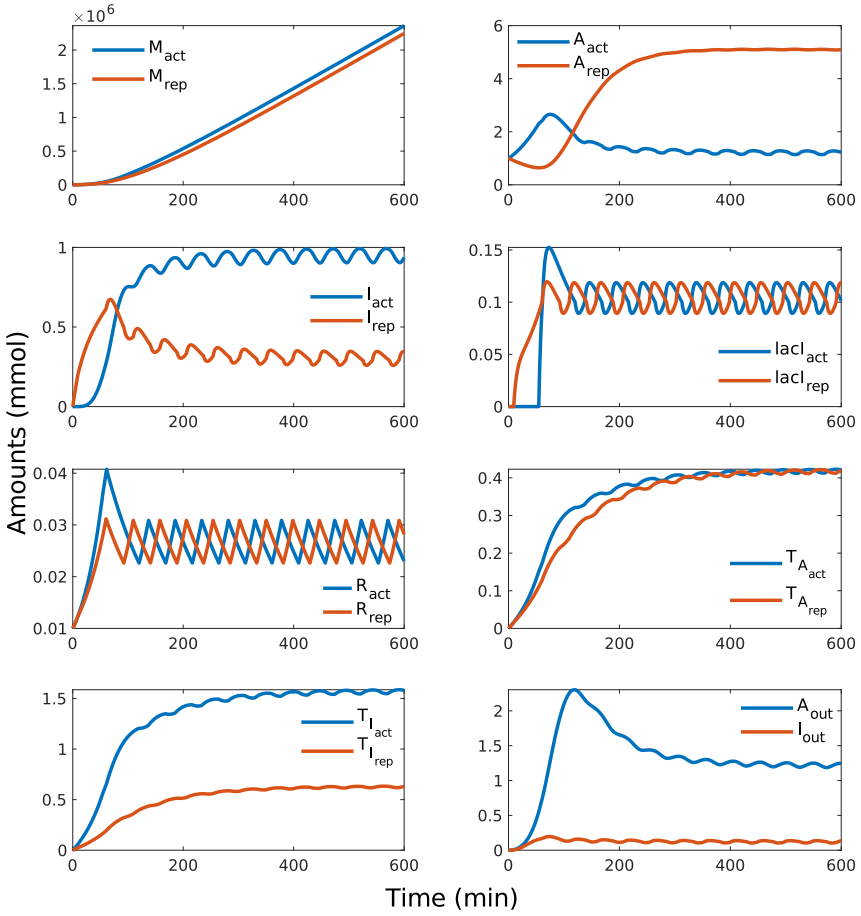


Figure 7.7: A dynamic trajectory of $H_{act-rep}$ over time period $[0, 600]$.

To understand the dynamic behaviors of $H_{act-rep}$, we set the nutrients N to $+\infty$. With the initial values

M_{act}	A_{act}	I_{act}	T_{act}	R_{act}	$T_{A_{act}}$	$T_{I_{act}}$	$lacI_{act}$	A_{out}
100	1	0	0	0.01	0	0	0	0

(7.12)

and

M_{rep}	A_{rep}	I_{rep}	T_{rep}	R_{rep}	$T_{A_{rep}}$	$T_{I_{rep}}$	$lacI_{rep}$	I_{out}
100	1	0	0	0.01	0	0	0	0

(7.13)

we obtain a dynamic trajectory of $H_{act-rep}$ in time span $[0, 600]$, see Figure 7.7. We can see that $H_{act-rep}$ starts in state $(\texttt{on}, \texttt{off}, \texttt{on}, \texttt{off})$ based on the initial values. Both $lacI_{act}$ and $lacI_{rep}$ are not produced in the beginning. However, with the production of the signaling molecule I_{rep}, $\overline{lacI}_{rep}$ turns on. This is the first state transition, with jump to state $(\texttt{on}, \texttt{off}, \texttt{on}, \texttt{on})$. Then, I_{act} also reaches its threshold by transporting signaling molecule I_{out} from the medium. The discrete state transits to $(\texttt{on}, \texttt{on}, \texttt{on}, \texttt{on})$. As the regulatory proteins $lacI_{act}$ and $lacI_{rep}$ are both synthesized in this state, $lacI_{act}$ reaches θ_{lacI} next. So, the state jumps to $(\texttt{off}, \texttt{on}, \texttt{on}, \texttt{on})$. Several minutes later, $lacI_{rep}$ also reaches its threshold and the discrete state turns to $(\texttt{off}, \texttt{on}, \texttt{off}, \texttt{on})$. In Figure 7.7, this is the first time R_{act} and R_{rep} decline. Then, the evolution of proteins exhibits the rhythms shown in Figure 7.7.

The amounts of the signaling molecules A_{act}, A_{rep}, I_{act} and I_{rep} always oscillate within a range, see their dynamics in Figure 7.7. However, the four signaling molecules are always above their thresholds. This means that during the stable oscillation, both $\overline{lacI}_{act}$ and $\overline{lacI}_{rep}$ are on. An interesting observation is that $lacI_{act}$, $lacI_{rep}$ do not keep increasing, albeit with on expression state but oscillate around the threshold $\theta_{lacI} = 0.1$. With the regulatory rules

$$\overline{R}_{act} = \texttt{on} \;\Leftrightarrow\; (A_{act} \geq \theta_A \;\text{ and }\; lacI_{act} < \theta_{lacI}), \tag{7.8}$$

$$\overline{R}_{rep} = \texttt{on} \;\Leftrightarrow\; (A_{rep} \geq \theta_A \;\text{ and }\; lacI_{rep} < \theta_{lacI}), \tag{7.9}$$

as a result, $\overline{R}_{act}$ and $\overline{R}_{rep}$ are shifting between on and off due to the oscillation of $lacI_{act}$ and $lacI_{rep}$ above and below θ_{lacI}.

Take the activator strain as an example. Since A_{act} is always oscillating above θ_A, $\overline{R}_{act}$ jumps to off only if $lacI_{act}$ above its threshold θ_{lacI}. In state off, the ribosome production is inhibited and R_{act} decreases. However, the reduction of the ribosome amount further decreases the protein expression level. Thus, $lacI_{act}$ will decline if its expression cannot compensate for the degradation. Whenever $lacI_{act}$ is below θ_{lacI}, $\overline{R}_{act}$ turns on again and R_{act} increases. When $lacI_{act} \geq \theta_{lacI}$, $\overline{R}_{act}$ turns off again. Then comes another turn.

The dynamics of the total biomass of the two strains is shown in Figure 7.8. The exponential increase of the biomass in both strains demonstrates that the community composed of the activator strain and the repressor strain lives well when cultured together, albeit with the positive and negative regulation.

In conclusion, this case study shows that there is a stable oscillation when the "activator" strain and the "repressor" strain in Figure 7.6 are cultured together. Both of them can survive and neither strain dies in our simulation even if the parameters, particularly the initial values, are set without any experimental data. We believe that modeling the metabolite dynamics and gene regulation at population-level is of great significance in the creation of cooperative strains for metabolic engineering or synthetic biology.

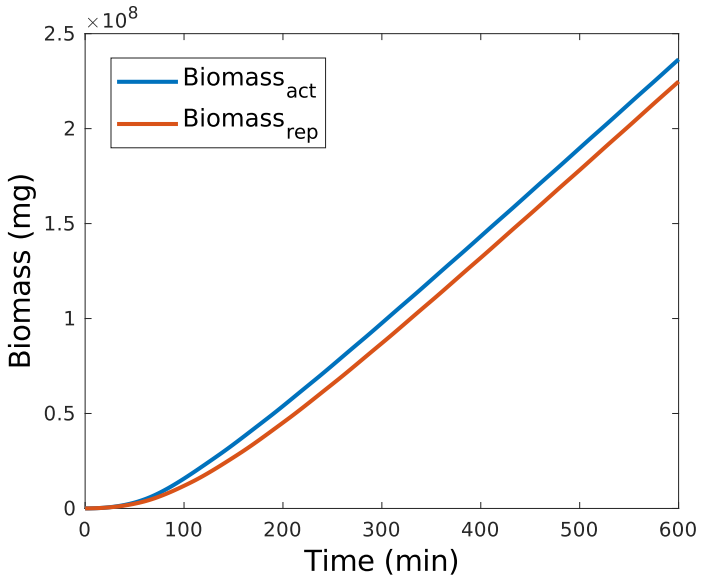


Figure 7.8: Dynamics of total biomass of the activator strain and the repressor strain.

7.5 Conclusion and discussion

In this chapter, we formalize population-level metabolic-regulatory networks by product automata, which allows including the metabolic interactions and transcriptional regulation between community members. As a proof of concept, we have exemplified two case studies to understand the population-level dynamics under gene regulation. The mathematical framework we introduce in this chapter allows not only an understanding of the metabolically cooperative and competitive relationships of cells, but also how metabolic dynamics and discrete states are affected by the shared extracellular resources and signaling molecules.

Yet, the cellular interactions considered in our product automata are mainly metabolic exchange interactions, e.g. competition for shared nutrients or mutual profits by cross-feeding, and transcriptional responses to signaling molecules of neighboring cells. It remains an enormous challenge to elucidate more general cellular interactions between community members, especially when physical interaction and signal transduction are included.

Up till now, we have shown several applications of the formalization of MRNs by hybrid automata in Chapter 5. We are able to explore the discrete and continuous dynamics of the system, and we can predict the optimal dynamics achieving maximal biomass. We also introduced the theory of product automata to study microbial communities. Yet, there is a precondition necessary for these applications: we need precise kinetic parameters to specify the hybrid automata. However, the enzymatic kinetic parameters are still highly difficult to obtain and most of them are unknown currently.

Chapter 7. Perspectives: Formalizing metabolic-regulatory networks at population-level by product automata

Model checking, as a common verification technique, provides another promising direction for making use of our hybrid automata describing metabolic-regulatory systems, especially when the kinetic parameters are unknown. In a hybrid system, the soundness of functionality depends on the parameter setting. Identifying the right constraints on the free parameters ensuring the sound functionality of a hybrid automaton has promising implications. In our hybrid automata of MRNs, the kinetic parameters define the continuous evolution of metabolites and are also important for the state transitions. The parameters also affect the correct functionality of a metabolic-regulatory network such as the management of the diauxic shift. Therefore, finding the suitable constraints for the unknown kinetic parameters would be among the most important problems in synthetic biology and engineering.

We believe this could be a promising way to do model-checking for a cellular metabolic-regulatory system. There are approaches available for proving the correctness of hybrid systems, although it is one of the most challenging problems in computer science and mathematics. For example, *differential dynamic logic* ($d\mathcal{L}$) has been introduced for reasoning and analysis of hybrid automata (Platzer, 2007). What's more, a hybrid theorem prover KeYmaera had been developed for validating hybrid systems (Platzer and Quesel, 2008). In the end, we hope that the hybrid system tools for simulation, reachability analysis, reasoning, verification can be used to study cellular mechanisms and make contributions to synthetic biology and biotechnologies.

Chapter 8

Conclusion

This PhD thesis focused on integrating the dynamics of metabolism, protein expression, transcriptional regulation, and resource allocation. Before the start of this PhD project, the constraint-based modeling approaches capable of dynamically integrating metabolic networks and gene regulatory rules were limited to rFBA (Covert et al., 2001) and FlexFlux (Marmiesse et al., 2015). Also, protein production costs and enzyme-reaction catalytic relationships were not covered in both methods. From another perspective, despite the fact that RBA (Goelzer and Fromion, 2011), deFBA (Waldherr et al., 2015) and cFBA (Rügen et al., 2015) had been developed to predict the resource allocation either at steady-state or in a dynamic setting, there were no algorithms allowing the dynamic combination of metabolism with enzyme-costs and the gene regulatory control at the beginning of this PhD project.

This PhD project was initially inspired by the paper (Covert et al., 2001), which is an early approach to the integration of metabolic networks with Boolean gene regulatory rules. We had the idea to explore the optimal solution space in rFBA after fully studying the paper and noticing that the solution is not unique. This is the first contribution of this thesis and is detailed in Chapter 3. We specifically proposed an analytical pipeline to explore the optimal solution space and implemented it on a core carbon network and the central metabolic network of *Escherichia coli*. By decomposing the vertices into a minimal set of EFMs, this analytic pipeline enables us to investigate which EFMs the model could utilize to achieve the maximal growth rate, and how they shifted during growth with the control of genetic regulation.

Hereafter, we aimed to incorporate the dynamics of protein expression and gene regulation into classical metabolic network models. At first, the intuitive strategy is combining Boolean regulatory rules with resource balance analysis (RBA) in an iterative way like in rFBA. Secondly, we can impose the updated regulatory state as bounds on the enzyme production rates since protein expression is covered in RBA. This is the work presented in Chapter 4, in which we first developed iRBA to

extend RBA to account for the dynamically changing extracellular metabolites and biomass. Then regulatory iRBA was proposed to further incorporate Boolean regulatory rules into the iterative procedure of iRBA. However, the dynamical change of proteins is not considered in both approaches, which limits their plausibility.

Next, in order to seamlessly bridge the dynamics of metabolism and transcriptional regulation, we further introduced *metabolic-regulatory networks* (MRNs), and formalized the MRNs by hybrid automata, combining continuous dynamics and discrete control. Besides classical mathematical frameworks of systems biology such as constraint-based approaches, ODE-based kinetic modeling and logic-based qualitative regulatory networks, our formalization in Chapter 5 provides an innovative mathematical framework that allows the integration of metabolic reactions, protein expression and transcriptional regulation within a hybrid discrete-continuous system. Also, this work makes it possible to utilize hybrid modeling tools such as model checking for the study of metabolic regulation.

One potential application is to investigate the optimal trajectories of the hybrid automata representing our MRNs. From a biological view, our metabolic-regulatory system basically covers biological processes from nutrient assimilation to gene expression under the control of gene regulation. Then a question arises: can we predict the dynamics of metabolism, resource allocation and discrete state transitions that maximize the cellular growth? The answer is given in Chapter 6, in which we developed a dynamic optimization approach termed as *regulatory dynamic enzyme-cost flux balance analysis* (r-deFBA) to predict the optimal strategy maximizing the biomass accumulation over a time interval of the hybrid automata describing MRNs. Unifying dynamic modeling of metabolism, resource allocation and transcriptional regulation control, our r-deFBA allows the prediction of the optimal dynamics of proteins, flux distributions, and state transitions.

As promising perspectives, we introduced the composition of hybrid automata representing MRNs to model the dynamics of metabolism and regulatory state of microbial populations in Chapter 7. Particularly, formalizing *metabolic-regulatory networks* at population-level as product automata provides a new framework for modeling the growth of microbial populations, while taking into account the cellular heterogeneity and the interactions between community members. For instance, one can consider the metabolic interactions, competition, cooperative behaviors and transcriptional reply in response to the signals of neighboring cells. Our framework allows the investigation of how a microbial community could be affected by diverse cells cultured together.

Bibliography

W. Abou-Jaoudé, P. Traynard, P. T. Monteiro, J. Saez-Rodriguez, T. Helikar, D. Thieffry, and C. Chaouiya. Logical modeling and dynamical analysis of cellular networks. *Frontiers in Genetics*, 7:94, 2016. 79

R. Aebersold and M. Mann. Mass spectrometry-based proteomics. *Nature*, 422 (6928):198, 2003. 1

R. Alur, C. Belta, F. Ivančić, V. Kumar, M. Mintz, G. J. Pappas, H. Rubin, and J. Schug. Hybrid modeling and simulation of biomolecular networks. In *International workshop on hybrid systems: Computation and control*, pages 19–32. Springer, LNCS 2034, 2001. 13

S. Anderson. Shotgun DNA sequencing using cloned DNase I-generated fragments. *Nucleic Acids Research*, 9(13):3015–3027, 1981. 1

B. Appel, L. A. Givan, and J. S. Eisen. Delta-Notch signaling and lateral inhibition in zebrafish spinal cord development. *BMC Developmental Biology*, 1(1):13, 2001. 116

G. Batt, D. Ropers, H. De Jong, J. Geiselmann, R. Mateescu, M. Page, D. Schneider, et al. Analysis and verification of qualitative models of genetic regulatory networks: A model-checking approach. In *International Joint Conference on Artificial Intelligence*, pages 370–375, 2005. 13

S. A. Becker, A. M. Feist, M. L. Mo, G. Hannum, B. Ø. Palsson, and M. J. Herrgard. Quantitative prediction of cellular metabolism with constraint-based models: the COBRA Toolbox. *Nature Protocols*, 2(3):727, 2007. 19

G. Bernot, J.-P. Comet, A. Richard, and J. Guespin. Application of formal methods to biological regulatory networks: extending Thomas' asynchronous logical approach with temporal logic. *Journal of Theoretical Biology*, 229(3):339–347, 2004. 13

A. Bockmayr and A. Courtois. Using hybrid concurrent constraint programming to model dynamic biological systems. In *International Conference on Logic Programming*, pages 85–99. Springer, LNCS 2401, 2002. 14, 76

A. Bordbar, J. M. Monk, Z. A. King, and B. Ø. Palsson. Constraint-based models predict metabolic and associated cellular functions. *Nature Reviews Genetics*, 15(2):107, 2014. 6, 59

L. Bortolussi and A. Policriti. Hybrid systems and biology. In *International School on Formal Methods for the Design of Computer, Communication and Software Systems*, pages 424–448. Springer, LNCS 5016, 2008. 14, 76

E. Bosdriesz, D. Molenaar, B. Teusink, and F. J. Bruggeman. How fast-growing bacteria robustly tune their ribosome concentration to approximate growth-rate maximization. *The FEBS journal*, 282(10):2029–2044, 2015. 69

F. Brandt, S. A. Etchells, J. O. Ortiz, A. H. Elcock, F. U. Hartl, and W. Baumeister. The native 3D organization of bacterial polysomes. *Cell*, 136(2):261–271, 2009. 82

H. Bremer, P. P. Dennis, et al. Modulation of chemical composition and other parameters of the cell by growth rate. *Escherichia coli and Salmonella: cellular and molecular biology*, 2(2):1553–69, 1996. 82

K. Brenner, L. You, and F. H. Arnold. Engineering microbial consortia: a new frontier in synthetic biology. *Trends in biotechnology*, 26(9):483–489, 2008. 116

A. P. Burgard, P. Pharkya, and C. D. Maranas. Optknock: a bilevel programming framework for identifying gene knockout strategies for microbial strain optimization. *Biotechnology and Bioengineering*, 84(6):647–657, 2003. 8

J. A. Campos-Ortega. Genetic mechanisms of early neurogenesis in *Drosophila melanogaster*. *Molecular Neurobiology*, 10(2-3):75–89, 1995. 116

B. Chance. The kinetics of the enzyme-substrate compound of peroxidase. *Journal of Biological Chemistry*, 151(2):553–577, 1943. 4

S. Chandrasekaran and N. D. Price. Probabilistic integrative modeling of genome-scale metabolic and regulatory networks in *Escherichia coli* and *Mycobacterium tuberculosis*. *Proceedings of the National Academy of Sciences of the United States of America*, 107(47):17845–50, 2010. 15, 35, 60

M. Chaves, D. A. Oyarzún, and J.-L. Gouzé. Analysis of a genetic-metabolic oscillator with piecewise linear models. *Journal of Theoretical Biology*, 462: 259–269, 2019. 12, 76

Y. Chen, J. K. Kim, A. J. Hirning, K. Josić, and M. R. Bennett. Emergent genetic oscillations in a synthetic microbial consortium. *Science*, 349(6251):986–989, 2015. 116, 124

M. W. Covert and B. Ø. Palsson. Transcriptional regulation in constraints-based metabolic models of *Escherichia coli*. *Journal of Biological Chemistry*, 277(31): 28058–28064, 2002. 35

M. W. Covert and B. Ø. Palsson. Constraints-based models: Regulation of gene expression reduces the steady-state solution space. *Journal of Theoretical Biology*, 221(3):309 – 325, 2003. 35, 36, 43

M. W. Covert, C. H. Schilling, and B. Ø. Palsson. Regulation of gene expression in flux balance models of metabolism. *Journal of Theoretical Biology*, 213(1): 73 – 88, 2001. 14, 20, 21, 22, 35, 39, 40, 41, 54, 60, 66, 70, 75, 76, 78, 88, 92, 93, 107, 114, 133

M. W. Covert, E. M. Knight, J. L. Reed, M. J. Herrgard, and B. Ø. Palsson. Integrating high-throughput and computational data elucidates bacterial networks. *Nature*, 429(6987):92 – 96, 2004. 35

M. W. Covert, N. Xiao, T. J. Chen, and J. R. Karr. Integrating metabolic, transcriptional regulatory and signal transduction models in *Escherichia coli*. *Bioinformatics*, 24(18):2044–2050, 2008. 14, 35, 60

H. De Jong. Modeling and simulation of genetic regulatory systems: A literature review. *Journal of Computational Biology*, 9(1):67–103, 2002. 6, 11

H. De Jong, J.-L. Gouzé, C. Hernandez, M. Page, T. Sari, and J. Geiselmann. Hybrid modeling and simulation of genetic regulatory networks: A qualitative approach. In *International Workshop on Hybrid Systems: Computation and Control*, pages 267–282. Springer, LNCS 2623, 2003. 14, 76

H. De Jong, J.-L. Gouzé, C. Hernandez, M. Page, T. Sari, and J. Geiselmann. Qualitative simulation of genetic regulatory networks using piecewise-linear models. *Bulletin of Mathematical Biology*, 66(2):301–340, 2004. 11

H. De Jong, S. Casagranda, N. Giordano, E. Cinquemani, D. Ropers, J. Geiselmann, and J.-L. Gouzé. Mathematical modelling of microbes: metabolism, gene expression and growth. *Journal of the Royal Society Interface*, 14(136): 20170502, 2017. 115

C. G. de Oliveira Dal'Molin, L.-E. Quek, R. W. Palfreyman, S. M. Brumbley, and L. K. Nielsen. AraGEM, a genome-scale reconstruction of the primary metabolic network in Arabidopsis. *Plant Physiology*, 152(2):579–589, 2010. 3

E. Dekel and U. Alon. Optimality and evolutionary tuning of the expression level of a protein. *Nature*, 436(7050):588, 2005. 9

V. Devloo, P. Hansen, and M. Labbé. Identification of all steady states in large networks by logical analysis. *Bulletin of Mathematical Biology*, 65(6):1025–1051, 2003. 13

N. C. Duarte, S. A. Becker, N. Jamshidi, I. Thiele, M. L. Mo, T. D. Vo, R. Srivas, and B. Ø. Palsson. Global reconstruction of the human metabolic network based on genomic and bibliomic data. *Proceedings of the National Academy of Sciences*, 104(6):1777–1782, 2007. 3

R. J. Duffin, G. B. Dantzig, and K. Fan. *Linear inequalities and related systems*. Number 38. Princeton university press, 1956. 97

J. S. Edwards, R. U. Ibarra, and B. Ø. Palsson. *In silico* predictions of *Escherichia coli* metabolic capabilities are consistent with experimental data. *Nature Biotechnology*, 19(2):125, 2001. 8

M. Faizi, T. Zavřel, C. Loureiro, J. Ječvený, and R. Steuer. A model of optimal protein allocation during phototrophic growth. *BioSystems*, 166:26–36, Apr 2018. 82

A. Fauré, A. Naldi, C. Chaouiya, and D. Thieffry. Dynamical analysis of a generic boolean model for the control of the mammalian cell cycle. *Bioinformatics*, 22 (14):e124–e131, 2006. 13

A. M. Feist, C. S. Henry, J. L. Reed, M. Krummenacker, A. R. Joyce, P. D. Karp, L. J. Broadbelt, V. Hatzimanikatis, and B. Ø. Palsson. A genome-scale metabolic reconstruction for *Escherichia coli* K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information. *Molecular Systems Biology*, 3(1):121, 2007. 2, 3, 45

E. Fischer and U. Sauer. Large-scale *in vivo* flux analysis shows rigidity and suboptimal performance of *Bacillus subtilis* metabolism. *Nature genetics*, 37(6): 636, 2005. 111

J. Förster, I. Famili, P. Fu, B. Ø. Palsson, and J. Nielsen. Genome-scale reconstruction of the *Saccharomyces cerevisiae* metabolic network. *Genome Research*, 13 (2):244–253, 2003a. 3

J. Förster, I. Famili, B. Ø. Palsson, and J. Nielsen. Large-scale evaluation of in silico gene deletions in *Saccharomyces cerevisiae*. *OMICS: A Journal of Integrative Biology*, 7(2):193–202, 2003b. 8

R. Ghosh and C. J. Tomlin. Lateral inhibition through delta-notch signaling: A piecewise affine hybrid model. In *International Workshop on Hybrid Systems: Computation and Control*, pages 232–246. Springer, LNCS 2034, 2001. 14

N. Giordano, F. Mairet, J.-L. Gouzé, J. Geiselmann, and H. De Jong. Dynamical allocation of cellular resources as an optimal control problem: novel insights into microbial growth strategies. *PLoS Computational Biology*, 12(3):e1004802, 2016. 9

L. Glass and S. A. Kauffman. The logical analysis of continuous, non-linear biochemical control networks. *Journal of Theoretical Biology*, 39(1):103–129, 1973. 11

A. Goelzer and V. Fromion. Bacterial growth rate reflects a bottleneck in resource allocation. *Biochimica et Biophysica Acta (BBA)-General Subjects*, 1810(10): 978–988, 2011. 9, 22, 60, 69, 133

A. Goelzer, V. Fromion, and G. Scorletti. Cell design in bacteria as a convex optimization problem. *Automatica*, 47(6):1210–1218, 2011. 9, 27, 28, 59, 69, 76

A. Goelzer, J. Muntel, V. Chubukov, M. Jules, E. Prestel, R. Nölker, M. Mariadassou, S. Aymerich, M. Hecker, P. Noirot, et al. Quantitative prediction of genome-wide resource allocation in bacteria. *Metabolic Engineering*, 32:232–243, 2015. 59, 69

B. Görke and J. Stülke. Carbon catabolite repression in bacteria: many ways to make the most out of nutrients. *Nature Reviews Microbiology*, 6(8):613–624, 2008. 81

W. Gottstein, B. G. Olivier, F. J. Bruggeman, and B. Teusink. Constraint-based stoichiometric modelling from single organisms to microbial communities. *Journal of the Royal Society Interface*, 13(124):20160627, 2016. 116

M. Grötschel, L. Lovsz, and A. Schrijver. Geometric algorithms and combinatorial optimization. *Algorithms and Combinatorics*, 2:65–84, 1988. 37

L. H. Hartwell, J. J. Hopfield, S. Leibler, and A. W. Murray. From molecular to modular cell biology. *Nature*, 402(6761supp):C47, 1999. 1

R. Heinrich and S. Schuster. *The regulation of cellular systems*. Chapman and Hall, 1996. 5

T. A. Henzinger. The theory of hybrid automata. In *Verification of Digital and Hybrid Systems*, pages 265–292. Springer, 2000. 80, 81

M. J. Herrgård, B.-S. Lee, V. Portnoy, and B. Ø. Palsson. Integrated analysis of regulatory and metabolic networks reveals novel regulatory mechanisms in *Saccharomyces cerevisiae*. *Genome Research*, 16(5):627–635, 2006. 14, 35

A. V. Hill. The possible effects of the aggregation of the molecules of haemoglobin on its dissociation curves. *The Journal of Physiology*, 40:4–7, 1910. 6, 10

H.-G. Holzhütter. The principle of flux minimization and its application to estimate stationary fluxes in metabolic networks. *European Journal of Biochemistry*, 271 (14):2905–2922, 2004. 8

F. Jacob and J. Monod. Genetic regulatory mechanisms in the synthesis of proteins. *Journal of Molecular Biology*, 3(3):318–356, 1961. 11, 12

G. Jeanne, A. Goelzer, S. Tebbani, D. Dumur, and V. Fromion. Dynamical resource allocation models for bioreactor optimization. *IFAC-PapersOnLine*, 51(19):20–23, 2018. 9, 60, 76

P. A. Jensen, K. A. Lutz, and J. A. Papin. TIGER: Toolbox for integrating genome-scale metabolic models, expression data, and transcriptional regulatory networks. *BMC Systems Biology*, 5(1):147, 2011. 15, 97, 99

S. Kauffman, C. Peterson, B. Samuelsson, and C. Troein. Random Boolean network models and the yeast transcriptional network. *Proceedings of the National Academy of Sciences*, 100(25):14796–14799, 2003. 12

S. A. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of Theoretical Biology*, 22(3):437–467, 1969. 12

S. M. Kelk, B. G. Olivier, L. Stougie, and F. J. Bruggeman. Optimal flux spaces of genome-scale stoichiometric models are determined by a few subnetworks. *Scientific Reports*, 2(580), 2012. 36, 37, 38

D. Kennell and H. Riezman. Transcription and translation initiation frequencies of the *Escherichia coli* lac operon. *Journal of Molecular Biology*, 114(1):1–21, 1977. 82

I. M. Keseler, J. Collado-Vides, A. Santos-Zavaleta, M. Peralta-Gil, S. Gama-Castro, L. Muñiz-Rascado, C. Bonavides-Martinez, S. Paley, M. Krummenacker, T. Altman, et al. EcoCyc: a comprehensive database of *Escherichia coli* biology. *Nucleic Acids Research*, 39(suppl_1):D583–D590, 2010. 82

Z. A. King, C. J. Lloyd, A. M. Feist, and B. Ø. Palsson. Next-generation genome-scale models for metabolic engineering. *Current Opinion in Biotechnology*, 35: 23–29, 2015. 6

H. Kitano. Systems biology: a brief overview. *Science*, 295(5560):1662–1664, 2002. 1

A. L. Knorr, R. Jain, and R. Srivastava. Bayesian-based selection of metabolic objective functions. *Bioinformatics*, 23(3):351–357, 2006. 8

A. L. Koch. Why can't a cell grow infinitely fast? *Canadian Journal of Microbiology*, 34(4):421–426, 1988. 9

A. Kremling, J. Geiselmann, D. Ropers, and H. de Jong. Understanding carbon catabolite repression in *Escherichia coli* using quantitative models. *Trends in Microbiology*, 23(2):99–109, 2015. 81

A. Kremling, J. Geiselmann, D. Ropers, and H. De Jong. An ensemble of mathematical models showing diauxic growth behaviour. *BMC Systems Biology*, 12 (1):82, 2018. 76, 81

N. Le Novere, B. Bornstein, A. Broicher, M. Courtot, M. Donizelli, H. Dharuri, L. Li, H. Sauro, M. Schilstra, B. Shapiro, et al. BioModels Database: a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. *Nucleic Acids Research*, 34(suppl_1):D689–D691, 2006. 3

J. M. Lee, E. P. Gianchandani, J. A. Eddy, and J. A. Papin. Dynamic analysis of integrated signaling, metabolic, and regulatory networks. *PLoS Computational Biology*, 4(5):e1000086, 2008. 14, 60

J. A. Lerman, D. R. Hyduke, H. Latif, V. A. Portnoy, et al. *In silico* method for modelling metabolism and gene product expression at genome scale. *Nature Communications*, 3, 2012. 9, 22, 59, 60, 76

N. E. Lewis, H. Nagarajan, and B. Ø. Palsson. Constraining the metabolic genotype-phenotype relationship using a phylogeny of *in silico* methods. *Nature Reviews Microbiology*, 10(4):291–305, Feb 2012. 59

F. Li, T. Long, Y. Lu, Q. Ouyang, and C. Tang. The yeast cell-cycle network is robustly designed. *Proceedings of the National Academy of Sciences*, 101(14): 4781–4786, 2004. 12

P. Lincoln and A. Tiwari. Symbolic systems biology: Hybrid modeling and analysis of biological networks. In *International Workshop on Hybrid Systems: Computation and Control*, pages 660–672. Springer, LNCS 2993, 2004. 13

L. Liu and A. Bockmayr. Hybrid modeling of metabolic-regulatory networks (extended abstract). In *International Workshop on Hybrid Systems Biology*, pages 177–180. Springer, LNCS 11705, 2019a. 75

L. Liu and A. Bockmayr. Formalizing metabolic-regulatory networks by hybrid automata. *Acta Biotheoretica*, Jul 2019b. 75, 93, 114

L. Liu and A. Bockmayr. Regulatory dynamic enzyme-cost flux balance analysis: A unifying framework for constraint-based modeling. *bioRxiv 802249*, 2019c. 91

C. J. Lloyd, A. Ebrahim, L. Yang, Z. A. King, E. Catoiu, E. J. O'Brien, J. K. Liu, and B. Ø. Palsson. COBRAme: A computational framework for genome-scale models of metabolism and gene expression. *PLoS Computational Biology*, 14 (7):e1006302, 2018. 6

N. M. Luscombe, M. M. Babu, H. Yu, M. Snyder, S. A. Teichmann, and M. Gerstein. Genomic analysis of regulatory network dynamics reveals large topological changes. *Nature*, 431(7006):308, 2004. 13

T. R. Maarleveld, M. T. Wortel, B. G. Olivier, B. Teusink, and F. J. Bruggeman. Interplay between constraints, objectives, and optimality for genome-scale stoichiometric models. *PLoS Computational Biology*, 11(4), 2015. 36, 37, 38

L. Magatao. *Mixed integer linear programming and constraint logic programming: towards a unified modeling framework*. PhD thesis, 2005.

R. Mahadevan and C. Schilling. The effects of alternate optimal solutions in constraint-based genome-scale metabolic models. *Metabolic Engineering*, 5(4): 264 – 276, 2003. 8, 36, 37

R. Mahadevan, J. S. Edwards, and F. J. D. III. Dynamic flux balance analysis of diauxic growth in *Escherichia coli*. *Biophysical Journal*, 83(3):1331 – 1340, 2002. 17, 18, 59, 60

A. Maitra and K. A. Dill. Bacterial growth laws reflect the evolutionary importance of energy efficiency. *Proceedings of the National Academy of Sciences*, 112(2): 406–411, 2015. 121

L. Marmiesse, R. Peyraud, and L. Cottret. FlexFlux: combining metabolic flux and regulatory network analyses. *BMC Systems Biology*, 9(1):93, 2015. 14, 60, 75, 133

A. G. Marr. Growth rate of *Escherichia coli*. *Microbiological Reviews*, 55(2): 316–333, 1991. 27

A. M. Maxam and W. Gilbert. A new method for sequencing DNA. *Proceedings of the National Academy of Sciences*, 74(2):560–564, 1977. 1

H. H. McAdams and L. Shapiro. Circuit simulation of genetic networks. *Science*, 269(5224):650–656, 1995. 13

L. Menten and M. Michaelis. Die Kinetik der Invertinwirkung. *Biochem Z*, 49: 333–369, 1913. 5, 118

D. Molenaar, R. van Berlo, D. de Ridder, and B. Teusink. Shifts in growth strategies reflect tradeoffs in cellular economics. *Molecular Systems Biology*, 5(1), 2009. 9, 22, 76, 78

J. Monod. The growth of bacterial cultures. *Annual Reviews in Microbiology*, 3 (1):371–394, 1949. 118

A. M. New, B. Cerulus, S. K. Govers, G. Perez-Samper, B. Zhu, S. Boogmans, J. B. Xavier, and K. J. Verstrepen. Different levels of catabolite repression optimize growth in stable and variable environments. *PLoS Biology*, 12(1):e1001764, 2014. 88

E. J. O'Brien, J. A. Lerman, R. L. Chang, D. R. Hyduke, and B. Ø. Palsson. Genome-scale models of metabolism and gene expression extend and refine growth phenotype prediction. *Molecular Systems Biology*, 9(1), 2013. 59

M. S. Okino and M. L. Mavrovouniotis. Simplification of mathematical models of chemical reaction systems. *Chemical Reviews*, 98(2):391–408, 1998. 6, 11

F. R. P. B. Orth, J. Reconstruction and use of microbial metabolic networks: the core *Escherichia coli* metabolic model as an educational guide. *EcoSal Plus*, 2010. 2, 3, 45, 49

J. D. Orth, I. Thiele, and B. Ø. Palsson. What is flux balance analysis? *Nature Biotechnology*, 28(3):245–248, Mar 2010. 8, 17

B. Palsson. The challenges of in silico biology. *Nature Biotechnology*, 18(11): 1147, 2000. 7

A. Platzer. Differential dynamic logic for verifying parametric hybrid systems. In *International Conference on Automated Reasoning with Analytic Tableaux and Related Methods*, pages 216–232. Springer, LNCS 4416, 2007. 132

A. Platzer and J.-D. Quesel. KeYmaera: A hybrid theorem prover for hybrid systems (system description). In *International Joint Conference on Automated Reasoning*, pages 171–178. Springer, LNCS 5195, 2008. 132

N. D. Price, J. A. Papin, and B. Ø. Palsson. Determination of redundancy and systems properties of the metabolic network of *Helicobacter pylori* using genome-scale extreme pathway analysis. *Genome Research*, 12(5):760–769, 2002. 8

N. D. Price, J. L. Reed, and B. Ø. Palsson. Genome-scale models of microbial cells: evaluating the consequences of constraints. *Nature Reviews Microbiology*, 2(11):886, 2004. 7, 8

H. Quastler. *Information theory in biology*. University of Illinois Press, 1953. 1

J.-F. Raskin. An introduction to hybrid automata. In D. Hristu-Varsakelis and W. Levine, editors, *Handbook of Networked and Embedded Control Systems*, pages 491–517. Birkhäuser Boston, 2005. 80, 81, 119

A.-M. Reimers. *Understanding metabolic regulation and cellular resource allocation through optimization*. PhD thesis, 2017. 33, 97

A.-M. Reimers, H. Knoop, A. Bockmayr, and R. Steuer. Cellular trade-offs and optimal resource allocation during cyanobacterial diurnal growth. *Proceedings of the National Academy of Sciences*, page 201617508, 2017a. 60, 76, 97, 99

A.-M. Reimers, H. Lindhorst, and S. Waldherr. A protocol for generating and exchanging (genome-scale) metabolic resource allocation models. *Metabolites*, 7(3):47, 2017b. 31, 77, 79, 93, 94

D. Ropers, H. De Jong, M. Page, D. Schneider, and J. Geiselmann. Qualitative simulation of the carbon starvation response in *Escherichia coli*. *Biosystems*, 84 (2):124–152, 2006. 76

N. Rosenfeld, J. W. Young, U. Alon, P. S. Swain, and M. B. Elowitz. Gene regulation at the single-cell level. *Science*, 307(5717):1962–1965, 2005. 6

M. Rügen, A. Bockmayr, and R. Steuer. Elucidating temporal resource allocation and diurnal dynamics in phototrophic metabolism using conditional FBA. *Scientific Reports*, 5, 2015. 9, 60, 76, 133

A. Samal and S. Jain. The regulatory network of *E. coli* metabolism as a Boolean dynamical system exhibits both homeostasis and flexibility of response. *BMC Systems Biology*, 2(1):21, 2008. 12, 20, 35

F. Sanger and A. R. Coulson. A rapid method for determining sequences in DNA by primed synthesis with DNA polymerase. *Journal of Molecular Biology*, 94 (3):441–448, 1975. 1

J. M. Savinell and B. Ø. Palsson. Network analysis of intermediary metabolism using linear optimization. I. development of mathematical formalism. *Journal of Theoretical Biology*, 154(4):421 – 454, 1992. 8

R. Schuetz, L. Kuepfer, and U. Sauer. Systematic evaluation of objective functions for predicting intracellular fluxes in *Escherichia coli*. *Molecular Systems Biology*, 3(1), 2007. 8

S. Schuster and C. Hilgetag. On elementary flux modes in biochemical reaction systems at steady state. *Journal of Biological Systems*, 2(02):165–182, 1994. 38

S. Schuster, C. Hilgetag, J. H. Woods, and D. A. Fell. Reaction routes in biochemical reaction systems: algebraic properties, validated calculation procedure and example from nucleotide metabolism. *Journal of Mathematical Biology*, 45(2): 153–181, 2002. 38

M. Scott, S. Klumpp, E. M. Mateescu, and T. Hwa. Emergence of robust growth laws from optimal regulation of ribosome synthesis. *Molecular Systems Biology*, 10(8), 2014. 69

C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27(4):379–423, 1948. 1

T. Shlomi, O. Berkman, and E. Ruppin. Regulatory on/off minimization of metabolic flux changes after genetic perturbations. *Proceedings of the national academy of sciences*, 102(21):7695–7700, 2005. 8

T. Shlomi, Y. Eisenberg, R. Sharan, and E. Ruppin. A genome-scale computational study of the interplay between transcriptional regulation and metabolism. *Molecular Systems Biology*, 3(1):101, 2007. 14, 15, 60, 97, 99

H. Siebert and A. Bockmayr. Temporal constraints in the logical analysis of regulatory networks. *Theoretical Computer Science*, 391(3):258–275, 2008. 13

S. O. Simmons, C.-Y. Fan, and R. Ramabhadran. Cellular stress response pathway system as a sentinel ensemble in toxicological screening. *Toxicological Sciences*, 111(2):202–225, 2009. 77

H. A. Simon. The architecture of complexity. In *Facets of systems science*, pages 457–476. Springer, 1991. 1

R. Singhania, R. M. Sramkoski, J. W. Jacobberger, and J. J. Tyson. A hybrid model of mammalian cell cycle regulation. *PLoS Computational Biology*, 7(2): e1001077, 2011. 14

E. H. Snoussi. Qualitative dynamics of piecewise-linear differential equations: a discrete mapping approach. *Dynamics and Stability of Systems*, 4(3-4):565–583, 1989. 11

J. Stülke and W. Hillen. Carbon catabolite repression in bacteria. *Current Opinion in Microbiology*, 2(2):195–201, 1999. 41

A. Succurro, F. W. Moejes, and O. Ebenhöh. A diverse community to study communities: integration of experiments and mathematical models to study microbial consortia. *Journal of Bacteriology*, 199(15):e00865–16, 2017. 116

M. Sugita. Functional analysis of chemical systems in vivo using a logical circuit equivalent. II. the idea of a molecular automaton. *Journal of Theoretical Biology*, 4(2):179–192, 1963. 11, 12, 13

M. Terzer. *Large scale methods to enumerate extreme rays and elementary modes*. PhD thesis, ETH Zurich, 2009. 8

M. Terzer and J. Stelling. Large-scale computation of elementary flux modes with bit pattern trees. *Bioinformatics*, 24(19):2229–2235, 2008. 36, 38

D. Thieffry, M. Colet, and R. Thomas. Formalisation of regulatory nets: A logical method and its automatization. *Mathematical Modelling and Scientific Computing*, 2:144–151, 1993. 13

I. Thiele, N. Jamshidi, R. M. Fleming, and B. Ø. Palsson. Genome-scale reconstruction of *Escherichia coli*'s transcriptional and translational machinery: a knowledge base, its mathematical formulation, and its functional characterization. *PLoS Computational Biology*, 5(3):e1000312, 2009. 9

R. Thomas. Boolean formalization of genetic control circuits. *Journal of Theoretical Biology*, 42(3):563–585, 1973. 11, 12, 88

R. Thomas and M. Kaufman. Multistationarity, the basis of cell differentiation and memory. I. Structural conditions of multistationarity and other nontrivial behavior. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 11(1): 170–179, 2001. 13, 114

H. E. Umbarger. Amino acid biosynthesis and its regulation. *Annual Review of Biochemistry*, 47(1):533–606, 1978. 40

A. Varma and B. Ø. Palsson. Stoichiometric flux balance models quantitatively predict growth and metabolic by-product secretion in wild-type *Escherichia coli* W3110. *Applied and Environmental Microbiology*, 60(10):3724–3731, Oct 1994. 18, 59, 60

E. O. Voit, H. A. Martens, and S. W. Omholt. 150 years of the mass action law. *PLoS Computational Biology*, 11(1):e1004012, 2015. 5

P. Waage and C. Guldberg. Studier over affiniteten. *Forhandlinger I Videnskabs-selskabet I Christiania*, 1:35–45, 1864. 5, 118

S. Waldherr, D. A. Oyarzún, and A. Bockmayr. Dynamic optimization of metabolic networks coupled with gene expression. *Journal of Theoretical Biology*, 365:469 – 485, 2015. 9, 29, 32, 60, 66, 76, 79, 92, 93, 94, 107, 114, 133

Y. Wang, T. Joshi, X.-S. Zhang, D. Xu, and L. Chen. Inferring gene regulatory networks from multiple microarray datasets. *Bioinformatics*, 22(19):2413–2420, 2006. 15

A. Y. Weiße, D. A. Oyarzún, V. Danos, and P. S. Swain. Mechanistic links between cellular trade-offs, gene expression, and growth. *Proceedings of the National Academy of Sciences of the United States of America*, 112(9):E1038–E1047, 2015. 9, 82, 84, 121

H. V. Westerhoff and B. Ø. Palsson. The evolution of molecular biology into systems biology. *Nature Biotechnology*, 22(10):1249, 2004. 1

N. Wiener. Cybernetics, or Communication and Control in the Animal and the Machine. *New York: ffiley*, 23, 1948. 1

L. Yang, A. Ebrahim, C. J. Lloyd, M. A. Saunders, and B. Ø. Palsson. DynamicME: Dynamic simulation and refinement of integrated models of metabolism and protein expression. *BMC Systems Biology*, 13(1):2, 2019. 9, 60, 63, 64, 72, 76

P. Ye, E. Entcheva, S. A. Smolka, and R. Grosu. Modelling excitable cells using cycle-linear hybrid automata. *IET Systems Biology*, 2(1):24–32, 2008. 14

I. Yegorov, F. Mairet, H. De Jong, and J.-L. Gouzé. Optimal control of bacterial growth for the maximization of metabolite production. *Journal of Mathematical Biology*, 78(4):985–1032, 2019. 9

Zusammenfassung

Metabolische und genregulatorische Netzwerke sind zwei klassische Modelle der Systembiologie. Biologisch bilden genregulatorische Netzwerke das Kontrollsystem der Proteinexpression, während metabolische Netzwerke, insbesondere die genomweiten Rekonstruktionen, aus Tausenden enzymatischer Reaktionen bestehen, die Nährstoffe in Ausgangsstoffe und Energie aufspalten, um das Überleben der Zelle zu sichern. Metabolisch-genetische Netzwerke umfassen darüber hinaus die Translationsprozesse als integriertes Modell klassischer metabolischer Netzwerke und der Genexpressionsmaschinerie. Umgekehrt wird die genetische Regulation auch durch metabolische Aktivitäten beeinflusst, die Feedback und Ausgangsstoffe für das regulatorische System liefern. Die beiden Systeme interagieren somit in hohem Maße und sind voneinander abhängig.

Bisher wurden bereits vielfältige Anstrengungen unternommen, um die beiden Netzwerktypen miteinander zu verbinden. Doch die dynamische Integration von metabolischen Netzwerken und Genregulation bleibt eine große Herausforderung der rechnergestützten Systembiologie.

Diese Dissertation ist ein Beitrag zu mathematischen Modellierungsansätzen für die Untersuchung metabolisch-regulatorischer Systeme. Inspiriert von der *regulatorischen Fließ-Gleichgewichts-Analyse* (rFBA) führen wir zunächst eine analytische Pipeline ein, um den optimalen Lösungsraum der rFBA zu untersuchen. Anschließend konzentrieren wir uns auf die dynamische Kombination metabolischer Netzwerke mit Enzymproduktionskosten und genetischer Regulation. Dazu verfolgen wir zunächst die intuitive Idee, boolesche Regulierungsregeln in eine iterierte *Ressourcen-Gleichgewichts-Analyse* zu integrieren. Bei der iterativen Strategie werden die Genexpressionszustände jedoch nur in diskreten Zeitschritten aktualisiert. Die Formalisierung *metabolisch-regulatorischer Netzwerke* (MRN) durch Hybridautomaten liefert darüber hinaus ein neues mathematisches Rahmenwerk. Dieses ermöglicht die quantitative Integration metabolisch-genetischen Netzwerken und Genregulation in ein hybrides diskret-kontinuierliches System. Zur Anwendung dieser theoretischen Formalisierung entwickeln wir den constraintbasierten Ansatz der *regulatorisch-dynamischen Fließ-Gleichgewichts-Analyse von Enzymkosten* (r-deFBA) als optimale Steuerungsstrategie für die Hybridautomaten, welche die MRN darstellen. Dies ermöglicht die Vorhersage optimaler regulatorischer Zustandsübergänge, der Dynamik des Stoffwechsels und der Ressourcenallokation, welche eine maximale Biomasseproduktion über ein simuliertes Zeitintervall erzielen. Die Dissertation endet mit einem Kapitel über Perspektiven: Wir benutzen die Theorie der Produktautomaten, um die Dynamik auf Populationsebene zu modellieren, welche kontinuierliche Stoffwechselprozesse und diskrete Regulation umfasst.

Acknowledgements

This PhD study has been one of the best and most challenging times of my life. It would not have been possible to do without the support, guidance, and warm encouragement from many people.

First and foremost, I owe my deepest gratitude to my PhD supervisor, Alexander Bockmayr. Thanks to him that I had the opportunity to pursue my PhD study in FUB and to work in computational systems biology as a student with the biological background. He helped me come up with the research ideas in the thesis and guided me through this challenging journey with great patience and excellent supervision. Without his invaluable suggestions and constant feedbacks, this PhD thesis would not have been achievable. Many thanks to you, Alexander. You have shown me, by your example, what a good scientist should be, and I have learned a lot from working with you.

My sincere thanks go to Hidde de Jong for accepting to be my second reviewer of this thesis. Many thanks for his time committed to reading and evaluating this thesis.

I wish to thank all the present and former members of the research groups, Mathematics in Life Science, and Discrete Biomathematics for contributing to a friendly research environment. Many thanks go to you all: Annika, Alexandra, Hannes, Katinka, Markus, Ling, Manon, Neveen, Therese, Heike, Adam, Elisa, Kirsten, Laura, Melania, Robert, and Katja! I am especially grateful to Alexandra and Annika who helped me a lot at the start of my PhD, when I was a beginner in this field. Moreover, I want to thank Markus for the help and fruitful discussions on Chapter 5 and Chapter 6. Also, a big thanks to Markus and Ling for proof-reading and commenting on several chapters.

I would like to acknowledge the financial support from China Scholarship Council and DFG GRK 1772 Computational Systems biology, which allow me to concentrate on the research.

I must express my profound appreciation to my family, my parents, my brother and sisters for always supporting and encouraging me to pursue my dreams. I wish to thank my loving and supportive boyfriend, Panshuo Wang. Thank you all for your love.

Last but not least, I want to thank all the friends I met in Berlin during this PhD study, for their love and by the side. I will always treasure the time we party, travel, cook, chat, discus, etc. I would not go through this challenging journey without all of you. Many thanks to you all!

Berlin, den January 7, 2020

Lin Liu

Declaration

Ich versichere diese Arbeit selbständig, ohne unerlaubte Hilfsmittel verfasst und keine außer den angeführten Quellen verwendet zu haben. Die Arbeit ist nicht schon einmal in einem früheren Promotionsverfahren eingereicht worden.

Berlin, den January 7, 2020

Lin Liu